Adaptive Rateless Coding for Data-Partitioned Video Streaming over a Broadband Wireless Channel

Laith Al-Jobouri, Martin Fleury, and Mohammed Ghanbari
School of Computer Science and Electronic Engineering
University of Essex, Colchester, United Kingdom
{lamoha, fleum, ghan}@essex.ac.uk

***Abstract*—Fixed-rate, rateless channel coding is attractive for its linear decoder computational complexity. Though fixed rate coding is appropriate for broadcast systems, this paper proposes adaptive coding for unicast, data-partitioned compressed video streams. The paper finds that the main gain from such a scheme is an increase in goodput and wireless channel utilization, which is important for a real-time service. In fact, in an ideal scheme no retransmissions are needed to repair packets corrupted by channel errors. Video quality is also potentially enhanced by the differential packet drop rates at buffers that result in less important data-partition bearing packets dropped more frequently.**

Keywords-broadband wireless, data-partitioning, rateless channel coding, source coding, video streaming

I. Introduction

Rateless channel coding [1] allows the code rate to be adaptively changed according to channel conditions, avoiding the thresholding effect associated with fixed-rate codes such as Reed-Solomon (RS). However, the linear decode complexity of one variant of rateless coding, Raptor coding [2], has made it attractive for its efficiency alone. For broadcast systems such as 3GPP's Multimedia Broadcast Multicast System (MBMS), [3], as channel conditions may vary for each receiver, the possibility of adapting the rate is not exploited. However, for unicast video-on-demand and time-shifted TV streaming it is possible to adaptively vary the rate according to measured channel conditions at the sender. These services are a commercially-attractive facility offered by IPTV [4] as they add value to a basic broadcast service.

We have found that data-partitioned, compressed video allows smaller packets to be associated with the more important data for video picture reconstruction. In a broadband wireless system, specifically IEEE 802.11e (mobile WiMAX) [5], the packet size critically affects the packet drop rate at a base station's (BS) transmission buffer and also increases the risk of packet corruption while the packet traverses the wireless channel. However, the additional protection offered by application-layer Forward Error Correction (FEC) risks increasing the packet drop rate through buffer overflow at the BS, which FEC is unable to protect against. Therefore, if it is possible to reduce the packet size by generating less redundant data when channel conditions permit, then the packet drop rate will reduce. Equally the reduction in total packet size will also reduce the risk of packet corruption during wireless transmission. In this paper, by assuming perfect channel knowledge we show that such a scheme has the potential to bring important gains.

The WiMAX standard already specifies [6] that a subscriber station (SS) should provide channel measurements that can form a basis for channel quality estimates. These are either Received Signal Strength Indicators or may be Carrier-to-Noise-and-Interference Ratio (CINR) measurements made over modulated carrier preambles. In respect to corruption of the packet contents over the channel, if sufficient data has arrived then in the proposed scheme a single Automatic Repeat ReQuest (ARQ) is permitted (to avoid excessive delay to the video-rate application), again allowing the rateless features of the channel code to be exploited. There is, however, a danger that if the number of packets corrupted in transmission increases then the overall delay will increase significantly. Thus, though a rateless coding scheme with single ARQ acts to drastically reduce the error rate if the redundancy percentage is fixed, it can still introduce excessive delay. The adaptive scheme works by only introducing sufficient protection to match the risk of error.

For simplicity of implementation, unequal error protection through additional FEC of the partitioned data is not applied, as the proposed scheme relies on the unequal packet sizes to give unequal protection. Interestingly, in data-partitioning as originally employed (see Section II), it was the placement of the more important data at the start of a packet that afforded extra protection and not the degree of protection given this data.

Data partitioning is applied through an H.264/Advanced Video Coding (AVC) [7] encoder, which increases the compression ratio of Digital Video Broadcasting (DVB)'s MPEG-2 codec by as much as 50%. This is why his codec is of interest to IPTV [4] over bandwidth-limited wireless channels. H.264/AVC, as part of its network-friendly approach, introduces Network Abstraction Layer units (NALUs) as a per-slice container for transmission. Data partitioning of NALUs into partition A, B, and C types [8] of decreasing importance for decoder reconstruction purposes can result in Real Time Protocol (RTP) packets of decreasing size. This is achieved by setting the variable-bit-rate video's quantization parameter (QP) in such a way that lower-priority texture data, which can be compensated for more easily at the decoder, occupies a larger proportion of a video frame's compressed data. Consequently, when partition C data are packetized in a WiMAX MAC Service Data Unit (MSDU) within a MAC Protocol Data Unit (MPDU), the longer packet is more likely to suffer error than MSDUs bearing data from other partitions. However, the decoder can apply motion copy error concealment to reconstruct missing partition C data by means of the motion vectors still available in partition A. Video quality in the proposed scheme is additionally improved through the inclusion of intra-refresh macroblocks (MBs) within predictively-coded pictures [9]. Because intra-refresh MBs are not predictively coded, their inclusion helps to prevent temporal error propagation. The paper now establishes the essential concepts for an understanding of the results.

II. CONTEXT

A. *Data-partitioning*

The H.264/AVC codec conceptually separates the Video Coding Layer (VCL) from the NAL. The VCL specifies the core compression features, while the NAL supports delivery over various types of network. This feature of the standard facilitates easier packetization and improved video delivery. In a communication channel the quality of service is affected by the two parameters of bandwidth and the probability of error. Therefore, as well as video compression efficiency, which is provided for through the VCL layer, adaptation to communication channels should be carefully considered. The concept of the NAL, together with the error resilience features in H.264/AVC, allows communication over a variety of different channels.

The NAL facilitates the delivery of the H.264/AVC VCL data to the underlying transport layers such as RTP/IP, H.32X and MPEG-2 systems. Each NAL unit could be considered as a packet that contains an integer number of bytes including a header and a payload. The header specifies the NALU type and the payload contains the related data. Table I is a summarized list of different NALU types. NALUs 6 to 12 are non-VCL units containing additional information such as parameter sets and supplemental information.

In H.264/AVC, when data-partitioning is enabled, every slice is divided into three separate partitions and each partition is located in either of type 2 to type-4 NALUs, as listed in Table I. NALU of type 2, also known as partition A, comprises the most important information of the compressed video bit-stream of P- and B-pictures (when used), including the MB addresses, motion vectors and essential headers. If any MBs in these pictures are intra-coded, their Discrete Cosine Transform (DCT) coefficients are packed into the type-3 NALU, also known as partition B. Type 4 NAL, also known as partition C, carries the DCT coefficients of the motion-compensated inter-picture coded MBs. It is worth noting that, as in I-pictures all MBs are spatially encoded, type 5 NALUs are very long. On the other hand A and B partitions of data-partitioned P- and B-pictures are much smaller but their C-type partition may be very long.

Fig. 1 is a comparison between the relative sizes of the partitions according to QP for the tested video clip. In H.264/AVC the QP range is from 0 to 51, with a low QP representing high quality. The *Paris* sequence is a studio scene with two head and shoulders images of presenters and moderate motion. The background is of high spatial complexity. Paris was Variable Bit-Rate (VBR) encoded at Common Intermediate Format (CIF) (352 × 288 pixel/frame), with a Group of Picture (GOP) structure of IPPP….. at 30 frame/s. We reserve future work to elaboration of a simple 'rule of thumb' to determine a QP that best benefits the data-partitioning scheme. This will depend on the proportion of intra-refresh MBs, which has the effect of changing the relative size of the B-partition's contribution.

The relatively small size of A- and B-partitions is a potential advantage at low QPs but this comes at a cost of a high bitrate. Conversely, at the low quality end of the QP range, partition-A NALUs become relatively vulnerable to packet loss by virtue of their relatively increased size.

TABLE I. NAL UNIT TYPES

NAL unit type	Class	Content of NALU
0	-	Unspecified
1	VCL	Coded slice
2	VCL	Coded slice partition A
3	VCL	Coded slice partition B
4	VCL	Coded slice partition C
5	VCL	Coded slice of an IDR picture
6-12	Non-VCL	Suppl. info., Parameter sets, etc.
13-23	-	Reserved
24-31	-	Unspecified

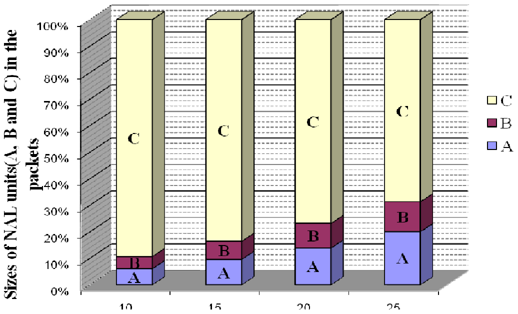

Figure 1.   Relative sizes of data partitions according to quantization parameter (QP) for the *Paris* video sequence, with 5% intra-refresh MBs.

B. *Rateless coding*

A fixed-rate (n, k) RS channel code over an alphabet of size $q = 2^L$ has the property that if *any* k out of the n symbols transmitted are received successfully then the original k symbols can be decoded. However, for RS coding not only must n, k, and q be small but also the computational complexity of the decoder is of order $n(n - k) \log_2 n$. The error rate must also be estimated in advance.

In contrast, the class of Fountain codes [1] allows a continual stream of additional symbols to be generated in the event that the original symbols could not be decoded. It is the ability to easily generate new symbols that makes Fountain codes rateless. Decoding will succeed with small probability of failure if any of k $(1 + \varepsilon)$ symbols are successfully received. In its simplest form, the symbols are combined in an exclusive OR (XOR) operation according to the order specified by a random low density generator matrix and in this case, the probability of decoder failure is $\partial = 2^{-k\varepsilon}$, which for large k approaches the Shannon limit. The random sequence must be known to the receiver but this is easily achieved through knowledge of the sequence seed.

Luby Transform (LT) codes reduce the complexity of decoding a simple Fountain code (which is of order k^3) by means of an iterative decoding procedure, provided that the column entries of the generator matrix are selected from a robust Soliton distribution. In the LT generator matrix case, the expected number of degree one combinations (no XORing of symbols) is $S = c \log_e(k/\partial)\sqrt{k}$, for small constant c. Setting $\varepsilon = 2 \log_e(S/\partial)\ S$ ensures that by sending $k(1 + \varepsilon)$ symbols these are decoded with probability $(1 - \partial)$ and decoding complexity of order $k \log_e k$.

If packets are pre-encoded with an inner code, a weakened LT transform can be applied to the symbols and their redundant symbols. The advantage of this Raptor code [2] is a decoding complexity that is linear in k. A systematic Raptor code is arrived at [2] by first applying the inverse of the inner code to the first k symbols before the outer pre-coding step.

III. SIMULATION MODEL

A. *Rateless coding evaluation*

In order to model Raptor coding, we employed the following statistical model [10]:

$$P_f(m,k) = 1 \quad \text{if } m < k,$$
$$= 0.85 \times 0.567^{m-k} \text{ if } m \geq k \qquad (1)$$

where $P_f(m,k)$ is the failure probability of the code with k source symbols if m symbols have been received. The authors of [10] remark and show that for $k > 200$ the model almost perfectly models the performance of the code.

In the experiments reported in this paper, the percentage redundancy for the Raptor code was set by default to 10% similarly to the usage in [11] for video streaming. However, the redundancy level was then varied to 5% and 15%, as a check on whether the setting was appropriate. The symbol size was set to bytes within a packet. Clearly, if instead 200 packets are accumulated before the rateless decoder can be applied (or at least equation (1) is relevant) there is a penalty in start-up delay for the video streaming and a cost in providing sufficient buffering at the mobile stations.

A corrupt packet can be detected by the Cyclic Redundancy Check (CRC) that is an optional part of the MPDU (WiMAX packet), refer to Fig. 2. Though this CRC also applies to the 4-byte MAC header, it does indicate the likelihood that a packet's payload is corrupt. Then, through measurement of channel conditions, an estimate of the number of symbols successfully received is made, giving a value *m'*. This implies from (1) that if less than k symbols (bytes) in the payload are successfully received then $k-m'+e$, $e > 0$ redundant bytes can be sent to reduce the risk of failure to below 50%. In tests, e = 4, resulting in a risk of 1.5%, because of the exponential decay of the risk evident from equation (1).

To reduce latency, the number of retransmissions, after an ARQ over the uplink, was limited to one. The effect of this decision can be judged by the results in Section IV. Fig. 3 shows how ARQ triggered retransmissions work. In the Figure, packet X is corrupted to such an extent that it cannot be reconstructed. Therefore, in packet X+1 some extra redundant data is included up to the level that its failure is no longer certain. If the extra redundant data is insufficient to reconstruct the original packet, the packet is simply dropped. Otherwise, of course, it is passed to the H.264/AVC decoder.

B. *Channel model*

To establish the behavior of rateless coding under WiMAX the well-known ns-2 simulator augmented with a module from the Change Gung University, Taiwan that has proved an effective way of modeling IEEE 802.16e's behavior. We also introduced a two-state Gilbert-Elliott channel model [13] in the physical layer of the simulation to simulate the channel model for WiMAX. The *PGG* (probability of remaining in a good state) was set to 0.95, *PBB (*probability of remaining in a bad state) = 0.96, *PG* (probability of byte corruption in a bad state) = 0.02 and *PB* (probability of byte corruption in a bad state) = 0.165 for the Gilbert-Elliott parameters. Effectively, this model emulates fast fading between good and bad conditions. A future expansion is to include four states to include the effect of slow fading as well.

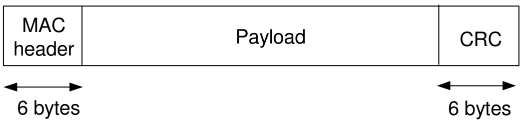

Figure 2. General format of a MAC PDU with optional CRC.

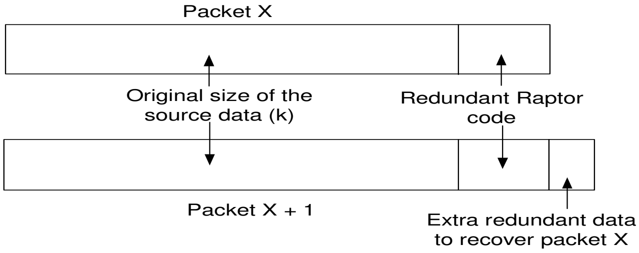
Packet X
Original size of the source data (k)
Redundant Raptor code
Packet X + 1
Extra redundant data to recover packet X

Figure 3. Division of payload data in a packet (MPDU) between source data, original redundant data and piggybacked data for a previous errored packet.

C. *WiMAX configuration*

The physical layer (PHY) settings selected for WiMAX simulation are given in Table II. For comparison purposes, the antenna is modeled as a half-wavelength dipole. The Time Division Duplex (TDD) frame length was varied in experiments, because it has an effect on the service rate at the BS. Video was transmitted over the downlink with UDP transport. In order to introduce sources of traffic congestion, an always available FTP source was introduced with TCP transport to the SS. Likewise a CBR source with packet size of 1000 B and inter-packet gap of 0.03 s was downloaded to the SS. While the CBR and FTP occupy the non-rtPS (non-real-time polling service) queue, rather than the rtPS queue, they still contribute to packet drops in the rtPS queue for the video, if the packet rtPS buffer is already full or nearly full, while the nrtPS queue is being serviced. We are aware that this congestion configuration is just one of many and future work should investigate this dimension of the problem.

D. *Video configuration*

The *Paris* sequence with 1065 frames with the same configuration of Section II.A was employed for the WiMAX downlink tests. As a GOP structure of IPPP.... was employed, it is necessary to protect against error propagation in the event of inter-coded P-picture slices being lost. To ensure higher quality video, 5% intra-refresh MBs (randomly placed) were included in each P-picture (apart for the first I-picture) to act as anchor points in the event of slice loss. At the decoder, motion-copy error concealment was set, allowing the motion vectors contained in partition A packets to indicate suitable replacement MBs within the last correctly received slice. The JM 14.2 version of the H.264/AVC codec software was employed with the Evalvid environment [13], which was used to reconstruct sequences according to packet losses reported by the simulator.

III. EVALUATION

Two types of erroneous packets were considered: 1) packet drops at the BS sender buffer and 2) corrupted packets that were received but affected by Gilbert-Elliott channel noise to the extent that they could not be immediately reconstructed without an ARQ triggered retransmission. Notice that if the retransmission of additional redundant data still fails to allow the packet to be reconstructed then the packet is simply dropped. The Raptor code equation (1) was applied to decide if a packet could be recovered, given the number of bytes that were declared to be in error. All data points are the mean of at least ten simulation runs.

TABLE II. IEEE 802.16 PARAMETER SETTINGS

Parameter	*Value*
PHY	OFDM
Frequency band	5 GHz
Bandwidth capacity	10 MHz
Duplexing mode	TDD
Frame length	5 – 20 ms
Max. packet length	1024 B
Raw data rate	10.67 Mbps
IFFT size	1024
Modulation	16-QAM 1/2
Guard band ratio	1/8
DL/UL ratio	3:1
MS transmit power	245 mW
BS transmit power	20 W
Approx. range to SS	1 km
Antenna type	Omni-directional
Antenna gains	0 dBD
MS antenna height	1.2 m
BS antenna height	30 m

OFDMA = Orthogonal Frequency Division Multiple Access, QAM = Quadrature Amplitude Modulation, TDD = Time Division Duplex

In the adaptive scheme, the probability of channel loss (*PL*) in the Gilbert-Elliott model served to predict the additional data that was piggybacked onto the next packet to be transmitted (*extra redundant data to recover packet X* in Fig. 3). If PGB and PBG are the probabilities of going from good to bad state and from going from bad to good state respectively, then

$$\pi_G = PBG/(PBG + PGB) \quad (2)$$

$$\pi_B = PGB/(PBG + PGB) \quad (3)$$

are the steady state probabilities of being in the good and bad states. Consequently, the mean probability of packet loss is given by

$$PL_{mean} = \mathrm{PG}.\ \pi_G + \mathrm{PB}.\ \pi_B \quad (4)$$

that is the mean of a Uniform distribution in this case.

Empirical investigation showed that this was insufficient provision unless approximately 10% extra data were added over and above that allowed for by a direct use of the *PL*. This is a consequence of the relationship in (1). Moreover, this adjustment varies according to channel conditions,

TABLE III. STREAMING PARIS WITH 10% REDUNDANT DATA

	Dropped packets (%)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	14.0	13.8	13.8
25	2.8	0	0
30	0	0	0
35	0	0	0
	Corrupted packets (%)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	7.8	6.7	8.1
25	9.2	8.2	10.0
30	8.1	8.6	7.1
35	9.5	7.2	9.4
	PSNR (dB)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	16.84	16.86	16.86
25	30.98	42.49	42.49
30	38.44	38.44	38.44
35	33.54	33.54	33.54
	Corrupted packet end-to-end delay (s)		
QP	5 ms TDD	10 ms TDD	10 ms TDD
20	0.0243	0.0279	0.0356
25	0.0203	0.0248	0.0318
30	0.0183	0.0236	0.0281
35	0.0171	0.0223	0.0256
	Packet end-to-end delay (s)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	0.0127	0.0161	0.0239
25	0.0099	0.0148	0.0219
30	0.0084	0.0133	0.0198
35	0.0071	0.0121	0.0180

TABLE IV. STREAMING PARIS WITH 5% REDUNDANT DATA

	Dropped packets (%)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	11.3	11.3	11.9
25	0	0	0
30	0	0	0
35	0	0	0
	Corrupted packets (%)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	39.4	40.8	39.0
25	46.2	46.3	46.4
30	43.3	44.0	45.4
35	44.6	43.4	42.1
	PSNR (dB)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	17.87	18.17	17.26
25	42.49	42.49	42.49
30	38.44	38.44	38.44
35	33.54	33.54	33.54
	Corrupted packet end-to-end delay (s)		
QP	5 ms TDD	10 ms TDD	10 ms TDD
20	0.0233	0.0272	0.0350
25	0.0200	0.0245	0.0320
30	0.0183	0.0231	0.0298
35	0.0171	0.0220	0.0281
	Packet end-to-end delay (s)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	0.0127	0.0160	0.0237
25	0.0100	0.0147	0.0217
30	0.0083	0.0132	0.0197
35	0.0071	0.0120	0.0180

though the change monotonically increases according to *PL*. On-going investigations seek a relationship between *PL* and the adjustment factor, *A*. If the original packet length is *L*, then the extra redundant data is given simply by

$$E = L \times PL \times A \qquad (5)$$

when A = 1.11in the evaluation experiments.

Initial experiments investigated how the size of the redundant provision in a fixed scheme affected the performance of the scheme. In Tables III and IV, the performance metrics are presented according to QP and WiMAX TDD frame size. Recall that increasing the QP increases the compression ratio, decreases the quality, and decreases the NALU size at the sender. Obviously if packets are lost in transmission, whatever the QP the video quality will deteriorate at the receiver after decoding. In general, increasing the WiMAX frame size allows more packets to be serviced as each of the SSs mentioned in Section III.C are serviced. However, an implementation may set the TDD frame size according to other factors than the impact upon video streaming, as other traffic has different requirements.

In these Tables, as in the event of transmission errors, data is retransmitted by a single ARQ then all corrupted packets suffer end-to-end delay. Therefore, the total time to transfer the video is affected by the extent that packets are corrupted. As we were interested in low delay video streaming to mobile devices a small buffer size of five packets was set.

From Tables III and IV, a number of observations can be made. The decrease in video quality at QP = 20 is largely due to the number of packets dropped at the buffers. This is a result of the larger packet sizes at QP = 20 leading to an increase in propagation time and hence greater queuing time while transmission is completed. There is a large increase in the percentage of corrupted packets when the Raptor code protection is reduced from 10% to 5%. Though this does not affect the video quality, because of the retransmission of extra redundant data, it does lead to a significant increase in overall transmission time for the video sequence. In fact, when packets are not dropped at the buffer, the video quality essentially depends on the QP value. If packets are dropped the quality falls well below an acceptable level for mobile video (about 25 dB) and is dependent on the pattern of packet drops. When corrupted packets are retransmitted, on average their overall end-to-end delay for the extra trip remains about the same between the two fixed redundancy schemes. The overall mean packet end-to-end delay is largely determined by packet size, which reflects propagation time.

Table V illustrates the results for the proposed adaptive scheme based on equation (2). Though packet drops increase compared to the 5% fixed redundancy the numbers are reduced compared to the 10% fixed rate scheme.

TABLE V. MEAN PERFORMANCE FOR STREAMING PARIS WITH ADAPTIVE RATELESS CODING

	Dropped packets (%)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	11.8	11.9	11.5
25	0	0	0
30	0	0	0
35	0	0	0
	Corrupted packets (%)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	0	0	0
25	0	0	0
30	0	0	0
35	0	0	0
	PSNR (dB)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	15.72	16.61	18.24
25	42.49	42.49	42.49
30	38.44	38.44	38.44
35	33.54	33.54	33.54
	Corrupted packet end-to-end delay (s)		
QP	5 ms TDD	10 ms TDD	10 ms TDD
20	0	0	0
25	0	0	0
30	0	0	0
35	0	0	0
	Packet end-to-end delay (s)		
QP	5 ms TDD	10 ms TDD	20 ms TDD
20	0.0122	0.0160	0.0238
25	0.0100	0.0146	0.0217
30	0.0083	0.0131	0.0196
35	0.0071	0.0120	0.0179

However, the main gain is that corrupted packets are completely eliminated. This should not be surprising as the amount of redundancy is molded to the channel conditions. The consequence is that delay of congested packets is also removed thus significantly increasing the goodput (rate of delivering good packets) and reducing the streaming delay. Though the reduction in packet drops did not have a significant effect in this scenario, we postulate that in other scenarios, differences in packet drop rates will prove to be significant.

Another interesting observation for the adaptive and other schemes was that no partition A packets were amongst those dropped. For the fixed 10% scheme, with 5 ms TDD frame size and QP = 20, 20 partition B and 85 partition C packets were dropped. Under the same conditions, for 5% redundancy, 21 partition B and 67 partition C packets were dropped, and for the adaptive scheme, 22 partition B and 70 partition C packets were dropped. Therefore, the packet drop pattern across the data-partitioning scheme reflects the relative packet sizes of the different partitions. Had no intra-refresh MBs been included then more partition A bearing packets may have been dropped, because in this case the partition B will be smaller than the partition A bearing packets. With an increase in partition A packets video quality would decline as motion copy error concealment would be affected. At the same time, though it is important to include intra-refresh MBs to reduce temporal error propagation, these MBs are not efficient to code compared to inter-coded MBs. Thus they tend to disproportionately contribute to the data rate relative to the area of the image they occupy.

IV. CONCLUSION

This paper has proposed that true rateless coding (rather than fixed rate rateless coding) can be exploited for video-on-demand varieties of video streaming over broadband wireless. The benefit is a drastic reduction in corrupted packets, i.e. packets that cannot be reconstructed at the receiver without further transmission of redundant data. Consequently, delay from retransmissions in tests with perfect channel knowledge was eliminated. An interesting finding was that when packet drops occurred the smaller partition A bearing packets escaped unscathed. In the scenarios tested the levels of packet loss were too high for this fact to be relevant. However, in situations in which less packet losses occur it is expected that it will be practically beneficial. Therefore, future work consists of further tests of the scheme across a wider range of scenarios.

REFERENCES

[1] D. J. C. MacKay, "Fountain codes," *IEE Proc.: Communications*, vol. 152, no. 6, pp. 1062–1068, 2005.

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

[3] J. Afzal, T. Stockhammer, T. Gasiba, and W. Xu, "Video streaming over MBMS: A system design approach," *J. of Multimedia*, vol. 1, no. 5, pp. 25-35, 2006.

[4] J. She, F. Hou, P.-H. Ho, and L.-L. Xie, "IPTV over WiMAX: Key success factors, challenges, and solutions," *IEEE Communs. Mag.*, vol. 45, no. 8, pp. 87-93, 2007.

[5] IEEE, 802.16e-2005, IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, 2005.

[6] L. Nuaymi, *WiMAX: Technology for Broadband Wireless Access*, J. Wiley & Sons Ltd, Chichester, UK, 2007

[7] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560-576, July 2003.

[8] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645-655, 2003.

[9] M. N. Hannuksela, Y.-K. Wang, and M. Gabbouj, "Random access using isolated areas," in IEEE Int'l Conf. on Image Processong, 2003, pp. 841-844 .

[10] M. Luby, T. Gasiba, T. Stockhammer, and M. Watson, "Reliable multimedia download delivery in cellular broadcast networks," *IEEE Trans. Broadcasting*, vol. 53, no. 1, pp. 235-246, 2007.

[11] S. Ahmad, R. Hamzaoui, M. Al-Akaidi, "Robust live unicast video streaming with rateless codes", in *Int'l PacketVideo Workshop*, Nov. 2007.

[12] G. Haßlinger and O. Hohlfeld, "The Gilbert-Elliott model for packet loss in real time services on the Internet," in *14th GI/ITG Conf. on Measurement, Modelling, and Evaluation of Computer and Commun. Systs.,* 2008, pp. 269-283.

[13] J. Klaue, B. Rathke, and A.Wolisz, "EvalVid - A framework for video transmission and quality evaluation," in *Int'l Conf. on Modeling Techniques and Tools for Computer Performance*, 2003, pp. 255-272.