

# Delay-Aware Interleaving and Forward-Error Correction for Video over Wireless: A Bluetooth Case Study

Rouzbeh Razavi  
University of Essex  
Wivenhoe Park  
Colchester, U.K.  
+44 1206 872442

Martin Fleury  
University of Essex  
Wivenhoe Park  
Colchester, U.K.  
+44 1206 872817

Mohammed Ghanbari  
University of Essex  
Wivenhoe Park  
Colchester, U.K.  
+44 1206 872434

Mahta Sadeghzadeh  
UCLA  
Los Angeles  
California, U.S.A  
+1 949 305 1795

rrazv@essex.ac.uk fleum@essex.ac.uk ghan@essex.ac.uk mahta@bolproxy.college.ucla.edu

## ABSTRACT

Packet-level interleaving combined with FEC has tended to be disregarded for video streaming over wireless, because of the delay introduced by the need to assemble the interleaved packets. However, packet level interleaving has the ability to cope with lengthy bursts of error on a wireless channel. This paper proposes adapting the degree of packet interleaving according to: the display deadlines of packets; transmit buffer occupation; and anticipated video input to the wireless channel. Conservative adaptation is applied to avoid missed deadlines and the picture type importance is taken into account. A detailed Bluetooth case study illustrates the value of the techniques and as a by-product the paper corrects some erroneous results of others for Bluetooth version 2's Enhanced Data Rate.

## Categories and Subject Descriptors

C.2.1 [Computer Communication Networks]:  
Network Architecture and Design, Wireless Communication

## General Terms

Performance

## Keywords

Delay-aware, FEC, interleaving, video streaming.

## 1. INTRODUCTION

Wireless channels in personal area and local area networks are prone to radio frequency noise, together with interference from multipath and short-term fading. This noise and interference often manifests itself as error bursts correlated in time. Unfortunately, lengthy bursts of errors are now accepted [24][16][12] as being more harmful to received video quality than randomly occurring errors. To combat noise and interference, Forward Error Correction (FEC) may be built-in at the data-link layer. For example, in IEEE 802.15.1, Bluetooth [7], Data Medium packets transmitted at the basic rate (1 Mbps gross air rate) include payload error coding rates of 1/3 and 2/3. However, stronger

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*ACM Int. Workshop on Wireless Multimedia Networking and Performance*, October 22–26, 2007, Chania, Crete, Greece.

Copyright 2007 ACM 1-58113-000-0/00/0004...\$5.00.

forms of FEC such as Turbo coding may be needed for Bluetooth [29] at SNR below approximately 10 dB, and the Enhanced Data Rate (EDR) modes of Bluetooth v. 2.0 do not include FEC. At the most, Bluetooth's expurgated (15, 10) Hamming code<sup>1</sup> can only cope with burst sizes of two, depending on decoder [11]. However, the ability of block codes to cope with burst errors is considerably improved if some form of data interleaving is implemented, because interleaving tends to remove error correlations, with the extent of interleaving being directly related to the maximum burst length that can be resisted.

Unfortunately, it is commonly considered [2] that for video data additional packet-level interleaving with accompanying packet-level FEC [21][9] will cause display deadlines to be missed. Thus, in solving one form of packet loss through error bursts, there is a danger of packet loss through missed deadlines. As a result, interleaving may be confined [3] to symbol-level interleaving within a packet. However, symbol level interleaving can only increase burst length resistance to the width of the packet reordering matrix (the distance between formerly adjacent symbols), while packet level interleaving further scales the immunity proportionally to the number of packets interleaved. Packet level interleaving is likely to be helpful in low SNR channel conditions and may be used with block coding without the complexity of Turbo or Low Density Parity Check (LDPC) coding [18].

In fact for video, the display deadlines of video packets are knowable from packet headers (without decoding the payload) and may be provided to the data-link layer through cross-layer interaction. It is the central contribution of this paper, that, given the knowledge of packet display deadlines and delays from other factors it is possible to adapt the extent of interleaving (assuming block coding) to avoid missed deadlines. Other causes of delay are: anticipated delay arising from already buffered (and interleaved) packets; and cross traffic, if it is serviced before the video stream packets are serviced. The latter occurs in a

---

<sup>1</sup>There is an unfortunate mistake, pointed out in [9], in the Bluetooth specification which incorrectly identifies the Bluetooth generating function as a shortened form, leading to the reporting of sub-optimal error decoding results, for example in [3], amongst many others. This paper reports corrected results from this and another apparent misunderstanding (refer to Section 3.1).

Bluetooth piconet because all slave-to-slave traffic must pass via the master node.

As this process relies on an estimation of delay caused by cross traffic, a conservative, robust algorithm [10] is necessary. Given also that video picture types are not all of equal importance, the estimation precautions in our work are weighted as to whether the packets originate from I-, P-, or B-pictures. Within a Group of Pictures (GOP), anchor pictures, I-pictures, are intra-coded, and influence the decoding of all other pictures in the GOP, typically 12 or 15 display frames, P-pictures contribute to the decoding of later pictures, whereas B-pictures are bi-predicted from adjacent reference pictures (I- and P-) and, hence, are of limited importance. Other more, complex weighting schemes [26] are possible but these may not be effective in real-time or may involve encoder intervention, which reduces their generality.

Consider the problem of finding how many constant-sized packets may be interleaved before there is a risk of missed display deadlines. For simplicity, assume that all packet-level FEC is contained within the packets themselves, rather than have separate FEC packets<sup>2</sup>. Assume also that there is a single transmit buffer and a single-layered stream of arriving encoding video data. If the buffer with FIFO service discipline is already occupied then there is a possibility of interleaving packets while the buffer empties. If  $T_s$  is the service time of the buffer,  $B$  is the number of packets currently buffered when an arriving packet is ready to be interleaved, then the time to the display deadline,  $T_d$ , is related to the maximum number of packets that can be collected for interleaving,  $M$ , by

$$T_d = (B + M)T_s \quad (1)$$

For a Time Division Multiple Access (TDMA) channel with centralized scheduling and no competing cross traffic, as, in a single Bluetooth interconnect,  $T_s$  is simply the transmission time and is thus known. However, the model is general as  $T_s$  may be variable, for example depending on medium access contention time as in IEEE 802.11.

Eqn. (1) can obviously be rearranged to find:

$$M = \frac{T_d}{T_s} - B \quad (2)$$

If each arriving packet occurs after inter-arrival time  $T_{in}$ , then the desire to avoid the buffer emptying before interleaving of subsequent arriving packets is complete is given by inequality (3).

$$(M - 1)T_{in} < BT_s \quad (3)$$

Substituting (2) in (1) results in an expression for the number of buffered packets that should already exist in the buffer at the moment of arrival of a packet with deadline  $T_d$ :

<sup>2</sup> As Section 3.2 discusses, it is actually preferable to have separate FEC packets.

$$B > \left( \frac{T_d}{T_s} - 1 \right) \left( 1 + \frac{T_s}{T_d} \right) \quad (4)$$

As previously mentioned,  $T_d$  itself should be conservative estimated, i.e. less than its maximum extent depending also on the packet's picture type importance. This is because if cross traffic is present, then  $T_s$  will be variable even with a centralized scheduler, as this scheduler will spend time servicing other traffic. An estimate of delay introduced by other cross traffic is required, which requires time-smoothed estimation based on previous arrival patterns. In this paper's examples a linear predictive filter [8] is employed. Adaptive linear prediction is especially suited to real-time variable bit rate (VBR) video [1] because it does not require prior knowledge of video statistics and does not require stationarity in time.

If the transmit buffer is already empty when the packet commencing subsequent interleaving arrives then there is no dependency on already buffered packets:

$$T_d = (M - 1)T_{in} + M T_s \quad (5)$$

which can be rearranged in terms of  $M$ , though (4) is obviously no longer true.

The examples in this paper relate to Bluetooth, but this is an example of a number of systems with centralized scheduling control. Wibree from Nokia is similar to Bluetooth with a gross air rate of 1.0 Mb/s (the Bluetooth basic gross air rate), but able to employ button-cell batteries. IEEE 802.15.4 (Zigbee) has some similarities to Bluetooth. Though its maximum data-rate is limited to 250 kbps, it is still able to stream video [5], compared to IEEE 802.11 (Wi-Fi)'s [6] typical current usage of 100-350 mA, Bluetooth's consumption is 1-35 mA, implying that for mobile multimedia applications with higher bandwidth capacity requirements, Bluetooth is a preferred solution. However, both IEEE 802.11's Point-Coordination Function and IEEE 802.11e's Hybrid Coordination Function allow for centralized packet scheduling and, hence, techniques applicable to Bluetooth are to some extent transferable to these. Centralized scheduling is preferable [14] for multimedia scheduling because of the problem of delay in real-time applications.

The remainder of this paper is organized as follows. Section 2 reports related work, mainly concerning interleaving and FEC for video communication over wireless channels. Section 3 prepares the way for the results reported in Section 4, but also, of course details the interleaving and FEC schemes. Finally, Section 5 draws some conclusions.

## 2. RELATED WORK

The work in [16] reports an influential model for the differential impact on video quality of packet losses, depending on packet burst length. Consequently in [15], simple packet interleaving (without FEC) was applied. In all cases, simple interleaving resulted in an improvement in quality, though it did not follow that continuing to increase the amount of interleaving resulted in

continued improvement of the received video quality. [15] considered the level of interleaving for a fixed-delay threshold but not the possibility of adapting the extent of interleaving. In [4], packet-level interleaving was taken further by spreading the video packets according to their content importance.

Broadcast/multicast [21] and communication over high delay-bandwidth product networks [9], i.e. any applications for which ARQ is inappropriate, are two applications of packet-level FEC. In fact, [21][9] employ NAKs, which, when received, require the application to send additional FEC bearing packets. Mixed packet level and symbol-level FEC for video multicasting are additionally considered in [13].

Video streaming is also an application in which ARQ may be unacceptable if delay is allowed to increase. For example, in Bluetooth the default ARQ setting permits an unbounded number of retransmissions. During a period of severe disruption [28] such retransmissions cause delay not only to a stalled video-bearing packet but also to any packets awaiting their turn for transmission. An interesting comparison between packet-level FEC and ARQ in terms of hardware implementation occurs in [20], when it is shown that Reed-Solomon (RS) burst erasure coding<sup>3</sup> can be implemented in a way that causes less delay than ARQ, while constructing the codes.

Packet-level interleaving is combined with FEC in [22] with the degree of interleaving dependent on the size of an MPEG-4 Video Object Plane (VOP). Because all interleaving is confined within a FEC, the authors of [22] did not consider delay to be an issue. However, in the general case if packet interleaving can be made delay-aware then considerable protection is afforded against burst errors.

### 3. METHODOLOGY

#### 3.1 Bluetooth Video Streaming

Like IEEE 802.11b,g (WiFi) Bluetooth operates in the unlicensed 2.4 GHz band. Bluetooth's short range (less than 10 m for class 2 devices), robust Frequency Hopping Spread Spectrum (FHSS) and centralized medium access control through TDMA and Time Division Duplex (TDD) means it is less prone to interference from other Bluetooth networks. Bluetooth employs variable-sized packets up to a maximum of five frequency-hopping time-slots of 625  $\mu$ s in duration. In TDD for transmit/receive separation, every Bluetooth frame consists of a packet transmitted from a transmitter node over 1, 3 or 5 timeslots, while a receiver replies with a packet occupying at least one slot, so that each frame has an even number of slots.

Bluetooth v.1.2 received comparatively limited investigation as a medium for streaming video. However, B/T v. 2.0 [27] increased the maximum gross user payload (mgup) bit rate from a basic rate of 0.7232 Mbps to 2.1781 Mbps. In fact, Bluetooth v. 2.0 Enhanced Data Rate (EDR) supports both a gross air rate of 3.0 Mbps and 2.0 Mbps (mgup of 1.4485 Mbps), through respectively  $\pi/4$ -DQPSK or 8DPSK modulation. In EDR, the symbol rate (1 Msps) remains the same as in the basic rate.

<sup>3</sup> An erasure code depends on knowing the location of incorrect packets.

Because of the variety of Bluetooth packet types and the additional modulation modes, it is important to establish in any experiments or simulations which packet type should be selected for a particular channel Carrier-to-Noise Ratio (CNR). Table 1 summarizes all the additional ACL mode EDR packet types currently available (according to the specification), as well as EDR DM-type packets (suggested as additional packet types) in the event that symbol-level FEC were to be added to EDR. In Table 1, only the maximum forward asymmetric data rates are shown, assuming a single time-slot return packet is also included in the Bluetooth frame.

Importantly, the user payload additionally includes a one or two byte CRC to detect packet error with a high likelihood. This means that in packet-level FEC an incorrectly received packet can be detected and erasure correction becomes appropriate. The incorrect packet can be discarded as it can be recreated from correctly received packets with the help of additional FEC-only packets (refer to Section 3.2).

**Table 1. Basic and EDR packet types in Bluetooth ACL mode**

Packet type	User payload in bytes	Asymmetric max. rate in kb/s
2-DM1	0-36	230.4
2-DM3	0-245	782.9
2-DM5	0-453	965.7
2-DH1	0-54	345.6
2-DH3	0-367	1174.4
2-DH5	0-679	1448.5
3-DM1	0-55	354.1
3-DM3	0-368	1184.3
3-DM5	0-681	1452.0
3-DH1	0-83	531.2
3-DH3	0-552	1776.4
3-DH5	0-1021	2178.1

Length and master to slave bitrates, for a single ACL master-slave logical link, with DM = Data Medium rate (FEC to be added) and DH = Data High rate (no FEC). 2-DH3 is 2.0 Mbps modulation three time-slot packet.

The Bluetooth Packet Error Rate (PER) is derived as follows. If  $p$  is the probability of a bit error then the PER for a DH packet (without symbol-level FEC) is

$$PER = 1 - (1 - p)^s, \quad (6)$$

where  $s$  is the packet size in bits. If it is assumed that only a single bit is correctable then the PER for DM (with symbol-level FEC) is

$$PER = 1 - ((1 - p)^{15} + 15p(1 - p)^{14})^{\frac{s}{15}} \quad (7)$$

given that the packet is divided into blocks of 15 bits. Notice that there are fifteen different ways for one bit error to occur if the remaining fourteen are without error.

If the ability of an expurgated (10,15) Hamming code to correct two consecutive errors is taken into account then an extra term should be added to account for the fourteen possible ways of selecting two consecutive bits from fifteen:

$$PER = 1 - ((1 - p)^{15} + 15 p(1 - p)^{14} + 14 p^2 (1 - p)^{13})^{\frac{s}{15}} \quad (8)$$

Substitution of (8) for (7) will be most noticeable at Bluetooth basic rates, though, in this paper, we do not report results on the basic rate. From one of the expressions for PER the data throughput in bps is found by

$$T = \frac{s(1 - PER)}{(v + 1)t} \quad (9)$$

where  $v$  is the number of time slots occupied by the Bluetooth packet and  $t$  is the duration of the Bluetooth time slot.

However, it should be carefully noted that  $p$  in (7), (8) and (9) depends on the type of modulation and will vary accordingly [23]. It should also not be applied *without* regard to modulation type as apparently erroneously occurred in [3], because, especially if (7) rather than (8) is applied, different and misleading estimates will arise.

Taking into account (7) and (9), with the equations relating CNR to  $p$  [23], and with double adjacent error and all single error correction, results in plots, Fig. 1, for the throughputs according to different CNRs. From the Figure, it is apparent that certainly below CNRs of 10 dB, only 2DM packets of the EDR types should be selected, though basic rate packets may also be selected. Equally 3DM packets have no role, contrary to results reported in [3], where 3DM had a role. The impact of packet size on 2D type packets is shown in Fig. 2, when it is apparent that there is a throughput penalty if smaller packet sizes are selected, while there are also optimal values for packet length, corresponding to fully-filled Bluetooth packets. This implies that the advantage of fully-filled packets may well outweigh the advantage of preserving slice boundaries in video, even though slices afford some error resilience, a topic returned to in Section 3.4.

### 3.2 FEC and Interleaving

In this paper, at the symbol level, FEC and interleaving follow the Bluetooth model. Each packet of length payload,  $s$ , is divided up into  $s/15$  blocks, with each block constituting a systematic FEC codeword with 10 data bits and 5 FEC-coded bits for a rate of  $2/3$ . In this work we assume that the header is immune to error since it is always transmitted at the reduced basic rate, and is protected with a triple redundancy code. As mentioned in Section 1, the generator for the Bluetooth (15,15) binary Hamming code is that of an expurgated code, i.e.

$$g(X) = (1 + X)(1 + X + X^4) \quad (10)$$

and, hence, a potential receiver can not only detect and correct single errors but also double consecutive errors. For interleaving, the bit sequence may be represented as being read into the columns of a matrix with height 15, whereas bits are read out into a packet by rows with width  $s/15$ . The result is that formerly adjacent bits are separated by distance  $s/15 - 1$ , which is the burst-size immunity provided.

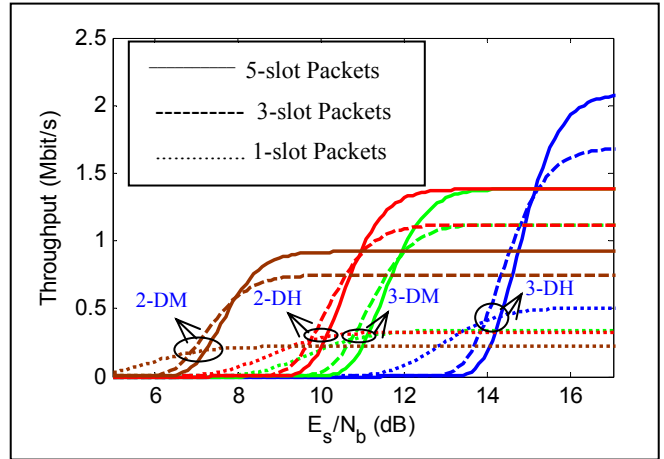


Figure 1. Throughput versus CNR for different EDR packet types, with and without symbol level FEC using double adjacent and all single error correction.

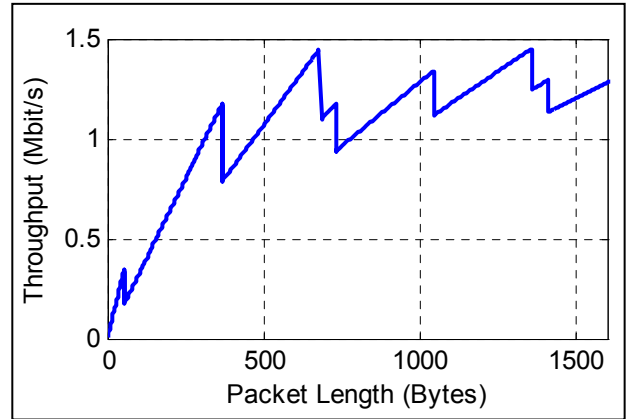


Figure 2. Packet length vs. maximum throughput for 2D packets.

For packet-level FEC, various schemes are possible [17], including a simple XOR scheme [9]. However, in this paper we suppose that if a packet has  $b$  bits then a codeword is taken from  $GF(2^b)$  according to RS erasure coding [18][21], assuming  $2^b$  is much greater than the number of arriving packets,  $K$ . If  $y_i \in GF(2^b)$  is the  $i^{\text{th}}$  packet data of length  $b$ , then FEC packets are formed from polynomials of the form

$$F(X) = y_1 + y_2 X + \dots + y_K X^{K-1} \quad (11)$$

using powers of  $\alpha$ , a primitive root of the Galois field GF, as the variable in (11). Then, by sending  $K + h = N$  packets, any  $K$  correctly received packets serve to reconstruct the original  $K$  packets. It is known that no other code can recover packets from fewer received packets, i.e. RS is a maximum distance separable code [18].

The advantage of separate FEC packets rather than embedding separate FEC on a per GOP basis with each packet and providing ‘horizontal’ FEC across packets was confirmed in [30]. Though [30] uses packet-level FEC, no interleaving took place and bit stuffing was necessary, as packets were not of constant size. In our study, interleaving proceeds similarly to symbol level interleaving. Each column of the interleaving matrix is formed by a sequence of  $N$  packets, while the packets are read out across the rows of the matrix. Fig. 3 illustrates the scheme.  $M$  is again the total number of data packets allowable by the delay-aware scheme. The numbering of packets gives the transmission order, which is the same as the packet arrival order. Because FEC is applied across non-consecutively transmitted packets, the FEC protection is better able to protect against a burst that affects more than one packet of the packet stream.

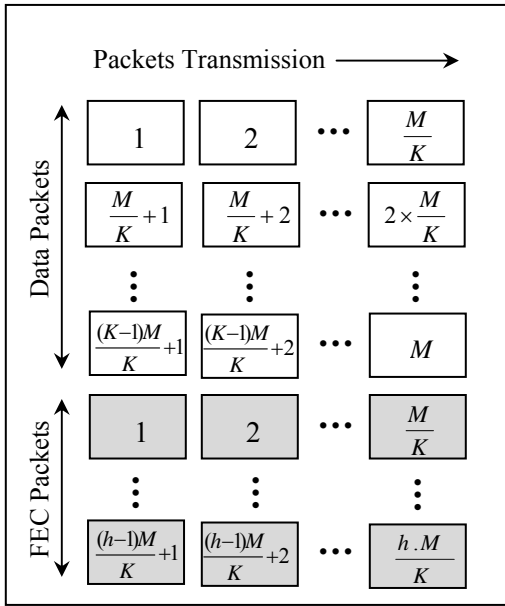


Figure 3. Packet interleaving scheme with FEC packets.

### 3.3 Channel Model

A two state discrete-time, ergodic Markov chain models the wireless channel error characteristics between a Bluetooth master and slave node. To simplify the interpretation probability of bit error in the good state was set at 0, while the probability of bit error in the bad state was set to 1. A similar channel model was adopted for the very same reason in [15], though in that research the time in any state was quantized in proportion to the frame duration. The mean duration of a good state was set to 30 kbit,

which corresponds to a probability of going from a good state (g) to a bad state (b) of  $P_{gb} = 3.33 \times 10^{-5}$  as

$$T_g = \frac{1}{1 - P_{gg}}, P_{gb} = 1 - P_{gg} \quad (12)$$

### 3.4 Arrival Rate Prediction

In order to estimate the extent of packet interleaving, an estimate of the video bit stream arrival rate needs to be made. It turns out that constant-sized Bluetooth packets are optimal for Bluetooth communication of video, and, hence, bit stream arrival rate is equivalent to packet inter-arrival rate. The rationale behind use of constant-sized Bluetooth packets is discussed later on in this Section.

The predicted rate was arrived at by a  $P$ -order linear prediction filter (LPF). The  $P$ -order LPF prediction filter is represented by

$$X(m+1) = \sum_{k=1}^P w_k \cdot X(m-k+1), \quad (13)$$

where  $X(m+1)$  is a predicted value estimated from  $P$  previous samples, while the  $w_k$  are the adaptive filter weights indexed by  $k$ . The weights are estimated [8] through (14)

$$\mathbf{w}(m+1) = \mathbf{w}(m) + \frac{e(m) \cdot \mathbf{X}(m)}{\|\mathbf{X}(m)\|^2}, \quad (14)$$

when  $\mathbf{w}(m)$  is  $m^{\text{th}}$  sample of the length  $P$  column vector of weights and  $\mathbf{X}$  is a length  $P$  column vector of measurements over time, as in (15).

$$\mathbf{X}(m) = [X(m), X(m-1), \dots, X(m-P+1)]^T, \quad (15)$$

when  $T$  represents the vector transpose. The variable  $e(m)$  is the error between the measured and the predicted value.

Unfortunately, if packetization takes place on a single video slice (one row of macro-blocks in the MPEG-2 codec) per Bluetooth packet the result is partially filled packets as well as many 1- or 3-slot packets, with a consequent drop in throughput (see Figure 2). Therefore, in [25] fully-filled Bluetooth packets were formed, regardless of slice boundaries. While this results in some loss in error resilience, as each slice contains a decoder synchronization marker, in [25] it is shown that the overall video performance is superior, as the throughput is superior. The H.264 codec allows for variable-sized slices, but, without knowledge in advance of channel conditions, the optimal packet mode and consequently the optimal slice size is not known, and, so in that respect H.264 is in no better position than earlier codecs with a limit on maximum slice extent.

### 3.5 Algorithm Implementation

The weighting system mentioned in Section I as a means of establishing a conservative estimate was applied as follows. Two weighting coefficients are applied to the known display deadline

for each packet. This display deadline value is constant for all packets.

The first coefficient,  $\alpha$ , accounts for the position of a packet in an interleaved group of protected packets. Each row of packets within the group will experience different delays. For example, in Fig. 3 packets numbered 1 to  $M/K$  have to wait for at least  $(M + (h - 1)(M / k) + 1)$  other packets to be received, if FEC functionality needs to be applied. Therefore, there are  $K$  different levels of coefficients:  $\alpha_i$   $1 \leq i \leq K$ , each corresponding to a row of data packets; where  $\alpha_i < \alpha_j$  if  $j > i$ . We set  $\alpha_1 = 2, \alpha_2 = 1$  for just two rows of packets.

The second weighting coefficient  $\beta$  considers the picture type of the packet. The algorithm is more conservative when dealing with reference packets, i.e.  $\beta_I < \beta_P < \beta_B$ . In simulations,  $\beta_I = 0.80$ ,  $\beta_P = 0.85$ ,  $\beta_B = 0.90$ .

Finally, a joint weighting of the common display deadline is formed by summing these weights across the group of interleaved packets to produce combined weighting  $W$ .

$$W = \frac{\sum_{x=1}^M \alpha_x \cdot \beta_x}{\sum_{x=1}^M \alpha_x} \quad (16)$$

Note that in (16) for packets that are in the same row of the interleaving matrix the values of  $\alpha_x$  will be the same, as previously stated.

A further important feature of the implementation of the delay-aware implementation is ‘lazy’ evaluation or last-minute determination of the number of interleaved packets in a group,  $M$ . The decision is normally taken at the point in time when there are just enough packets in the transmit buffer to overlap with the time that it takes to construct the FEC. This procedure has an important advantage because it means that the various estimates required to form  $M$  are given the maximum time to be right. Prematurely making an estimate of the arriving traffic rate might lead to an over-ambitious estimate of  $M$ , leading to missed display deadlines.

## 4. RESULTS

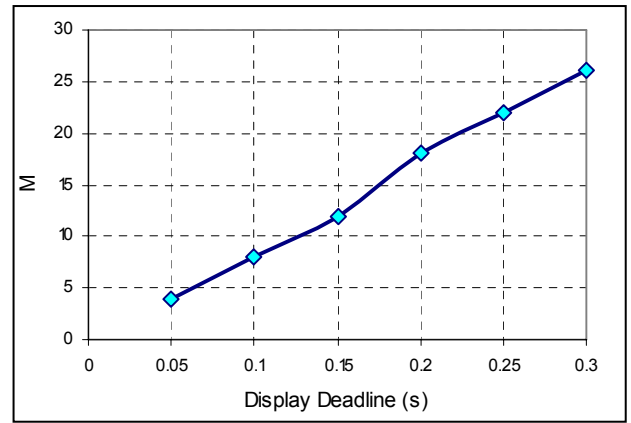
We were pleased to employ the University of Cincinnati Bluetooth (UCBT) extension to the well-known ns-2 network simulator (v. 2.28 used). The UCBT extension supports Bluetooth EDR but is also built on the air models of previous Bluetooth extensions such as BlueHoc from IBM and Blueware.

Simulations were carried out with input from an MPEG-2 encoded bitstream at a mean rate of 1.5 Mbit/s for a 40 s video clip with moderate motion, showing a newsreader and changing backdrop, which we designate ‘News’. PSNR was found by reconstructing with a reference MPEG-2 decoder. The display

rate was 25 frame/s resulting in 1000 frames in each run. The source video was European SIF-sized ( $366 \times 288$  pixels) with a GOP structure of  $N = 12$ , and  $M = 3$ .

Fig. 4 plots values of  $M$  selected by the algorithm against the display deadline setting, assuming RS(2,3), i.e. two video packets for every one FEC packet. With a suitable-sized playout buffer at the receiver then quite high extents of packet interleaving are permissible.

Figs. 5 and 6 are a comparison of symbol interleaving and two different levels of packet interleaving. The Figures show packet loss and matching video quality, after averaging over 20 runs to establish convergence. The packet size was for 2-DM3 type (Table 1) and the channel model was the simplified Marcovian burst error channel of Section 3.3. For symbol interleaving within a packet the block size was 15 bits. For symbol interleaving, as previously mentioned, the expurgated (10,15) Hamming code was employed with double consecutive error correction (as in Section 3.1).



**Figure 4. Relationship between display deadline and the number of interleaved packets in a group.**

The packet-level FEC ratio for RS erasure correction coding was set at the same level as for the Bluetooth Hamming code, i.e. a rate of 2/3. This meant that for every two packets one FEC packet was generated.

It is clear from Figs. 5 and 6, that there is a significant advantage from packet interleaving and that advantage grows if the level of interleaving increases from a maximum of four packets to a maximum of eight packets (not including FEC packets). The only situation in which symbol interleaving is competitive is when packet burst size is short. This is not surprising, as symbol level interleaving only converts burst errors to random errors for burst sizes less than  $s / 15 - 1$  bits. However, the degree of difference is surprising, given that the PSNR plots are on a logarithmic scale.

## 5. CONCLUSIONS

Symbol-level interleaving with FEC is commonly applied to combat noise and interference across a wireless channel. However, this paper has shown that packet-level interleaving, at

least in the Bluetooth case study, is much superior to symbol-level interleaving when used with RS packet erasure coding. Given that Bluetooth detects packet error through CRC, packet-level interleaving with FEC should normally be preferred. However, interleaving causes delay. Therefore, this paper introduces a delay aware algorithm for calculating the extent of interleaving. The algorithm is conservative, avoiding any risk of missed delays by taking into account the position of packets within an interleaving matrix and the picture type importance of the packets. Moreover, because estimates of packet arrival rate are required, the algorithm decides the extent of interleaving at the last moment, which we call lazy evaluation, so as to form a firm estimate. Further work consists of comprehensive evaluation of the algorithm and combining symbol- with packet-level interleaving to judge what may be gained. We also intend to use Fountain codes [19] as an improvement upon RS erasure codes.

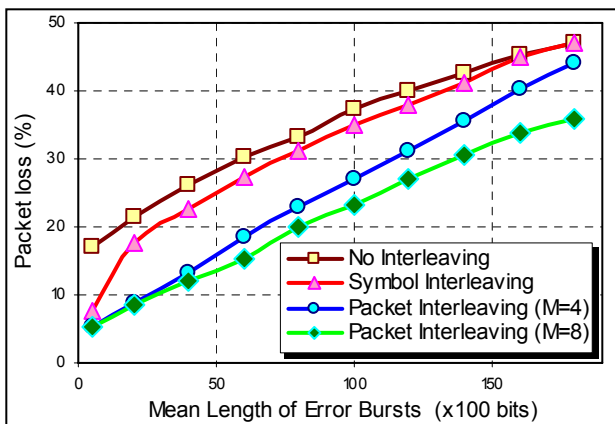


Figure 5. Packet loss after streaming the News video across a Bluetooth channel.

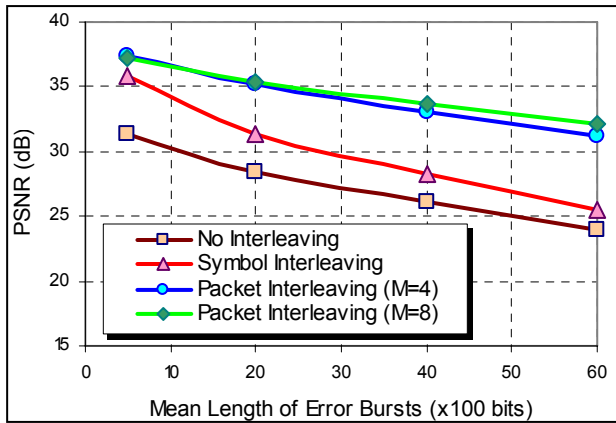


Figure 6. Video quality after streaming the News video across a Bluetooth channel.

## 6. ACKNOWLEDGMENTS

This work was supported by the EPSRC, UK under grant no. EP/C538692/1.

## 7. REFERENCES

- [1] Adas, M. A., "Using adaptive linear prediction to support real-time VBR video under RCBP networks service model", *IEEE/ACM Trans. on Networking*, vol.6, no. 5, pp.635-644, 1998.
- [2] Cai, J. and Chen, C. W., "FEC-based video streaming over packet loss networks with pre-interleaving", in *Int. Conf. on Information Tech.: Coding and Computing*, pp. 10-14, April 2001.
- [3] Chen, J.-J., Sun, T., and Chen, Y.-C., "Improving Bluetooth EDR throughput using FEC and interleaving", *Int. Con. on Mobile Ad Hoc and Sensor Networks*, pp. 726-736, Hong Kong, 2006.
- [4] Choi, J.-Y. and Shin, J., "Content-aware packet-level interleaving method for video transmission over wireless networks", in *Wired/Wireless Internet Comms.*, pp. 154-163, 2005.
- [5] Deshpande, S., "Adaptive low bitrate streaming over IEEE 802.14.4 low rate wireless personal area networks (LR-WPAN) based on link quality indication", *Int. Wireless Comms. and Wireless Computing Conf.*, pp. 863-868, July 2006.
- [6] Ferro E. and Potori, F., "Bluetooth and Wi-Fi wireless protocols: A survey and a comparison", *IEEE Wireless Communications*, vol.12, no. 1, pp. 12-26, Feb. 2005.
- [7] Haartsen, J., "The Bluetooth radio system", *IEEE Personal Comms.*, vol. 7, no. 1, pp. 28-26, 2000.
- [8] Honig, M. K. and Messerschmitt, D. G., *Adaptive Filters Structures, Algorithms, and Applications*, Kluwer, Boston, MA, 1990.
- [9] Huitema, C. "The case for packet level FEC", in *5<sup>th</sup> Workshop in Protocols for High-Speed Networks*, pp. 109-120, Oct. 1996.
- [10] Jurca, D. and Frossard, P., "Packet media streaming with imprecise rate estimation", *Advances in Multimedia*, 2007.
- [11] Lampe, L., Jain, M. and Schober, R., "Improved decoding for Bluetooth systems", *IEEE Trans. on Comms.*, vol. 53, no. 1, pp. 1-4, 2003.
- [12] Lee, C.-W., Yang, C.-S., and Su, Y.-C., "Adaptive UEP and packet size assignment for scalable video transmission over burst-error channels", *EURASIP J. on Applied Signal Processing*, vol. 2, pp. 1-8, 2005.
- [13] Lee, T.-W. A., Chan, S.-H. G., Zhang, Q., Zhu, W.-Z. and Zhang, Y.-Q., "Optimal allocation of packet-level and byte-level FEC in video multicasting over wired and wireless networks", in *IEEE GLOBECOM*, pp. 1994-1998, Nov. 2001.

- [14] Li, Q., van der Schaar, M., "Providing QoS to Layered Video Over Wireless Local Area Networks Through Real-Time Retry Limit Adaptation", *IEEE Trans. on Multimedia*, vol. 6, no. 2, pp. 278-290, 2004.
- [15] Liang, Y. L., Apostolopoulos, J. G. and Girod, B., "Model-based delay distortion optimization for video streaming using packet interleaving", in *37<sup>th</sup> Asilomar Conf.*, pp. 1315-1319, Nov. 2002.
- [16] Liang, J. Y., Apostolopoulos, J. G. and Girod, B., "Analysis of packet loss for compressed video: Does burst-length matter?", in *Proceedings of IEEE Int. Conf. on Acoustics, Speech, and Signal Processing*, vol. 5, pp. 684-687, Hong Kong, April 2003.
- [17] Libman, L. and Orda, A., "Optimal packet-level FEC strategies in connections with large delay-bandwidth products", *IEEE Trans. On Wireless Comms.*, vol. 5, no. 7, pp. 1645-1649, 2006.
- [18] Lin, S. and Costello, D. J., *Error Control Coding*, 2<sup>nd</sup> edition, Prentice Hall, Upper Saddle River, NJ, 2004.
- [19] MacKay, D.J.C., "Fountain codes", *IEE Proceedings on Communications*, vol. 152, no. 6, pp. 1062-1068, 2005.
- [20] McAuley, A. J., "Reliable broadband communication using a burst erasure correcting code", in *ACM SIGCOMM*, pp. 297-306, Sept. 1990.
- [21] Metzner, J. L., "An improved broadcast retransmission protocol", *IEEE Trans. On Comms.*, vol. 32, no. 6, pp. 679-683, June 1984.
- [22] Nafaa, Ahmed, T. and Mehaoua, A., "Unequal and interleaved FEC protocol for robust MPEG-4 multicasting over wireless LANs", in *IEEE Int. Conf. on Comms.*, pp. 1431-1435, June 2004.
- [23] Proakis, J. G., *Digital Communications*, McGraw-Hill, Boston, 2001.
- [24] Qu, Q., Pei, Y., Modestino, J. W., and Tian, X., "Source-adaptation-based wireless video transport: A cross-layer approach", *EURASIP J. on Applied Signal Processing*, pp. 1-14, 2006.
- [25] Razavi, R., Fleury, M. and Ghanbari, M., "An Efficient Packetization Scheme for Bluetooth Video Transmission", *Elec. Letters*, vol. 42, no. 20, pp. 1143-1145, 2006.
- [26] Shan, Y., "Cross Layer Techniques for Adaptive Video Streaming over Wireless Networks", *EURASIP J. on Applied Signal Processing*, vol. 2, pp. 220-228, 2005.
- [27] Specification of the Bluetooth system --- 2.0 + EDR, Available online at <http://www.bluetooth.com>, 2004.
- [28] Subramanian, V., Ramakrishnan, K. K. and Kalyanaraman, S., "Disruption-tolerant link-level mechanism for extreme wireless networks environments", in *IEEE/Create-Net/ICST Int. Conf. on Comm. Sys. Software and Middleware*, Bangalore, Jan. 2007.
- [29] Valenti, M. C. and Robert, M. "Improving the QoS of Bluetooth through Turbo Coding", *IEEE Military Conf.*, pp. 1057-1061, 2002.
- [30] Zhai, F., Eisenberg, Y., Luna, C. E., Pappas, T. N., Berry, R., Katsaggelos, A. K., "Packetization schemes for forward error correction in Internet video streaming", in *41<sup>st</sup> Allerton Conf. on Comms., Control, and Computing*, Oct. 2005.