Noname manuscript No.
(will be inserted by the editor)

Unequal Error Protection for Data Partitioned H.264/AVC Video Broadcasting

Sajid Nazir · Dejan Vukobratović · Vladimir Stanković · Ivan Andonović · Kristian Nybom · Stefan Grönroos

Received: date / Accepted: date

Abstract Application Layer Forward Error Correction (AL-FEC) is becoming a popular addition to protocols for real-time video delivery over IP-based wireless networks. In particular, rateless codes are identified as suitable solution for AL-FEC due to their flexibility and capacity-approaching performance. Since each part of video data is not equally important for video reconstruction, it is beneficial to group it based on its importance, and then provide different degree of protection using Unequal Error Protection (UEP). Data partitioning (DP) is one such low-cost feature in H.264/AVC enabling partitioning of video data based on its importance. We propose schemes for the DP H.264/AVC video transmission using Raptor and Random Linear Codes (RLC) and investigate their performance as AL-FEC solutions in Digital Video Broadcasting. We provide comparisons between optimized Non-Overlapping Window RLC and Expanding Window (EW) RLC, which are two effective UEP RLC strategies. The results using realistic channel traces show viability of the EW RLC as a promising solution for multimedia broadcast applications.

Keywords Video compression, rateless coding, Digital Video Broadcasting, joint source-channel coding

Nazir, Stanković, Andonović
Dept of EEE, University of Strathclyde, Glasgow, UK.
Tel: +44-(141)-5482679
Fax: +44-(141)-5524968
E-mail: vladimir.stankovic@strath.ac.uk

Vukobratović
University of Novi Sad, Serbia
E-mail: dejanv@uns.ac.rs

Nybom and Grönroos
Åbo Akademi University, Turku, Finland
E-mail: kristian.nybom@abo.fi

1 Introduction

Multimedia broadcasting over wireless networks is rapidly increasing. Since wireless channels are prone to errors and packet losses, error mitigation measures must be incorporated. Forward Error Correction (FEC) is the favored approach as retransmissions in broadcasting applications are usually counter-productive [1], [2]. Indeed, FEC codes are successfully applied in Digital Video Broadcast (DVB) networks. For example, DVB-Handheld (DVB-H) specifies a Multi-Protocol Encapsulation - FEC (MPE-FEC) solution at the link layer designed for real-time services [3], which applies adaptive punctured Reed-Solomon (RS) codes against packet losses. The DVB-H standard also provides a possibility of using Application Layer (AL) FEC solution for Internet Protocol (IP) datacasting services using a well-known class of rateless codes called Digital Fountain (DF) Raptor codes [2], [4]. Though DF Raptor codes are currently used only for non real-time services, they have been investigated for multi-burst protection and compared to MPE-FEC in terms of performance and delay for real-time services over DVB-H [5] and DVB-NGH (Next Generation Handheld).

Apart from DF Raptor codes, a class of rateless codes that have been gaining increased popularity recently for applications in wireless networks are Random Linear Codes (RLC) [6], [7]. RLC show near-optimal performance even for very low codeword lengths, but suffer from high decoding complexity of the Gaussian Elimination (GE) decoder as the codeword length increases. However, for real-time video transmission, RLC is usually applied over short chunks of video data (called generations) thus working in the domain of practically implementable codeword lengths [8], [9]. In addition, as multihop and cooperative communications are becoming increasingly popular in emerging wireless network architectures, introduction of RLC may serve as a step forward towards exploring the benefits of network coding [6]. In the context of future DVB networks, this may be important for emerging concepts such as hybrid broadcast/cellular networks (with clients equipped with multiple wireless broadband interfaces) and device-to-device communications [10].

The focus of this study is to analyse the use of the data partitioning (DP) error resilient feature of H.264/AVC [13] in combination with Raptor codes and RLC as an AL-FEC solution for video broadcasting. Furthermore, the benefits of unequal error protection (UEP) RLC, tailored with the DP feature of H.264/AVC is investigated in detail. The DP feature of H.264/AVC is chosen because it effectively prioritizes video stream by partitioning it into classes of decreasing importance to video reconstruction with a very small decrease in overall performance. This enables simple rate adaptation crucial in wireless broadcasting [3], [14].

UEP using H.264/AVC with DP is suggested in [16]. DP H.264/AVC and UEP are later used for wireless video delivery in [17] using rate compatible punctured convolutional (RCPC) codes, in [18] with hierarchical quadrature amplitude modulation, in [19] with Raptor codes for IPTV, and in [20] with growth codes. In [21], design of a Raptor generator matrix based on frame dependencies within a group of picture (GOP) has been proposed. The redundancy allocation process is tied to the knowledge of the channel loss which may not be suitable for real-time applications and video broadcasting. The proposed scheme also needs a mechanism to transport the generator matrix coefficients to decoder. DP H.264/AVC with UEP is used in [22] where only two most important partitions are protected using rateless codes, while the third partition is sent unprotected.

Expanding Window Fountain (EWF) codes [23] are LT-based UEP layered scheme where the protection of lesser important layers also includes the more important layers. In [23], EWF codes are optimized for scalable video delivery. In [24], a layer-aware FEC mechanism has been proposed, using UEP low-density parity-check (LDPC) codes and Raptor codes for protection of H.264 Scalable Video Coding (SVC) video. The work also explores the suitability of the proposed codes to physical and application layer FEC protection for the latest video communication standards including DVB-H. Extending the UEP design methods of [23] from DF to RLC, UEP-based RLC strategies for scalable code delivery have been investigated in [25].

While [25] is more focused on performance analysis of UEP RLCs schemes over random erasure channel, this paper addresses design challenges of UEP RLCs schemes with DP H.264/AVC for real-world implementation for broadcasting and compares their performance to standard DF Raptor codes. While both DF Raptor codes and RLC have been individually well studied in literature, interestingly, no systematic performance/complexity comparison has been conducted yet. The results reported here are based on DVB-H and on DVB-terrestrial 2 (DVB-T2) [26] configured according to the DVB-T2 Lite profile, which is a profile intended to allow simpler receiver implementations for very low capacity applications such as mobile broadcasting, although it may also be received by conventional stationary receivers. Thus, the goal of this paper is to evaluate performance of the proposed schemes in a real-world environment and provide several optimized design recommendations.

The rest of the paper is structured as follows. Section 2 briefly covers the necessary background. The proposed UEP RLC schemes are described in Sections 3 and 4. Simulation results are given in Section 5. Finally, Section 6 concludes the paper.

2 Background

In this section we briefly review H.264/AVC and its DP feature, Raptor codes and RLC, and parts of DVB-H/DVB-T2 protocol stack relevant to this work.

2.1 H.264/AVC and Data Partitioning (DP)

H.264/AVC provides many error-resilience features to mitigate the effect of lost packets during transmission. One such scheme available in the extended profile is data partitioning [16] which supports the partitioning of a slice in up to three partitions (denoted as A, B, and C) based on the importance of the encoded video syntax elements to video reconstruction. Data partition A (DP A) contains the most important data comprising slice header, quantization parameters, and motion vectors. DP B contains the intra-coded macroblocks (MB) residual data (Intra coded block pattern and Intra coefficients) and requires that DP A is fully decoded. DP C contains inter-coded MB residual data (Inter coded block pattern and Inter coefficients) and is the least important partition.

The decoding of DP A is always independent of DP B and C. However, if a DP A is lost the remaining partitions cannot be utilized. The decoding of DP B is possible without DP C by setting the Constrained Intra Prediction (CIP) parameter in H.264/AVC encoding [27]. Despite a loss of a packet that carries DPs B or C, DP A can still be decoded; however, this loss could result in error propagation to the subsequent frames, which would degrade the video quality beyond the effect of the lost Network Abstraction Layer (NAL) unit. To offset the error propagation, insertion of periodic I (intra) frames could be used but is rate costly. A relatively low cost solution is to insert periodic MB intra updates (MBIU) which can limit the error propagation to a certain extent. It is shown in [20], that DP has the least overhead as compared to other common error resilience techniques in H.264/AVC.

2.2 Raptor Codes and RLC

Raptor codes [4] are rateless (fountain) codes, i.e., they provide a flexibility to generate as many encoded symbols as desired from the set of source symbols. The Raptor decoder can recover the original source symbols from any set of encoded symbols, as long as their number is at least equal (or more precisely, at least slightly exceeds) the number of source symbols. These have recently been adopted for use in various communications standards including DVB-H. As an AL-FEC solution in DVB-H, systematic Raptor codes provide improved system reliability and a large degree of freedom in the choice of transmission parameters [5]. Raptor codes have a constant encoding and linear decoding complexity. For an explanation of the encoding and decoding algorithms and implementation guidelines, see [28].

Another class of rateless codes which has become popular recently are RLC [6]. RLC applied over a source message produce encoded symbols as random linear combinations of source symbols with coefficients randomly selected from a given finite field. As a packet level AL-FEC solution, RLC are simple to implement and perform as near-optimal erasure codes for sufficiently large finite field used for creating linear combinations of source symbols (one-byte field GF(256) is usually sufficiently good [6]). This makes RLC an attractive alternative to Raptor codes as a universal FEC/network coding solution for emerging wireless communication systems, such as LTE-A, mobile WiMAX, and DVB-NGH [7], [10], [11]. The major limitation of the application of RLC is the decoding complexity of GE decoding, which is polynomial in the number of symbols. However, for short lengths of the source messages (in the order of up to 256 symbols) and small GF, the decoding complexity is acceptable (see [8], [9], [12] and references therein). Moreover, as shown later in the paper, there is no performance penalty for using short-length codes as compared to standard rateless codes.

2.3 DVB-H and DVB-T2

DVB-H is the European standard for digital TV signals broadcast to handheld devices and is based on the DVB-T standard. DVB-H is IP-based and has two important new

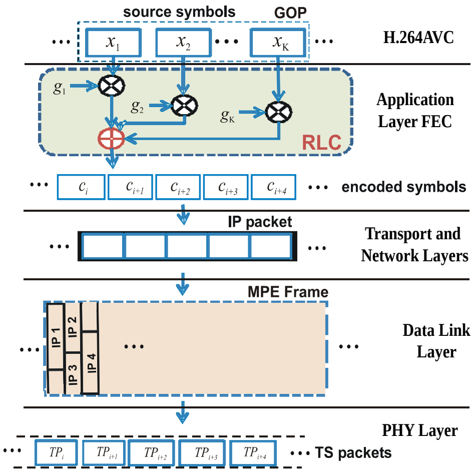

Fig. 1 DVB-H protocol structure.

features introduced in it. Firstly, the transmission takes place in intermittent bursts of maximum data rate of up to 2Mbps. Secondly, to mitigate errors in mobile and wireless environments the DVB-H standard introduces an optional link-layer FEC mechanism, called MPE-FEC.

Assuming that the H.264/AVC standard is used for video encoding, the DVB-H protocol stack, which successively adds headers and prepares the video data for transmission, is shown in Fig. 1. The H.264/AVC encoder generates NAL units which are then encapsulated into RTP/UDP packets. These packets are then placed into IP packets, which are inserted into MPE sections column by column. The MPE frame containing MPE sections is then divided into transport stream (TS) packets for transmission over the physical DVB-H layer (PHY). All TS packets belonging to one MPE frame are sent in one transmission burst. Note that, if AL-FEC is used, the link-layer FEC mechanism (based on RS codes) is switched off.

If a whole IP packet cannot fit into one TS packet, then it will be carried by two or more TS packets. Moreover, a single TS packet can contain data from two consecutive IP packets. Clearly, due to encapsulation the total overhead could be substantial especially for small IP packet sizes. The header overhead and unused bytes in a data packet are detrimental to the performance of the FEC scheme. Hence it is important to provide FEC configurations which can be adapted to the lower layers of the protocol structure.

As compared to DVB-T, DVB-T2 provides increased performance for HDTV broadcasting through the use of state-of-the-art technologies at the physical layer [26]. An important feature of the DVB-T2 standard is the use of physical layer pipes (PLPs). The TS services are assigned to separate PLPs, and each PLP can have a code rate, modulation, and time interleaving length of its own, allowing for service specific robustness. Additionally, the DVB-T2 Lite profile has recently been specified in Annex I of the DVB-T2 standard [26]. The DVB-T2 Lite profile is intended to allow simpler receiver implementations for very low capacity applications such as mobile broadcasting, although it may also be received by conventional stationary receivers. T2-Lite is based on a limited sub-set of the modes of the T2-base profile, and by avoiding modes which require the most complexity and memory, allows much more efficient receiver designs to be used.

3 Video Broadcasting Based on DP H.264/AVC and UEP AL-FEC

In the following, we propose a source-channel coding scheme for video broadcasting that exploits both state-of-the-art error resilient video coding and rateless AL-FEC to protect video against packet losses.

3.1 Video Encoding

For each non-I frame, the video encoder output is split into three partitions: DP A, DP B, and DP C, as explained in the previous section. The encoding is done with the CIP flag set and the MBIU feature to limit the effect of error propagation. All the partitions of a particular type of all frames within one GOP are extracted and aggregated together. That is, DPs A from all encoded frames of a GOP together with the encoded I frame are grouped generating the most important source class or layer. DPs B and C form second and third importance class/layers, respectively.

Each of the layers is packetized into equal-length source symbols/packets. Note that the length of the source symbols determines the number of source symbols contained in the source message, k, which directly influences FEC efficiency and decoding complexity. For Raptor codes, ideally, short source symbols should be selected resulting in higher values of source message length k, typically in the order of thousands of symbols. The performance is improved for larger k whereas decoding complexity increases only linearly with k. In the case of RLC, however, large source symbols are favoured in order to yield small values of k, typically in the order of hundreds, because of high decoding complexity of RLC. The key issue to be investigated is the impact of different selection of k, imposed by performance and complexity issues, to these two codes.

3.2 AL-FEC Coding

A straightforward approach to AL-FEC is to apply one code over the entire source block (i.e., a GOP), that is, to equally protect the whole stream. This, equal error protection (EEP), however, does not capture adequately unequal importance of the three classes of the DP H.264/AVC coded stream.

A preferable option is a UEP scheme which assigns redundancy to the source classes based on their importance to video reconstruction. We explore two different UEP design approaches. The first one uses a separate AL-FEC code, either Raptor or

RLC, for each of the DP layers, and by using the codes of different code rate, UEP is realized. The second approach uses the expanding window approach [23], [25] to encode jointly all the classes.

In the first scheme, each class of DP H.264/AVC is protected with a separate AL-FEC code. The level of protection will decrease from the first class onwards to account for the decrease of importance of subsequent classes. Following the terminology introduced in [25], we call this scheme the non-overlapping window (NOW) scheme, where each class is regarded as one window.

The second approach follows the concept of [23], where EWF codes as a class of UEP fountain codes are proposed. EWF codes are based on the idea of creating a set of nested windows over the source block. The rateless encoding process is then adapted to use this windowing information while producing encoded packets. We define the set of windows over the groups of source symbols of unequal importance, which are generated as a result of the DP H.264/AVC video stream. The coding is then performed over progressively increasing source block subset windows aligned with this most-to-least importance subsets. The decoding of a window is the same as FEC decoding. In [25], the expanding window approach is applied to RLC, leading to an EW-RLC scheme.

The encoding process for both NOW and EW codes has one important initial step, that is, to first select a window from which the encoded symbol is to be generated. This selection of a window is independently performed for each encoded symbol and is determined by selection probability of a window which is a pre-assigned parameter keeping in mind the importance of different layers and the data rate available.

3.3 Data Packetization

After AL-FEC coding, NOW or EW with Raptor or RLC codes, encoded symbols are grouped to fit the content of a single IP packet. This process also takes into consideration the header overhead. IP packets are placed into the MPE frames where each IP packet is encapsulated within a single MPE section with its own overhead containing the error-detection field.

Each MPE frame is transmitted within a single transmission burst by mapping the MPE frame data onto 188-bytes long PHY TS packets. We always set the IP packet size to be an integer multiple of 184 bytes (the content size of TS packet), i.e., more precisely, each MPE section is transmitted over the integer number of TS packets. IP packets are either correctly received at the decoder or discarded due to erroneous content and therefore lost in the transmission process. An IP packet is considered lost, if at least one TS packet that forms it is lost, which is the standard operation of the DVB-H link layer [5], [29].

For simplicity, we assume that MPE sections and TS packets are aligned, i.e., the borders between MPE sections at the Link layer are borders between TS packets at the physical layer. Unfortunately, in practice this is not always possible, and that is why sometimes a single TS packet loss can cause a loss of two consecutive IP packets. However, since this situation is rare we neglect it and assume perfect alignment of MPE sections and TS packets.

4 System Parameter Selection

In this section, we experimentally compare the performance of the standard Raptor codes with RLC for different source message sizes in order to select source symbol size, message size, and IP packet size that provide the best tradeoff between coding efficiency and delay/complexity over a range of channel parameters. Then, after setting the message/packet/symbol size the UEP optimization can be done to find the optimal error protection strategies.

4.1 Symbol Size Selection for Raptor and RLC Codes

We perform simulations using DVB-H TS packet error traces obtained by realistic DVB-H link layer simulator [30] to determine adequate message/symbol/packet size for systematic Raptor codes (RTC) and RLC. We compare different RTC and RLC coding configurations in terms of reconstruction performance as well as decoding complexity, where a configuration refers to one pair of source symbol and IP packet size used.

After AL-FEC coding (with either RTC or RLC), encoded symbols are grouped into sets of $N_s \geq 1$ symbols and packed into a single IP packet. The number of encoded symbols N_s carried by an IP packet depends on the symbol size l_s, the IP packet size l_{IP} and the set of protocol headers within each IP packet, described as follows. Each IP packet contains a four-byte random number generator seed used to recreate the RTC/RLC encoding coefficients for all the encoding symbols contained within the packet. The remaining symbols derive their coefficients based on the first symbol's random number generator seed. Then, another 60 header bytes comprising RTP/UDP/IP headers are added to form an IP packet (this number can be reduced using robust header compression). Therefore, IP packet size equals:

$$l_{IP} = l_{OH}^{(IP)} + N_s \cdot l_s, \tag{1}$$

where $l_{OH}^{(IP)} = 64$ bytes.

After IP packetization, at lower DVB-H layers, each IP packet is arranged into a separate link-layer MPE section containing MPE header of $l_{OH}^{(MPE)} = 16$ bytes. MPE section is split into TS packets each with a four-byte header followed by 184 bytes of payload. Assuming that each IP packet with the corresponding MPE section overhead fits into exactly $N_{TS} = (l_{IP} + 16)/184$ TS packets, the total overhead per IP packet equals:

$$l_{OH} = l_{OH}^{(IP)} + l_{OH}^{(MPE)} + N_{TS} \cdot 4 = 80 + \frac{l_{IP} + 16}{184} \cdot 4. \tag{2}$$

Different IP packet sizes l_{IP} are tested ranging from 300 bytes to over 1450 bytes (we limit the IP packet size to 1500 bytes). First, the IP packet size l_{IP} is picked so that, jointly with MPE overhead, it fits a whole number of TS packets, i.e., $l_{IP} = N_{TS} \cdot 184 - 16$, where N_{TS} is a positive integer. Then, the available capacity of IP packet is divided into a selected number of N_s equal-length encoded symbols, where the symbol size is determined as $l_s = (l_{IP} - l_{OH}^{(IP)})/N_s$.

Table 1 System configurations.

FEC	l_{IP}[bytes]	N_{TS}	l_s[bytes]	N_s	$\frac{l_{OH}}{N_{TS}\cdot 188}$[%]
RTC656	352	2	144	2	23.40
RTC721	720	4	131	5	12.76
RTC1475	1088	6	62	16	9.22
RTC1086	1456	8	87	16	7.45
RTC136	1456	8	696	2	7.45
RLC144	720	4	656	1	12.76
RLC185	1088	6	512	2	9.22
RLC68	1456	8	1392	1	7.45
RLC136	1456	8	696	2	7.45

Let the total source bit budget be B bytes; for example, this is the size of a GOP. Then, the number of source symbols in the source message is $k = ceil(B/l_s)$. By varying l_s and l_{IP}, we obtain different coding configurations. We test five different RTCs and four RLCs shown in Table 1. Four configurations use l_{IP} in the range of $1400 - 1500$ bytes, two configurations use the IP size of roughly 1000 bytes, while three use very low IP packet sizes. This way we covered low, medium and large IP packet sizes.

The number within the FEC scheme's name denotes the source message length k, e.g., RTC656 is a Raptor code of source message length k = 656 symbols. The last column of the table describes the transmission efficiency in terms of percentage of overhead data within the total transmitted data. From (2), we note that with increasing N_{TS}, the overhead grows much slower than the total amount of transmitted data, which makes the overall transmission more efficient for larger IP packets.

The symbol sizes l_s are generally selected to be small for RTC, so as to yield larger source message size (in conventional FEC, the increase of message length generally improves performance). However, for a fair comparison, we also include RTC with smaller source message sizes. The reverse is true for RLC, i.e., larger symbol sizes l_s are selected so as to yield shorter source message sizes k to make the decoding complexity feasible. The selection of different l_{IP} for RTC and RLC is motivated by the need to design and test schemes with different N_{TS}, so that the scheme's response to the TS error trace files could be evaluated over different possibilities. l_{IP} is always picked so that after adding a 16-byte MPE header, an integer multiple of 184 (payload of a TS packet) bytes is obtained.

Let $B = 94,400$ bytes. Then, for example, in the RLC68 configuration, the source message length of $k = 68$ symbols results in the symbol size l=1392 bytes, which is obtained after adding padding zeros to $\lceil 94400/68 \rceil = 1389$ bytes. In this case we put $N_s = 1$ encoded symbol in the IP packet. This symbol gets added with 64 bytes of header data to yield an IP packet of size $l_{IP} = 1456$ bytes. During transmission, this IP packet occupies $N_{TS} = 8$ TS packets after having added the MPE header to the IP packet and includes the total overhead of l_{OH} bytes, resulting in the ratio of overhead data in the data stream equal $\frac{l_{OH}}{N_{TS}\cdot 188} = 7.45[\%]$.

4.2 Simulated EEP Performance of Raptor and RLC codes

In this section, video sequence Foreman in CIF format encoded using the H.264/AVC software JM version 16.2 [31] is used without any error resilient option. All simulations have been performed using a GOP size of 64 frames, one slice per frame, and a frame rate of 25 frames per second. This configuration implies a GOP length of 2.56 seconds of video data and is also used in [29].

The trace files with the losses of TS packets for different signal-to-noise ratio (SNR) in the DVB-H channel are obtained by using a realistic link-level simulator, which is confirmed to mimic actual reception in the field [30]. The link-level simulator takes as input TS packet error traces, analyzes them, and generates new TS packet error traces of arbitrary length with similar characteristics. In this sense, the link-level simulator provides a flexible and easy-to-use tool for generating realistic traces, while reducing physical layer simulation times. For generating the input to the link-level simulator, the physical layer simulation model assumes a TU-6 DVB-H channel model with a constant Doppler frequency of 10 Hz, 8k mode, and guard interval 1/4. The simulated modulation is 16-QAM encoded with a convolutional code of rate 1/2. MPE-FEC error correction is not used.

The service bit rate in these simulations was set to 452 kbps. To take into account the varying nature of mobile reception, the video bitrate was limited to 295 kbps, leaving 1/3 of the available bandwidth for transmitting redundant data. The video bitrate also implies a GOP size, and source bit budget B of 94,400 bytes.

Each FEC configuration is simulated with at least 1000 runs for each channel SNR value. As in [29], we use performance criteria called ESR5(20) (Erroneous Second Ratio) which defines a limit of at most 1 second of erroneous transmission within each 20-second segment of video transmission. Thus, we present our results as the minimum overall data rate (including all the headers), called measured data rate, that satisfies the ESR5(20) performance criteria vs. SNR in the DVB-H channel. The results for two different SNRs are shown in Table 2. The error-free data rate is the one which is required by the FEC scheme to transport the GOP data if the channel were error free. It can be seen from the table that the best results are obtained with RLC136 and RTC136 in terms of minimizing transmission overhead. However, note that RLC185, RLC68 and RTC1086 suffer only a negligible rate loss. Similar results are obtained for other SNRs and different rates.

We have also used expected peak signal-to-noise ratio (PSNR) averaged over all frames in the sequence as a criteria for quality comparison. In the case of failure of RLC decoding, it would not be possible to have any frames available for that GOP. Thus, the entire GOP is considered lost. In such cases, the decoder applies a simple error concealment technique by replacing the lost GOP by the last frame of the previously successfully decoded GOP. To compare the quality for each of the used configurations, the overall data rate was fixed to be 452 kbps and each configuration is subjected to the same loss patterns. The results are shown in Fig. 2. It can be seen that several configurations, RTC1475, RTC1086, RTC136, RLC68, RLC136 provide the best PSNR performance. The worst performing sequence is RTC656 (not shown in the figure for clarity) and then RTC721 which have the heaviest data overhead. At SNR=15dB all eight schemes reach error-free PSNR performance of 36.61dB.

Table 2 Raptor and RLC configurations results.

FEC	**Error Free Data Rate** [kbps]	**Measured Data Rate** SNR12 [kbps]	**Measured Data Rate** SNR14 [kbps]
RTC656	385.2	606.3	452.3
RTC721	341.5	559.3	406.5
RTC1475	334.5	560.5	394.8
RTC1086	325.5	554.6	394.8
RTC136	318.8	549.9	394.8
RLC144	337.3	549.9	401.8
RLC185	325.2	542.8	394.8
RLC68	318.8	564	390.1
RLC136	318.8	535.8	390.1

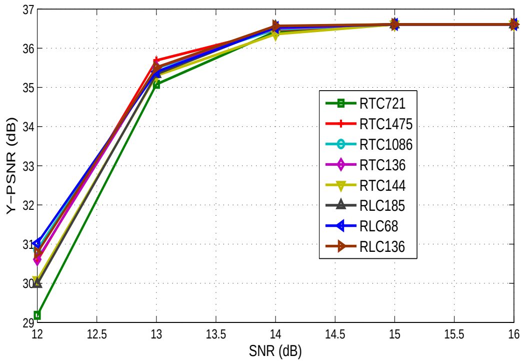

Fig. 2 PSNR comparison between the four selected EEP configurations: Y-PSNR vs. SNR in the DVB-H channel.

Besides performance comparison, it is important to conduct a complexity analysis as well. For source block size k and the number of received symbols n, the failure probability of an RTC for (n, k) decreases exponentially with increasing the number of received symbols. For roughly 12 additional symbols the failure probability is 0.1%, whereas for about 24 additional symbols the failure probability reduces to 0.0001% [1]. For the RLC case and GF(2^8) field size that we used in our setup, the failure probability with zero additional received symbols is around 0.01%, and with one additional received symbol, it drops below 0.000001%.

The decoding complexity of the RTC is linear with the source message length k, i.e., it grows as $O(k)$, thus mobile phones can easily handle decoding of RTC codes of message in the order of thousand symbols. On the other hand, the GE decoder of RLC codes has complexity that grows as $O(k^3)$. This clearly makes it impossible to use RLC for large source blocks. However, for short block lengths that we focus on, the asymptotic behavior does not say much about the decoding complexity. In several recent papers, the decoding complexity and possibilities of practical implementation of RLC is investigated [8], [9], [32], [34]. In [32], the study focused on streaming server solutions, and using optimized algorithms and GPU it is demonstrated that

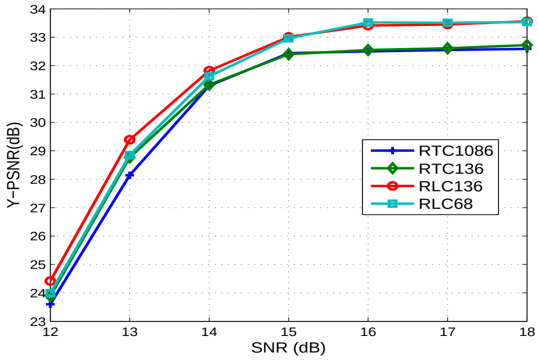

Fig. 3 PSNR comparison between different UEP RTC and RLC configurations: Y-PSNR vs. SNR in the DVB-H channel.

RLC with 128 blocks and large symbol sizes can achieve encoding rates up to 294 Mbps and decoding rates up to 254 Mbps. It is also argued therein that with such high rates, deploying RLC as an encoding solution is feasible for streaming servers. In [8] and [9], the study focused on mobile user applications within smart-phones, the symbol and block sizes for RLC similar to the one used in this study are practically implemented. It is concluded that the implementations of RLC decoding with short codes (source length $k = 64$ or 128) are feasible with standard video bit-rates of several hundreds of kbps. In [34], energy analysis of a custom VLSI implementation of RLC decoding is conducted showing that it is possible to design an ultra low-power implementation in the order of 10s mW, with a coding throughput of 60MB/s.

From the above analysis, we adopt two best performing schemes from each code class (RTC and RLC), i.e., RTC136, RTC1086, RLC68, and RLC136, having in mind that shorter message sizes are preferable option if performance is not severely affected.

5 DP H.264/AVC Video Broadcasting with UEP AL-FEC

In this section, we organize DPs into classes of different importance. Intuitively, we put Instantaneous Decoder Refresh (IDR) and DP A always in so called High Priority Class (HPC), because the remaining data cannot be decoded unless IDR and DPs A are correctly received. Furthermore, we create two schemes, RLC-IAB, where we put DP B together with DP A in the HPC and treat DP C as the less importance class called Low Priority Class (LPC). The size of HPC is then roughly 70% of the total size. However, the second layer provides a significant performance improvement of over 7dB of the Foreman sequence. In the second scheme, RLC-IA, we put only IDR and DP A into HPC, and DP B and DP C comprise LPC. The size of HPC is then about 55% of the total stream size.

Fig. 3 shows frame-average PSNR vs. channel SNR for the four selected RLC-IAB configurations, shown in Table 3, for the Foreman sequence when transmission

Table 3 Four selected coding configurations for the UEP analysis.

Scheme	**No of Symbols**		**Symbol Size**	**No. Symbols**
	HPC	LPC	[bytes]	**in IP packet**
RTC1086	776	305	87	16
RTC136	97	39	696	2
RLC68	49	19	1394	1
RLC136	97	39	696	2

data rate is fixed at 383 kbps and with selection probability of HPC of 75%. We present results with values of SNR that were available from the link-level simulator. RLC136 is the best performing configuration for all SNRs. The PSNR difference between RLC68 and RTC136 is marginal. RTC1086 provides the worst overall performance.

Having in mind the limited complexity of GE decoding for RLC68 (due to small code length), very small delay introduced, and competitive performance results, we conclude that it is a good candidate for real-world implementations. Thus, in all further simulations we present different UEP configurations obtained with RLC68.

If channel characteristics are known, it is possible to provide optimal allocation of redundancy to different source priority classes. For this task, we can maximize the expected PSNR at channel packet loss rate ϵ using analytically computed probabilities of decoding error performance. That is,

$$\max_{\boldsymbol{\pi}} PSNR(\boldsymbol{\pi}, \epsilon) = \sum_{i=0}^{L} P(i, \epsilon) psnr(i), \tag{3}$$

where L is the number of classes, $P(i, \epsilon)$ is probability that class i will be the highest class recovered at channel packet loss rate ϵ, $P(0)$ is the probability that nothing is recovered, $psnr(i)$ is PSNR of the reconstruction if all classes up to and including class i are recovered ($i = 1, \ldots, L$), $\boldsymbol{\pi}$ is an L-tuple vector of window selection probabilities that determines the UEP allocation scheme, and $PSNR(\boldsymbol{\pi}, \epsilon)$ is the expected PSNR when UEP scheme $\boldsymbol{\pi}$ is used. Analytical expressions for probabilities $P(i, \epsilon)$ assuming a random channel loss model, for NOW and EW schemes, can be found in [25].

Above, however, it is assumed that channel characteristics are known at the transmitter, which is unrealistic assumption in many broadcasting applications, such as DVB-H. Thus, instead, we maximize the average performance over a range of expected packet loss rates:

$$\max_{\boldsymbol{\pi}} \overline{PSNR(\boldsymbol{\pi})} = \sum_{\epsilon_j \in \Xi} \sum_{i=0}^{L} P(i, \epsilon_j) psnr(i), \tag{4}$$

where Ξ is the set of probable packet loss rates. This way, we aim to find a scheme that performs in average the best over a range of channel conditions. Since the number of possible UEP schemes is finite and small (L ==2 or 3), in the following we maximize (4) using exhaustive search, which is very fast.

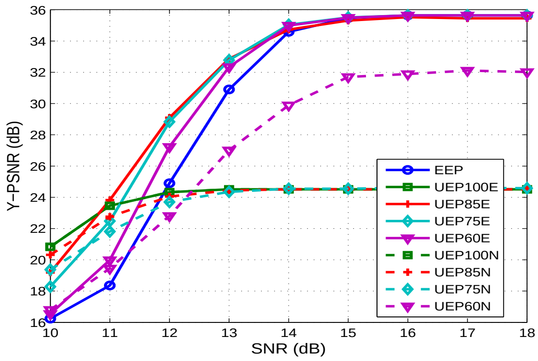

Fig. 4 PSNR vs. SNR in the DVB-H channel for four different NOW and EW UEP schemes and the single-layer EEP scheme of RLC68-IAB at data rate of 383 kbps.

5.1 Performance Evaluation of UEP NOW and EW RLC

In this subsection, we compare NOW and EW schemes using RLC68. We used four different video sequences: the standard CIF Foreman and Coastguard, QCIF Bus and 4CIF City sequence. The same scheme configuration as in the previous section is used. For illustration, the employed EW schemes for the Foreman sequence are shown in Table 4. Similar values are used for the other three sequences. The symbol size for all the schemes is 1394 bytes. EW1 refers to the single-layer EEP scheme (with all parameters set as in the EW2 scheme). Results for the Foreman sequence for four NOW UEP schemes, four EW2 UEP schemes and the EEP scheme of the RLC68-IAB scheme at data rate of 383 kbps are shown in Fig. 4. UEPxN, x=60, 75, 85, or 100 refers to a NOW UEP scheme with x% probability of selection for HPC. Similarly, UEPxE, is an EW scheme with x% probability of selection for HPC. We show frame-averaged PSNR vs. SNR in the DVB-H channel.

As it can be seen from the figure, the EW UEP schemes outperform EEP and NOW schemes for all SNRs. The EEP scheme outperforms the best NOW scheme for high SNRs, while for low SNR, the performance of the NOW UEP schemes is better. This is because at lower SNRs the contribution to PSNR by successful decoding of HPC is significant. As the SNR is increased, the UEP NOW curves get a performance penalty, with UEP schemes with higher protection of HPC being the ones which are severely affected. This is because at high SNRs the successful decoding of both classes is less probable due to higher selection probability of the HPC.

Superiority of the EW schemes over NOW schemes is clear. The main reason for performance gains of EW compared to NOW lies in the fact that NOW attempts to select symbols from either class in a mutually exclusive manner, in the sense, that if we increase the protection for HPC, this directly detriments LPC. In case of the EW schemes the windows are nested and correctly received packets from smaller windows may be used in recovery of larger windows [25]. The optimal EW scheme in terms of maximizing (4) was UEP(85,15), whereas the optimal NOW scheme was

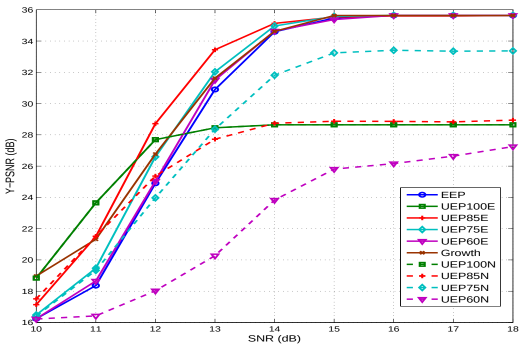

Fig. 5 PSNR vs. SNR in the DVB-H channel for four different NOW and EW UEP schemes and the single-layer EEP scheme of RLC-IA at data rate of 383 kbps. The standard CIF Foreman video sequence is used. ’Growth’ denotes the scheme of [20].

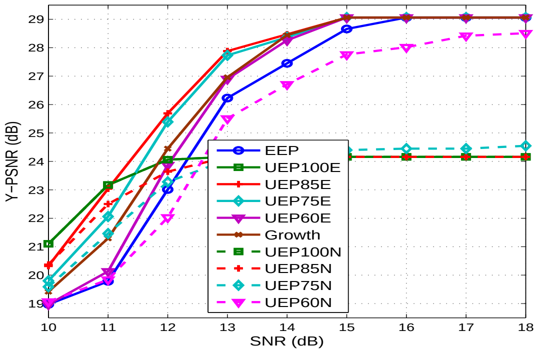

Fig. 6 PSNR vs. SNR in the DVB-H channel for four different NOW and EW UEP schemes and the single-layer EEP scheme of RLC-IA at data rate of 383 kbps. The standard CIF Coastguard video sequence is used. ’Growth’ denotes the scheme of [20].

UEP(60,40). In both cases we assume SNR values are integers between 10dB and 18dB, find average packet loss rates for each SNR, and perform maximization in (4).

The PSNR results obtained by varying the SNR over its entire available range and keeping the data rate fixed at 383 kbps are shown in Fig. 5 for RLC-IA. Similar conclusions as for the RLC-IAB scheme can be taken. NOW schemes again lag behind EW for higher SNRs. It can be seen from the two figures that the optimal EW2 UEP(85,15) scheme is superior to all the other schemes including EEP, except at low SNRs, when UEP(100,0) is better for only up to 1dB. This justifies the proposed optimization strategy and shows robustness of the EW RLC UEP schemes over large range of SNRs. It can also been seen that the optimized proposed UEP schemes outperforms the scheme of [20] for the entire range of SNRs. The scheme of [20] uses a nested window structure to realize UEP where the inner window contains IDR, DP A and DP B, while the outer window contains also DP C. The protection

Table 4 Configurations and the window structure for EW RLC68.

Scheme	Window	No. Syms	No. IP pkts	No. TS pkts
EW1	W1	68	68	544
EW2 (RLC-IAB)	W1	49	49	392
	W2	19	19	152
EW2 (RLC-IA)	W1	37	37	296
	W2	31	31	248
EW3	W1	37	37	296
	W2	12	12	96
	W3	19	19	152

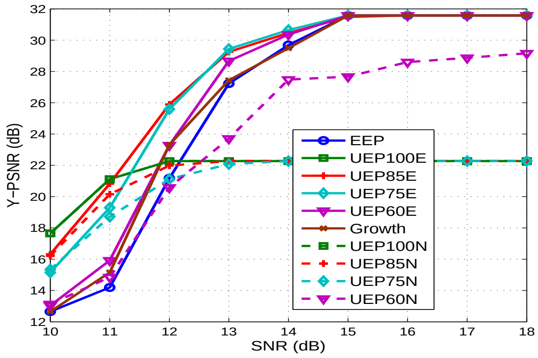

Fig. 7 PSNR vs. SNR in the DVB-H channel for four different NOW and EW UEP schemes and the single-layer EEP scheme at data rate of 383 kbps. The standard QCIF Bus video sequence is used. 'Growth' denotes the scheme of [20].

is equally divided between the two windows. Since Growth codes, used in [20], have similar performance to DF Raptor codes, and hence RLC codes, the advantage of the proposed solution comes from the optimal parameter selection.

Similar behavior can be observed from the next figure that shows results for the CIF Coastguard sequence. UEP85E is the optimal scheme for SNR higher than 11dB. The scheme of [20] loses roughly 1-2dB compared to the UEP85E at low SNRs. Figures 7 and 8 show results obtained with the QCIF Bus and 4CIF City sequence, respectively. It can be seen from the figures that the optimized EW UEP scheme is the most efficient even for different video resolutions.

Fig. 9 shows frame-averaged PSNR vs. data rate for EW1 and EW2 (RLC-IAB) and two different SNRs for the Foreman sequence. Single-layer EEP refers to EW1 scheme in Table 4, where EEPTS12 and EEPTS14 refer to an EEP scheme at SNR=12 and 14dB, respectively. Similarly, UEP60TS14 represents a UEP scheme with 60% selection probability for HPC at SNR=14dB. The results for UEP75TS12/14=UEP(75, 25) and UEP(85, 15) provide significant improvement over the single-layer case. As can be seen from the figure, UEP(85, 15) provides better results overall, whereas UEP(100, 0), in which only W1 is protected, provides good results only at low data rates. This is expected, because in this case, there is not enough bandwidth to transmit both classes, thus, it is better not to waste the bandwidth in trying to recover the LPC

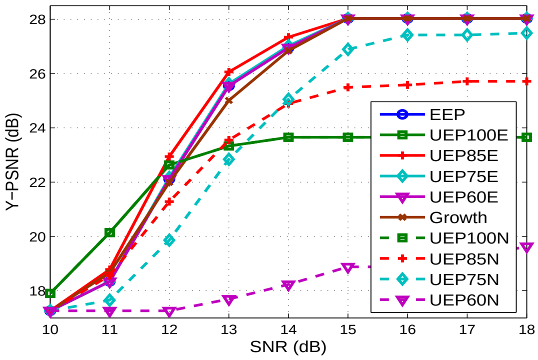

Fig. 8 PSNR vs. SNR in the DVB-H channel for four different NOW and EW UEP schemes and the single-layer EEP scheme at data rate of 383 kbps. The standard 4CIF City video sequence is used. 'Growth' denotes the scheme of [20].

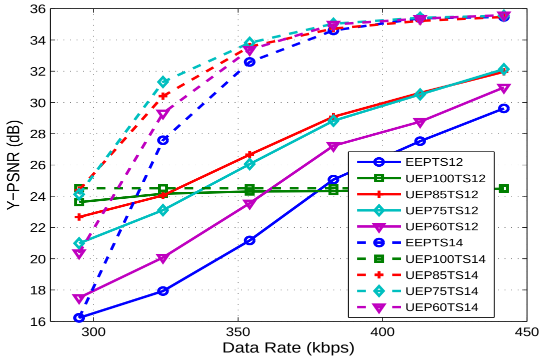

Fig. 9 PSNR vs. data rate for EW2 (RLC-IA) and EW1 (single-layer) at SNR 12dB for the Foreman sequence.

and use all the resources to ensure recovery of the HPC. Note also that UEP(100,0) reaches almost saturation for higher rates since then, HPC is always delivered and the remaining resources are not used. Similar results are obtained for the other three sequences.

The PSNR at various transmission data rates for EW3 and EW1 is shown in Fig. 10. In the EW3 scheme, we put IDR and all DP A's in the first class, all DP B's in the second, and all DP C's in the third. As can be seen from the figure, UEP(58, 20, 22) performs the best overall. In this case, protecting W1 alone, which is smaller as compared to the EW2 case, assumes an almost horizontal alignment, which indicates that it is not affected adversely by decrease in data rate. At low data rates, the performance of UEP schemes is better in general than that of the single-layer scheme. Note that the PSNR performance of EW2 is slightly better than that of the EW3 schemes. However, EW3 provides fine grain control over the transmitted data rate.

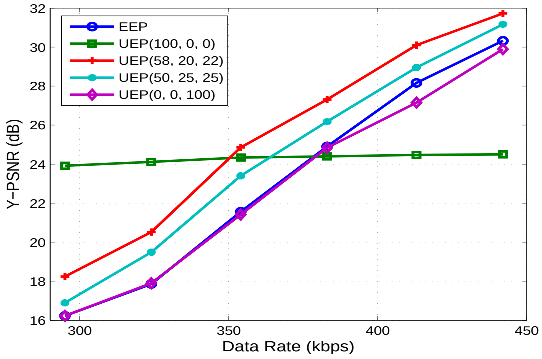

Fig. 10 PSNR vs. data rate for EW3 and EW1 (single-layer) at SNR 12dB for the Foreman sequence.

5.2 Performance Evaluation of EW-RLC based Solution with DVB-T2

Details on DVB-T2 Lite can be found in Annex I [33]. DVB-T2 Lite profile is intended to allow simpler receiver implementations for very low capacity applications such as mobile broadcasting, although it may also be received by conventional stationary receivers. The SNRs for the simulations have been obtained through DVB-T field measurements performed in Turku in August 2011, where each SNR value was measured with 250 ms intervals (corresponding to the maximal DVB-T2 frame duration). The length of the SNR series corresponds to 1040.5 s of reception of a DVB-T2 signal. The SNRs have been used for running DVB-T2 simulations with the following DVB-T2 Lite configurations: 16-QAM and 64-QAM modulations, 16200 bits LDPC codewords with rate 1/2, 8k FFT size, 1/4 guard interval, and TU-6 channel model with perfect channel estimation. By using the SNR's from the DVB-T field measurements as input to DVB-T2 simulations, realistic time variant behaviour of mobile reception was obtained. By using the SNR's from the DVB-T field measurements as input to DVB-T2 simulations, realistic time variant behaviour of mobile reception was obtained.

With these settings, each time interleaver (TI) frame contained 64 or 97 FEC frames for the 16-QAM and 64-QAM simulation scenarios, respectively. The FEC frames have been decoded and a baseband frame error trace has been collected. The baseband frame error traces have then been converted to TS packet error traces, taking into account that the size of each baseband frame is 7032 bits.

Fig. 11 shows the frame-averaged PSNR vs. data rate for the Foreman video sequence and EW1 and EW2 (RLC-IA) for 64-QAM. In case of NOW-RLC except for UEP60N, all schemes suffer from an overprotection of HPC which is detrimental to overall performance. Hence, these schemes get constrained to a PSNR performance of about 24.51 dB, the best achievable with HPC alone. For EW-RLC schemes, the performance is poor at 295 kbps and thereafter all the schemes perform equally well at the increased data rate.

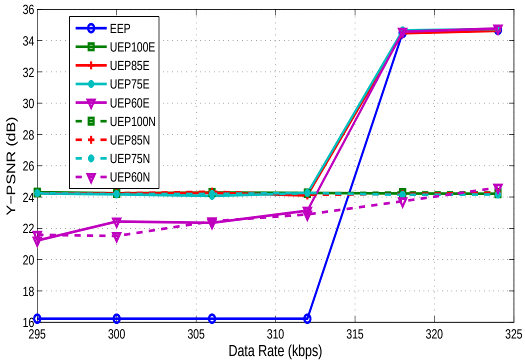

Fig. 11 PSNR comparison between different RTC and RLC configurations: Y-PSNR vs. Data rate in the DVB-T2 channel.

6 Conclusion and Future Work

The study has extensively compared the performance of RLC and Raptor codes by assigning different degree of protection to the data partitions of H.264/AVC video under different packet loss scenarios at different data rates. RLC are seen to perform very close to or somewhat better than Raptor codes. Within RLC configurations, it is seen that the scheme with less number of symbols results in better performance. This is promising result for the use of RLC for multimedia broadcast applications, as the decoding complexity for such configurations is manageable.

The average PSNR of UEP schemes with NOW configuration is not as good as that of single-layer AVC at high data rates. However, it is interesting to note that at lower data rates UEP schemes provide good results with smaller number of decoding failures. The performance of RLC codes using EW-RLC has been analyzed and compared to the case of single-layer coding. EW2 and EW3 both perform up to 5dB better compared to the single-layer case and show robustness to channel fluctuations. The EW3 configuration provides better degree of control to match a particular data rate and erasure channel characteristics, while EW2 overall provides slightly better performance. The performance of EW-RLC schemes is significantly better than corresponding NOW-RLC schemes. The number of symbols per codeword used was 68, which is a very promising result for the use of RLC for multimedia broadcast applications, as the decoding complexity for such configurations is manageable.

References

1. T. Stockhammer, A. Shokrollahi, M. Watson, M. Luby, and T. Gasiba, *Application Layer FEC for Mobile Multimedia Broadcasting (in Handbook of Mobile Broadcasting),* CRC Press (Eds: B. Furht, and S. Ahson), 2008.
2. M. Luby, T. Gasiba, T. Stockhammer, and M. Watson, “Reliable multimedia download delivery in cellular broadcast networks,” *IEEE Trans. Broadcast.,* vol. 53, no. 1, pp. 235-246, Mar. 2007.
3. G. Faria, J. A. Henriksson, E. Stare, and P. Talmola, “DVB-H: Digital broadcast services to handheld devices,” *Proc. of the IEEE,* vol. 94, no. 1, pp. 194-209, Jan. 2006.

4. A. Shokrollahi, “Raptor codes,” *IEEE Trans. Inform. Theory,* vol. 52, no. 6, pp. 2251-2567, June 2006.
5. D. Gomez-Barquero, D. Gozalvez, and N. Cardona, “Application layer FEC for mobile TV delivery in IP datacast over DVB-H systems,” *IEEE Trans. Broadcas.,* vol. 55, no. 2, pp. 396–406, June 2009.
6. D. Lun, M. Medard, R. Koetter, and M. Effros, “On coding for reliable communication over packet networks,” *Physical Communication,* vol. 1, no. 1, pp. 22-30, 2008.
7. J. Jin, B. Li, and T. Kong, “Is random network coding helpful in wimax?,” *in Proc. IEEE INFOCOM 2008,* Phoenix, AZ, April 2008.
8. H. Shojania and B. Li, “Random network coding on the iphone: Fact or fiction?,” *in Proc. NOSSDAV-2009,* Williamsburg, USA, June 2009.
9. P. Vingelmann, F.H.P. Fitzek, M.V. Pedersen, J. Heide, and H. Charaf, “Synchronized multimedia streaming on the iPhone platform with network coding,” *in Proc. CCNC IEEE Consumer Commun. and Networking Conf.,* Las Vegas, USA, Jan. 2011.
10. L. Benacem and S.D. Blostein, “Raptor-network coding enabled strategies for energy saving in DVB-H multimedia communications,” *in Proc. 1st Int. Conf. Green Circuits and Systems,* pp. 527–532, June 2010.
11. S. Teerapittayanon, K.M. Fouli, M. Medard, M.-J. Montpetit, X. Shi, I. Seskar, and A. Gosain, ”Network coding as a WiMAX link reliability mechanism”, in *Proc. MACOM-2012 Int. Workshop on Multiple Access Communications,* Dublin, Ireland, Nov. 2012.
12. S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, ”XORs in the air: Practical wireless network coding,” in *Proc. ACM SIGCOMM-2006 Computer Communication Review,*, Pisa, Italy, Sept. 2006.
13. T. Wiegand, G. Sullivan, G. Bjøntegaard, and A. Luthra, “Overview of the H.264/AVC Video Coding Standard,” *IEEE Trans. Circuits and Sys. Video Techn.*, vol. 13, no. 7, Jul. 2003.
14. T. Schierl, T. Stockhammer, and T. Wiegand, “Mobile video transmission using Scalable Video Coding,” *IEEE Trans. Circuits and Sys. Video Techn.,* vol. 17, pp. 1204-1217, Sept. 2007.
15. H. Schwarz, D. Marpe, and T.Wiegand, “Overview of the scalable video coding extension of the H.264/AVC standard,” *IEEE Trans. Circ. and Syst. for Video Tech.,* vol. 17, no. 9, pp. 1103-1120, 2007.
16. S. Wenger, “H.264/AVC over IP,” *IEEE Trans. Circuits and Syst. for Video Technol.,* vol. 13, no. 7, pp. 645-655, July 2003.
17. T. Stockhammer and M. Bystrom, “H.264/AVC data partitioning for mobile video communication,” *in Proc. ICIP-2004 Intl. Conf. Image Proc.*, pp. 545-548, Singapore, Oct. 2004.
18. B. Barmada, M.M. Ghandi, E.V. Jones, and M. Ghanbari, “Prioritized transmission of data partitioned H.264 video with hierarchical QAM,” *IEEE Signal Processing Letters,* vol. 12. no. 8, pp. 577-580, Aug. 2005.
19. L. Al-Jobouri, M. Fleury, and M. Ghanbari, “Error-resilient IPTV for an IEEE 802.16e channel,” *Wireless Engineering and Technology,* 2:2, pp. 70-79, April 2011.
20. R. Razavi, M. Fleury, M. Altaf, H. Sammak, and M. Ghanbari, “H.264 video streaming with data-partitioning and Growth codes,” *in Proc. ICIP-2009 Intl. Conf. Image Proc.,* Cairo, Egypt, Oct. 2009.
21. A. Bouabdallah and J. Lacan, “Dependency-aware unequal erasure protection codes,” *J. Zheijiang Univ. Science A,* vol. 7, pp. 27-33, Apr. 2006.
22. T. Connie, P. Nasiopoulos, Y.P. Yasser, and V.C.M. Leung, “SCTP-transmission of data-partitioned H.264 video,” *in Proc. 4th ACM Workshop Wireless Multimedia Networking and Performance,* pp. 32–36, 2008.
23. D. Vukobratović, V. Stanković, D. Sejdinović, L. Stanković, and Z. Xiong, “Expanding window fountain codes for scalable video multicast,” *IEEE Trans. Multimedia,* vol. 11, no. 6, pp. 1094-1104, 2009.
24. 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, pp. 551-562, Jun 2011.
25. D. Vukobratović and V. Stanković, “Unequal error protection random linear coding for erasure channels,” *IEEE Trans. Communications,* vol. 60, pp. 1243-1252, May 2012..
26. L. Vangelista et al., “Key technologies for next-generation terrestrial digital television standard DVB-T2”, *IEEE Commun. Magazine,* vol 47. no 10, pp. 146-153, Oct. 2009.
27. Y. Dhondt, S. Mys, K. Vermeirsch, and R. Van de Walle, “Constrained inter prediction: Removing dependencies between different data partitions,” *in Proc. Advanced Concepts for Intelligent Visual Systems,* pp. 720–731, 2007.
28. “ETSI TS 126 346, universal mobile telecommunications system (umts); multimedia broadcast/multicast service (mbms); protocols and codecs,” ETSI Tech. Spec., 2005.
29. K. Nybom, S. Grönroos, and J. Björkqvist, “Expanding window fountain coded scalable video in broadcasting,” *in Proc. ICME-2010 IEEE Int. Conf. Multimedia and Expo,* Singapore, July 2010.

30. J. Poikonen and J. Paavola, “Error models for the transport stream packet channel in the DVB-H link layer”, *in Proc. ICC-2006 IEEE Int. Conf. Commun.*, pp. 1861–1866, Istanbul, Turkey, 2006.
31. H.264/AVC Reference Software, http:/iphome.hhi.de.
32. H. Shojania and B. Li, “Parallelized progressive network coding with hardware acceleration,” *in Proc. IWQoS-2007 the 15th Intl. Workshop Quality of Service,* Chicago, IL, July 2007.
33. ETSI EN 302 755, “Digital Video Broadcasting (DVB); Frame structure channel coding and modulation for a second generation digital terrestrial television broadcasting system (DVB-T2)”, *V1.3.1, DRAFT,* Oct. 2011.
34. G. Angelopoulos, M. Medard, and A.P. Chandrakasan, ”Energy-aware hardware implementation of network coding,” *Lecture Notes in Computer Science,* vol. 6827, pp. 137–144, 2011.