BLOCK-BASED, TYPE I HYBRID ARQ FOR POWER-EFFICIENT VIDEO STREAMING OVER A WIRELESS INTERCONNECT

R. Razavi M. Fleury M. Ghanbari
UNIVERSITY OF ESSEX, UNITED KINGDOM

Reducing retransmitted data arising from wireless channel errors contributes to power savings on battery-powered wireless devices. The paper introduces block-based hybrid ARQ in which only blocks with non-correctable errors are retransmitted, rather than complete packets. In video streaming, as an unbounded number of retransmissions results in effective packet loss through missed display deadlines, this form of type I hybrid ARQ should be deadline aware. In a Bluetooth example, the scheme is moderated by relative packet efficiency. The result is significant improvements in video quality at the receiver, while at the same time the transmission power efficiency also improves.

Keywords: Wireless, error control, video streaming

1. INTRODUCTION

In a wireless network, error control is crucial, owing to the presence of radio frequency noise and many causes of interference such as multi-path and fast fading. In general, as a form of error control for video streaming, Automatic Repeat Request (ARQ) is preferred [1]. The alternative, Forward Error Control (FEC), results in an overhead in packet size and computation power usage whether or not the channel conditions are poor. For selective request (SR) ARQ, retransmission is effectively a gamble with battery power and delay, because the cause of errors may still exist, though for fast ARQ, when ARQ is automatic, the situation is somewhat different. Hybrid ARQ (HARQ), the subject of this paper, combines ARQ with FEC, and with a novel form of block-based HARQ, we show that power efficiency is considerably improved in some situations compared to packet-based retransmission.

Consider an IEEE 802.15.1 (Bluetooth) [2] wireless interconnect, in which master-slave node polling results in an exchange of packets, forming a frame consisting of an even number of time slots. Because of send/receive separation at the transceiver, a reply packet will always be available for ARQ. (Send/receive separation makes for a single chip implementation.) The default Bluetooth policy [3] is that, if an error occurs, ARQ will continue until the transmission succeeds, though there is an eventual timeout. For a mobile device, this policy will have an impact on battery power usage, as approximately 70% of power is consumed in transmission [4][5]. Efficient ARQ management is the key to both power management and for a video application, ensuring acceptable display quality at the receiver device. However, ARQ management is a multi-faceted control problem, as account must also be taken of wireless channel conditions, and the display deadlines of the pictures being transmitted.

In the case of video streaming, default Bluetooth ARQ is unsatisfactory. Repeated transmission of a packet may result not only in missed display deadlines (the packet is effectively lost) but the waiting time of all queued packets increases, also with a risk of missed deadlines. On the other hand, motion estimation and compensation allows redundancy between successive frames to be exploited. Consequently, a loss of a packet bearing data from an MPEG encoded video bit-stream [6], may have an enduring effect within a Group of Pictures (GOP) --- usually 12 to 15 pictures. SR HARQ [7] is one proposed solution to this problem. A packet that has been found to have non-correctable errors is re-sent, usually with an enhanced level of FEC or help to correct the original packet.

Unfortunately for Bluetooth, this proposal, as its stands, has some weaknesses. Bursts of errors are common on a wireless channel, while Bluetooth FEC is only able to correct single errors and two adjacent errors [8], and, indeed, it is often assumed, for example [9], that a Bluetooth receiver can only decode single, randomly occurring errors. Just as with other wireless systems, a further weakness is that if the whole packet is retransmitted then the size of the packet results in delay and power efficiency is violated.

In this paper, a simple or type 1 form of link layer HARQ [10] is proposed, whereby only blocks within a packet that have incurred errors are retransmitted, thus reducing the data retransmitted. This scheme is appropriate for situations where block level FEC is adequate (Additive White Gaussian Noise and Fast Fading channels) and is not suitable during deep fades or frequency collisions in a slow frequency hopping spread spectrum system (FHSS) (as Bluetooth is). The former is addressed by packet-level interleaving and FEC, whereas adaptive frequency hopping [11] is already a Bluetooth feature intended to cope with the latter problem.

The existing Bluetooth FEC scheme is able to detect all two error patterns within a block. As more than three errors are very rare within the 15-bit blocks used even under high error conditions, the existing FEC scheme can also act as a block error detector. For both the original transmission and the subsequent transmission, FEC is applied but no form of FEC enhancement occurs. As an example, Bluetooth's fast (stop-and- wait) ARQ packets would return a bit map indicating which blocks were in error. These blocks are either retransmitted immediately in a short packet or are accumulated with subsequent retransmitted blocks. The size of a retransmitted packet becomes a trade-off between: the number of expected blocks in error; the amount of data needed in a block header to indicate the identity of the block; and the packet error rate, which has a dependency upon packet size. Retransmitted blocks may also be subject to error, and, therefore, a retransmission timeout should be set, which will be less than the display deadline. Block-based HARQ is suitable for higher error conditions. Therefore in the paper, it is simulated using the Bluetooth version 2 packet type and data rate appropriate to such conditions but with the addition of block-level FEC. Video is normally transported through UDP and, therefore, no conflict with transport layer ARQ will arise.

The paper supplies a mathematical model for block-based HARQ. The model confirms the advantage of this form of HARQ over default Bluetooth ARQ in a low Carrier-to-Noise Ratio (CNR) channel. In particular, the ratio of successfully received bits to transmitted bits is shown to be superior, which immediately implies the power efficiency of the scheme. Though, this paper has taken Bluetooth as an example, the technique is applicable to video streaming over wireless networks in general, as its main requirements are simply block-based FEC and ARQ, which could be of a selective request form.

2. RELATED WORK

In [12], it was observed that classic error control methods work poorly in terms of energy conservation, in line with similar comments in Section 1. It was proposed in [12] that the channel should be probed to find the error conditions, whereupon the level of ARQ retransmissions is adjusted. However, the volatility of the wireless channel may make measurements unreliable. The work in [13] proposed a scheme of error control which varied according to the channel conditions and to the relative energy budget for RS coding and SR ARQ. As in our paper, a two-state type model allowed (Rayleigh) fading conditions to be modelled, in way that is independent of packet size. It was found that there was a threshold, beyond which FEC was necessary, despite the increase in energy budget. Because TCP was assumed, link level retransmissions should take place before transport layer retransmission was needed. In the case of a speech transmission application packet drops through expired deadlines at the sender also occurred.

In [14], packet-level FEC (not block-level as in our paper) and power allocation are jointly optimised across cellular radio. The work combines layered video coding with FEC, with the degree of protection varying according the priority of the layer. The layers actually transmitted depend on the power resources of the sender.

3. BLUETOOTH VERSION 2

Bluetooth is a short-range (less than 10 m for class 2 devices), radio frequency interconnect. Bluetooth's robust FHSS and centralized medium access control through time division multiple access and 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 μs in duration. Every Bluetooth frame consists of a packet transmitted from a sender 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. Therefore, in master to slave transmission, a single slot packet serves for a link layer stop-and-go ARQ message, whenever a corrupted packet payload is detected.

The Enhanced Data Rate (EDR) of Bluetooth version 2.0 [3] now has a peak user payload of 2.2 Mb/s (gross air rate 3.0 Mb/s), which is the same average rate offered by some implementations of IP-TV. It also has an additional EDR mode with a gross air rate of 2.0 Mb/s. However, Bluetooth still must compete with lower power alternatives, such as Wibree from Nokia, intended for button-cell batteries, with a gross air rate of 1.0 Mb/s. Compared to IEEE 802.11 (Wi-Fi)'s [15] 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.

3.1 Bluetooth error control

For Bluetooth, an ARQ may occur in the following circumstances [16]: a) failure to synchronize on the access header code; b) header corruption detected by a triple redundancy code; c) payload corruption detected by cyclic redundancy check; d) failure to synchronize with the return packet header; e) header corruption of the return packet. Notice that a faulty ARQ packet can itself cause retransmission. The main cause of packet error [16], however, is c) payload corruption, which is the simplified assumption in this paper. As Bluetooth key header fields are protected, with the addition of bit scrambling, and, as headers are transmitted at the legacy gross air rate of 1.0 Mb/s even in EDR modes, the risk of header error is small.

The Bluetooth model of FEC adopted is as follows. Each packet of length payload, L, is divided up into $L/15$ blocks, with each block constituting a systematic FEC codeword with 10 data bits and 5

FEC-coded bits for a rate of 2/3. The generator for the Bluetooth (15,10) binary Hamming code [3] is that of an expurgated code, i.e.

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

and, hence, a potential receiver can not only correct single errors but also double adjacent errors [17], which is applicable in non-memoryless channels. Moreover, as with all Hamming codes, any two errors can be detected [18], allowing block-based error detection. Reed-Solomon coding is avoided in Bluetooth, despite its improved performance, because its processing is known to be a drain on battery power [13].

3.2 Bluetooth Packet Types

However, currently Bluetooth version 2 only includes FEC for the basic data-rate and not for EDR. Therefore, we have assumed FEC to be added to the EDR modes. Two new types of packet, 2DM and 3DM, are created and their characteristics have been added to Table 1. These packet types have different sizes according to the number of time slots occupied. The user payload in Table 1 for the new packet types does not include FEC bytes. However, of course, it is the packet length with the addition of FEC that is affected by errors (assuming a systematic code). Therefore, that length is given by the 2DH and 3DH equivalents.

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.

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

4. BLOCK-BASED TYPE-1 HARQ

4.1 Block-based Model

This Section provides a mathematical model for the proposed block-based HARQ. From the model, formulas for the power efficiency are derived for the cases considered, compared to the equivalent formulas for ARQ.

Consider a block-based FEC scheme, in which FEC is applied to blocks of size *b* bits making up a packet. Suppose the total size of a packet is *L* bits, then the probability of a packet in error is given by:

$$P_p = 1 - (1 - P_B)^{L/b} \quad (2)$$

where P_B is the probability of a block in error. The expected number of retransmissions, $E[N]$, for unbounded ARQ is

$$E[N] = 0 \times P_s + 1 \times P_S \times (1 - P_S) + 2 \times P_S \times (1 - P_S)^2 + ... \quad (3)$$

where P_S is the probability of a successful transmission. This means that

$$E[N] = \frac{1 - P_S}{P_s} \quad (4)$$

which implies that the expected total number of transmissions, $E[T]$, is simply

$$E[T] = E[N] + 1 = \frac{1}{P_S}. \quad (5)$$

However, $P_S = 1 - P_p$ from which it follows provided *L* is constant that the expected number of bits transmitted is

$$E[n] = \frac{L}{1 - P_p} \quad (6)$$

or equivalently, substituting (2) into (6)

$$E[n] = \frac{L}{(1 - P_B)^{L/b}} \quad (7)$$

Equation (7) corresponds to unbounded ARQ, with retransmission of a complete packet upon detecting a packet in error at the receiver.

An expected transmission power efficiency factor, P_E, can be defined as the expected number of successfully received bits divided by the expected number of successfully received bits. In other words P_E is the goodput divided by the throughput. For unbounded ARQ, as in this case the number of successfully received bits is simply the number originally transmitted

$$P_E = \frac{L}{E[n]} = (1 - P_P) = P_S \quad (8)$$

Consider now that only blocks in error are retransmitted but that each block in error requires *h* extra header bits to indicate its position in the packet from which it has been extracted. In general, *h* may be greater than block size *b*. Now the probability of an error within the retransmitted block and its additional header becomes

$$P_{BH} = 1 - (1 - P_B)^{(h/b)+1} \quad (9)$$

The expected total number of bits transmitted in this situation is given by

$$E[n] = L + P_B.L.\left(\frac{h}{b} + 1\right) + P_{BH}.P_B.L.\left(\frac{h}{b} + 1\right) + P_{BH}^2.P_B.L.\left(\frac{h}{b} + 1\right)\ldots \quad (10)$$

which simplifies to

$$E[n] = L.\left(1 + \frac{P_B.\left(\frac{h}{b} + 1\right)}{1 - P_{BH}}\right) \quad (11)$$

From (9) in the case of unlimited retransmission of blocks in error

$$P_E = 1 \Big/ \left(1 + \frac{P_B.\left(\frac{h}{b} + 1\right)}{1 - P_{BH}}\right) \quad (12)$$

Now consider a limited number of retransmissions, rather than as previously unbounded retransmission. Then for a maximum of just one possible retransmission using complete packet ARQ

$$P_E = \frac{L.(1 - P_P^2)}{L.(1 - P_P) + 2.L.P_P} \quad (13)$$

which again reduces to simply the probability of a successful transmission, i.e.

$$P_E = (1 - P_P) = P_S \quad (14)$$

For blocked retransmission, with just one retransmission, first set the probability of a successful transmission as

$$P_S = (1 - P_B)^{(L/b).((h/b)+1).P_B} \quad (15)$$

then the probability of a failed transmission is

$$P_F = 1 - P_S \quad (16)$$

causing the power efficiency to now be

$$P_E = \frac{L.(1 - P_P.P_F)}{(1 - P_P).L + P_P(L + P_B.L.((h/b) + 1))} \quad (17)$$

from which a further simplification would be to remove the common factor of *L*.

4.2 Block-based HARQ Performance

Fig. 1 compares the power efficiency of schemes with unbounded retransmission. No account has been taken of display deadlines in this Section. The plots for full-packet retransmission and block retransmission both include FEC according to the Bluetooth model, while the plot marked ‘No FEC’ includes retransmission. Therefore, the ‘No FEC’ plot is the same as the default Bluetooth ARQ scheme. The packet size was set to that of 2DM5, though recall from Section 3 that the effective size subject to bit errors is that of 2DH5 in Table 1. The plot for block-based retransmission includes a setting for the header overhead, *h*, as outlined in Section 4.1. The analytical results from Section 4.1 have been confirmed by means of a Monte-Carlo simulation. From Fig. 1 it is very clear that as the bit error rate (BER) increases then each of the protection methods becomes inadequate, leaving just the block retransmission scheme at higher BERs.

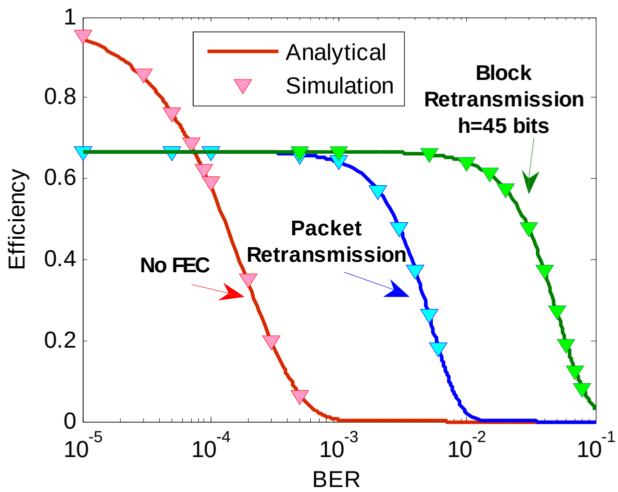


Figure 1: Bit Error Rate (BER) versus transmission power efficiency for unbounded retransmission.

When there is a maximum of just one retransmission, Fig. 2 shows the power efficiency in that case. From (8) and (14), the power efficiency of unbounded ARQ and single retransmission ARQ is the same, the probability of a successful packet transmission.

However, the gain in efficiency from employing block retransmission drops off sooner at higher BERs.

The power efficiency ratio of block retransmission (h = 45 bits) to packet retransmission is plotted in Fig. 3 for various BERs. Single retransmission was employed. The advantage of the block-based system is most apparent for smaller sized packets. There is a non-linear decline in efficiency ratio as the packet size increases. Therefore, because of the differing modulation types, larger packet sizes are not necessarily selected for higher BERs.

For the Bluetooth example in this paper, the efficiencies must be scaled by the relative advantages of the different packet sizes. Because the block-based retransmission scheme only re-sends a limited number of blocks, it is possible that partially filled packets will occur. On the other hand, for default Bluetooth ARQ with FEC, for video streaming we have employed fully-filled packets (within the CNR range tested). The choice of fully-filled packets is justified in Section 5. The packet efficiency, *PACeff*, for block-based ARQ under Bluetooth is found from:

$$PAC_{eff} = \frac{(no.of.time.slots.to.send.n.bits) \times T_{slot}}{\max.available.bit.rate / n}$$

where *Tslot* is the time duration of one slot, which in Bluetooth whatever the modulation scheme is 625 µs. The value of n, the number of bits transmitted depends on the number of blocks in a retransmission and the header overhead.

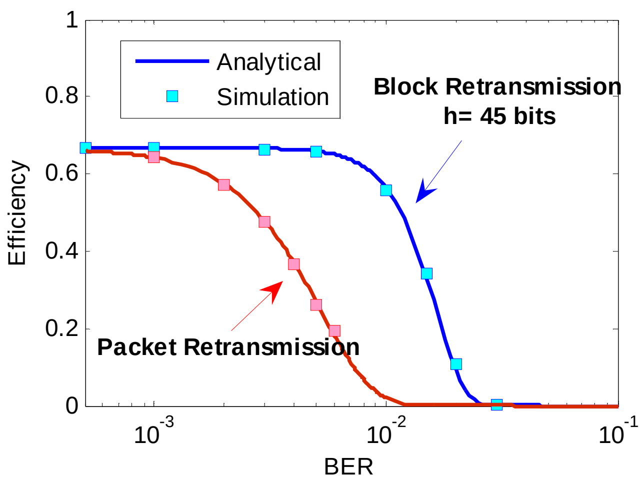


Figure 2: BER versus transmission power efficiency for single retransmission.

Fig. 4 charts the packet efficiency for various number, N, of packets from which blocks in error are selected. For example, $N=1$ means that however many blocks in error exist these then form one or more error packets which are retransmitted. Similarly, for $N=3$ all blocks in error from three successive packets are collected and retransmitted. Given even very high error conditions, experiments confirm that it is unlikely that more than one retransmission packet will be sent, even for $N=5$.

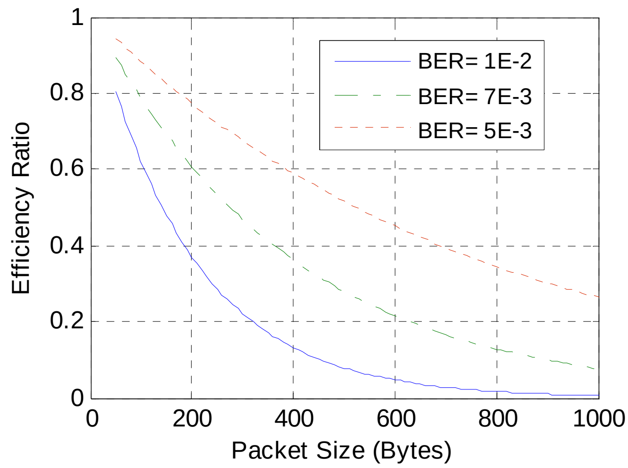


Figure 3: BER versus packet size for the transmission power efficiency of block based over packet-based single retransmission, both with FEC.

For low BERs, few sets of packets create error packets but increasingly more partially filled packets need to be re-sent, which leads to a decline in packet efficiency. As the BER worsens further, the number of blocks requiring retransmission increases. This results in the error packets being more fully filled with the result that packetization efficiency increases again.

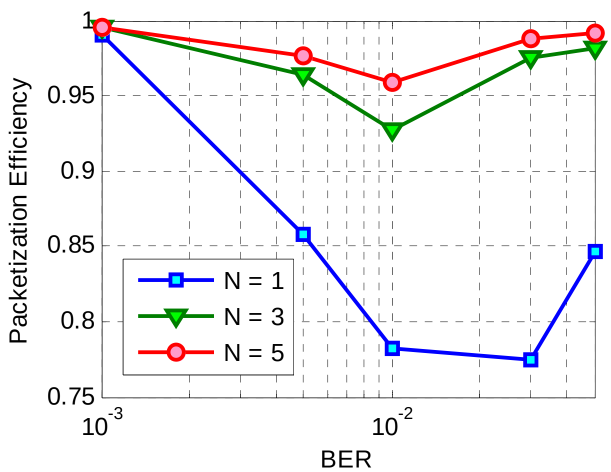


Figure 4: BER versus Bluetooth packet efficiency

5. VIDEO STREAMING EXPERIMENTS

While the previous experiments were conducted with the Simulink simulator of Matlab, more detailed simulation of Bluetooth was performed with the University of Cincinatti 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.

In Bluetooth systems, the wireless channel is usually taken to be frequency non-selective and wide-sense stationary. A Gilbert-Elliott [19][20] two state discrete-time, ergodic Markov chain modelled the wireless channel states between a Bluetooth master and slave node. By adopting this model it is possible to simulate burst errors of the kind that cause problems to an ARQ mechanism. The mean duration of a good state, T_g, was set at 2 s and in experiments the mean duration of the bad state, T_b, was varied according to the ratio of bad to total duration, which ratio is designated as *P*. In units of the Bluetooth time slot duration, T_g= 3200, which implies from:

$$T_g = \frac{1}{1 - Pgg}, \qquad (18)$$

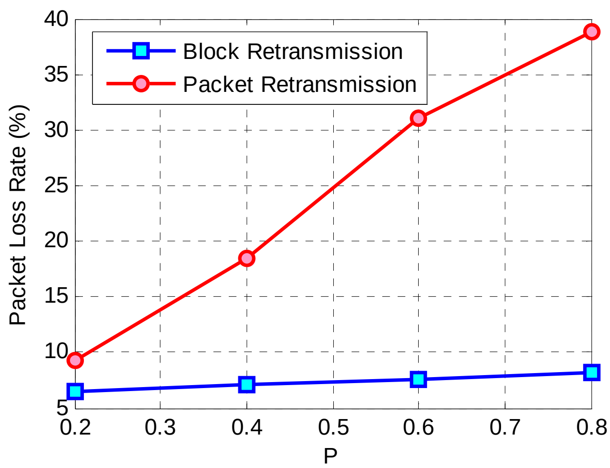
that, given the current state is good (g), Pgg, the probability that the next state is also g , is 0.9996875. At a Bluetooth gross air rate of 2.0 Mb/s, the BER during a good state was set to 2×10^{-4} and during a bad state to 5×10^{-3}, which is equivalent to CNRs of 10.34 dB and 7.65 dB. Thus, the CNRs are appropriate to a 2DM5 packet type regime. Notice that BERs in the two states are independent of the state transition behaviour.

Simulations were carried out with input from an MPEG-2 encoded bitstream at a mean rate of 0.9 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 (366×288 pixels) with a GOP structure of N = 12, and M = 3. Error concealment was by simple previous frame replacement. In [21], fully filled Bluetooth packets were formed using maximal bandwidth time-slot packets, regardless of slice boundaries. While this results in some loss in error resilience, as each MPEG-2 slice contains a decoder synchronization marker, in [21] it is shown that the overall video performance is superior to choice of smaller packet sizes to preserve slice boundaries.

For simplicity of interpretation, packets were treated of equal priority, though an extension of this work would consider the impact of picture type upon packet priority. The display deadline, effectively set by the size of the playout buffer at the receiving device, was 230 ms. However, to ensure safe arrival, this was robustly and conservatively adjusted [22] by a factor of 0.9.

In an implementation, the video complexity (motion, scene changes, and texture), together with the buffer size, and possible cross traffic, all make it difficult to estimate how many packets to accumulate blocks in error from. Therefore, a lazy evaluation technique is applied, whereby the blocks are accumulated and sent when all retransmitted blocks in a packet are close to their deadlines. In the simulations, the buffer size was set to fifty, which is a typical size, cf. [23].

In Fig. 5, block retransmission and packet retransmission are compared when the retransmission limit is set to one. Block-based transmission was defined in Section 4.1, *i.e.* FEC is applied to blocks of *b* bits, with the blocks forming a packet. The same block size as in Section 4.2 was applied to the arriving video stream data, *i.e.* 45 bits per block. The same packet size, with or without per-block FEC, as in Section 4.2, *i.e.* 2DM5 was also selected. The proportion of time allocated to the bad state was increased, resulting in ratio *P* increasing. It is apparent that there is a large difference in packet losses between the two schemes. This is because, as full packets are resent under the packet retransmission scheme, there is a greater risk of error to those packets. Buffer overflow is also more likely under packet retransmission, because the longer retransmitted packets cause greater propagation delay. Increasing the buffer size leads to missed display deadlines, whereas in these simulations effective packet loss through missed deadlines did not occur.



<u>Figure 5:</u> Packet loss under packet and block retransmission, with varying ratio of bad state channel

In Fig. 6, the video quality is plotted in terms of Peak Signal to Noise Ratio (PSNR). While the quality remains high under block-based HARC, under packet retransmission the quality declines to unacceptable levels below 30 dB as the channel conditions worsen. The relative power efficiency is plotted in Fig. 7. This Figure confirms that not only is the video quality considerably improved, despite the deteriorating channel but the effectiveness of transmission also follows a similar trend. In other words block retransmission sends fewer bits for higher quality video at the receiver.

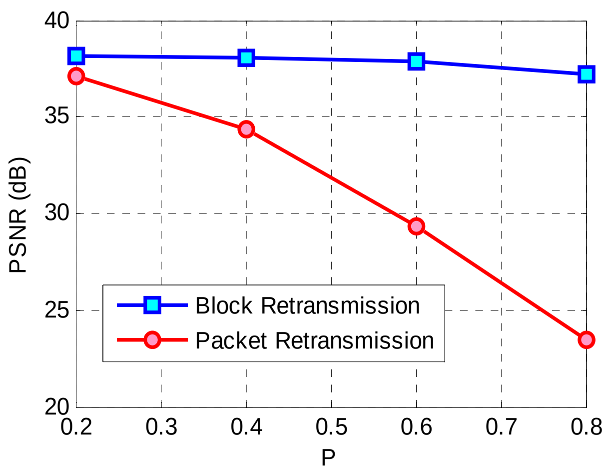


Figure 6: Video quality under packet and block retransmission, with varying ratio of bad state channel

6. CONCLUSION

Type 1 Hybrid ARQ is a relatively simply technique to allow a trade-off between different wireless channel conditions. When combined with block-based retransmission it results in power savings because data transmission is made more effective. Because packet loss is reduced, for video streaming, there is also a marked improvement in video quality over a packet retransmission scheme. This was demonstrated for Bluetooth interconnects, for which, because fast ARQ is obligatory, the scheme has significant advantages. The paper has shown the advantage of introducing two new packet types into the Bluetooth scheme, with minimal implementation cost as many of the necessary features already exist. This will help to maintain Bluetooth as a low power competitor to other personal area networks.

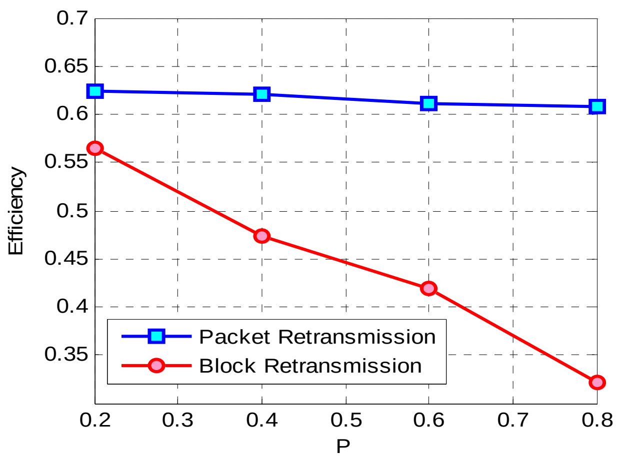


Figure 7: Power efficiency under packet and block retransmission, with varying ratio of bad state channel

ACKNOWLEDGMENT

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

REFERENCES

[1] Zhang, Q., Zhu, W. and Zhang, Y.-Q., "End-to-End QoS for Video Delivery Over Wireless Internet", Proc. of the IEEE, vol. 93, no. 1, pp. 123-134, 2005.

[2] Haartsen, J., "The Bluetooth Radio System", IEEE Personal Comms.,vol. 7, no. 1, pp. 28-26, 2000.

[3] Specification of the Bluetooth System -- 2.0 + EDR, 2004, Available online at http://www.bluetooth.com.

[4] Agrawal, P., Chen, J.-C., Kishore, S., Ramanathan, P., and Sivalingam, K., "Battery Power Sensitive Video Processing in Wireless Networks", IEEE Int. Symposium on Personal, Indoor, and Mobile Radio Comms., vol. 1, pp. 116-120, Sept. 1998.

[5] Chen, J.-C., Sivalingam, K.M., Agrawal, P. and Kishore, S., "A Comparison of MAC Protocols for Wireless Local Networks based on Battery Power Consumption", IEEE INFOCOM, vol. 1, pp. 150-157, March 1998.

[6] Ghanbari, M., *Standard Codecs: Image Compression to Advanced Video Coding*, IEE, London, UK, 2003.

[7] Caire, G. and Tuninetti, D., "The Throughput of Hybrid-ARQ Protocols for the Gaussian Collision Channel", IEEE Trans. on Inform. Theory, vol. 47, no. 5, pp. 1971-1988, July 2001.

[8] Lampe, L., Jain, M. and Schober, R., "Improved Decoding for Bluetooth Systems", IEEE Trans. on Comms., vol. 53, no. 1, pp. 1-4, 2003.

[9] Conti, A., Dardari, D., Pasolini, G. and Andrisano, O. "Bluetooth and IEEE 802.llb Coexistence: Analytical Performance Evaluation in Fading Channels", IEEE J. Sel. Areas Commun., vol. 21, pp. 259–269, Feb. 2003.

[10] Tanaka, H. and Sakakibara, K., "Performance of Reed-Solomon Coded Type I Hybrid ARQ Scheme on Fading Channels", IEEE GLOBECOM, vol. 3, pp. 2148 – 2152, Nov 1996.

[11] Golmie, N., Chevrolier, N. and Rebala, O., "Bluetooth and WLAN Coexistence: Challenges and Solutions", IEEE Wireless Communications, vol. 6, pp. 22-29, 2003.

[12] Zorsi, M. and Rao, R. R., "Error Control and Energy Consumption in Communications for

Nomadic Computing", IEEE Trans. On Computers, vol. 46, no. 3, pp. 279-289, March 1997.

[13] Letteri, P., Fragouli, C., and Srivastava, M. B., "Low Power Error Control for Wireless Links", ACM/IEEE MobiCom, pp. 139-150, 1997.

[14] Zhao, S., Xiong, Z. and Wang, X. "Joint Error Control and Power Allocation for Video Transmission over CDMA Networks with Multiuser Detection", IEEE Trans. On Circuits and Systems for Video Technology, vol. 12, no. 6, pp. 425-437, June 2002.

[15] Ferro, E. and Potorì, F., "Bluetooth and Wi-Fi Wireless Protocols: A Survey and a Comparison", IEEE Wireless Communications, vol.12, no. 1, pp. 12-26, Feb. 2005.

[16] Valenti, M. C., Robert, M. and Reed, J. H., "On the Throughput of Bluetooth Data Transmissions", IEEE Wireless Commun. and Networking Conf., pp. 119-123., March 2002.

[17] Blahut, R. E., *Theory and Practice of Error Control Codes*, Addison-Wesley, Reading, MA, 1983.

[18] Lin, S. and Costello, D. J., *Error Control Coding*, 2nd edition, Pearson Educ., London, UK, 2004.

[19] Gilbert, E. N., "Capacity of Burst-Noise Channel", Bell System Technical J., vol. 39, pp. 1253-1265, 1960.

[20] Elliott, E. O., "Estimates of Error Rates for Codes on Burst Noise Channels", Bell System Technical J., vol. 42, pp. 1977-1997, 1963.

[21] Razavi, R., Fleury, M., and Ghanbari, M., "An Efficient Packetization Scheme for Bluetooth Video Transmission", Electronic Letters, vol. 42, no. 20, pp. 1143-1145, 2006.

[22] Jurca, D. and Frossard, P., "Packet Media Streaming with Imprecise Rate Estimation", Advances in Multimedia, 2007.

[23] Li, Q., and van der Schaar, M., "Providing QoS to Layered Video Over Wireless Local Area Networks Through Real-Time Retry Limit Adaptation", IEEE Transactions on Multimedia, vol. 6, no. 2, pp. 278-290, 2004.

Address of Corresponding Author

Dr. M. Fleury, Electronic Systems Engineering Department, University of Essex, Wivenhoe Park, Colchester, Essex , CO4 3SQ, United Kingdom.