Deadline-Aware Video Delivery in a Disrupted Bluetooth Network

R. Razavi
ESE Dept., University of Essex
Colchester CO4 3SQ, U.K.
rrazav@essex.ac.uk

M. Fleury
ESE. Dept., University of Essex
Colchester CO4 3SQ, U.K.
fleum@essex.ac.uk

M. Ghanbari
ESE Dept., University of Essex
Colchester CO4 3SQ, U.K.
ghan@essex.ac.uk

Abstract

Adaptive ARQ is applied to a wireless network, altering retransmission policy according to content importance. By a video deadline-aware buffer discard policy, repeated transmission of expired packets and buffer overflow are mitigated, helping to maintain video quality. Multiple deadlines are applied according to picture type. The combined policy is demonstrated for a Bluetooth piconet achieving fast recovery after disruption.

INTRODUCTION

Wireless networks are susceptible to interference and other disruptions, which have a severe impact on delay-sensitive services such as video streaming. There exist two deadline types for encoded video: 1) a display deadline, herein set by the playout buffer size; and 2) a decode deadline relative to the duration of a Group of Pictures (GOP). Packets forming an anchor frame (I- and P-picture) are still of value in retroactive decoding of subsequent P- or B-pictures [7]. However, if disruption is prolonged retained packets contribute to increasing buffer fullness, which may lead to excessive delay and, in severe cases, to send buffer overflow. These circumstances lead to a need for a deadline-aware buffer (DAB) [8]. In the face of disruption, the DAB discards packets according to content importance. If Automatic Repeat Request (ARQ) is available, and in Bluetooth (B/T) [6] (IEEE 802.15.1) it comes for free by virtue of slave polling, then it too should be content adaptive.

In this paper, DAB and adaptive ARQ are combined to allow fast recovery from disruption, which in our examples takes the form either of prolonged error bursts across a single B/T interconnect or while dynamic reconfiguration of a B/T piconet [12] occurs. Dynamic reconfiguration is applied to maximize overall piconet throughput. DAB with adaptive ARQ work together to avoid late packet arrivals. Otherwise, in conditions of heavy traffic and/or high bit error rates (BER), as mentioned, the send buffer may overflow. The combination of the two techniques in a form that is relatively simple to implement on a B/T piconet is the paper's main contribution.

DEADLINE-AWARE BUFFER

For reasons of error resilience, encoded video is transmitted as a repeating sequence of Group of Pictures (GOP), with the start of each GOP formed by an Intra-coded or I-picture. An I-picture is the basis for prediction of all other pictures in the GOP (usually 12 to 15 pictures in all) and, hence, its loss has drastic consequences for all other pictures in the GOP. Other predictive pictures or P-pictures also are essential for the reconstruction of some bi-directionally predicted or B-pictures within the GOP, though, in some cases, anchor pictures retained in the decode buffer can form the basis for error concealment. Lastly, the third type of picture, the B-picture, has no predictive value but is dependent on macro-blocks from adjacent anchor pictures. The picture type is identifiable through the bit-stream header without decoding.

In the conservative sender buffer discard policy of this paper, all packets of whatever picture type have a display deadline which is the size of the playout buffer expressed as a time beyond which buffer underflow will occur. In a conservative policy, in which there is no need for playout buffer fullness updates, the deadline is set as the maximum time that the playout buffer can delay the need for a packet. Playout buffers are normally present to smooth out jitter across a network path (if the B/T master was also an access point) and in this paper the size is assumed to be constant. In the discard policy, the propagation delay is assumed (optimistically) to be constant without retransmissions.

In addition to the display deadline, all I-picture packets have a decode deadline, which is the display time remaining until the end of the GOP. Thus, for a 12 frame GOP, this is the time to display 11 frames, i.e. 0.44 s at 25 frame/s. For P-picture packets, the time will vary depending on the number of frames to the end of the GOP. For B-pictures the decode deadline is set to zero. The decode deadline is added to the display deadline and a packet is discarded from the DAB after its total deadline expires. By storing the GOP end time, an implementation performs one subtraction to find each decode deadline. Account has been taken of I- B- P-picture reordering, Fig. 1, at encode and send buffer output, which has an effect on buffer fullness. Reordering is introduced to ensure that reference pictures arrive and can be decoded before the dependent B-pictures.

ADAPTIVE-RETRY

B/T employs variable-sized packets up to a maximum of five frequency-hopping slots of 625 μs duration. Due to the risk of RF noise, by default, Bluetooth provides repeat packet transmissions, if a packet is not acknowledged within a pre-set time. For every B/T packet transmitted from a B/T traffic source, the receiver replies with a packet occupying at least one slot. Therefore, a single-slot packet serves for a link layer stop-and-go ARQ message whenever a corrupted packet payload is detected.

An ARQ under B/T may occur in the following circumstances: a) failure to synchronize on the access header code; b) header corruption detected by a triple redundancy code; c) payload corruption detected by CRC; d) failure to synchronize with the return packet header; e) header corruption of the return packet.

The default value of the ARQ retransmission timeout (RTO) in most B/T chipsets is set to infinity. On general grounds, this is unwise in conditions of fast fading caused by multi-path echoes, as error bursts occur. In [2], a fixed RTO and an adaptive RTO were considered. The disadvantage of a fixed RTO is that it is difficult to arrive at a value that avoids either excessive delay or excessive packet drops in *all* circumstances. The adaptive RTO, which was upper and lower- bounded, was based in [2] on a smoothed round-trip time (srtt). The RTO was adapted downwards or upwards if the new rtt respectively is less than or more than the previous srtt.

In our experiments, the ARQ RTO is adaptively selected in terms of number of retransmissions allowed, to avoid further delay after the packet enters the tail of the B/T FIFO send buffer. A threshold is set that is the maximum number of retransmissions allowed when the buffer is empty. The maximum number of retransmissions is subsequently changed by a factor depending on the buffer fullness reported by the B/T module, which in [11] was found to be more appropriate a measure than packet loss or delay. The formula employed is summarized as

$$N = round\left(\frac{m.(c-f)}{c}\right), \qquad (1)$$

where N is the maximum number of retransmissions allowed -- the RTO, m is the maximum number of retransmissions allowed when the buffer is empty, f is the number of packets buffered in the send buffer (buffer fullness), and c is the buffer capacity (set to 50 in the simulations). The operator *round* returns the nearest integer.

According to (1), the maximum number of retransmissions allowed is a function of buffer fullness. Figure 2 plots this function when m = 2, 3, and 5, and $c = 50$ packets. When the buffer is empty, f = 0, then the maximum number of retransmissions occurs, whereas when the buffer approaches full occupation no retransmissions may occur. The smaller the value of m the sooner this latter event occurs.

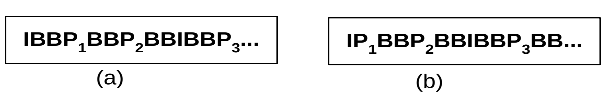

Figure 1. I- B- P- picture reordering: a) display order b) send buffer output order.

We adapt the value of m to the type of picture being transmitted. If the picture is of I-type, upon which all other pictures in the GOP depend, then m is set to five. Similarly for P- and B-pictures, m is respectively set to three and two.

In fact, P-frames are not uniform, as they may contain intra-coded macro-blocks (ones not formed predictively) and inter-coded macro-blocks. A significant number of intra-coded macro-blocks may indicate a scene change, revealed objects, a camera zoom/pan, or high motion but is encoder dependent. The significance varies with sequence type. For simplicity of result presentation, in this paper, the picture type priority scheme is not extended in this way.

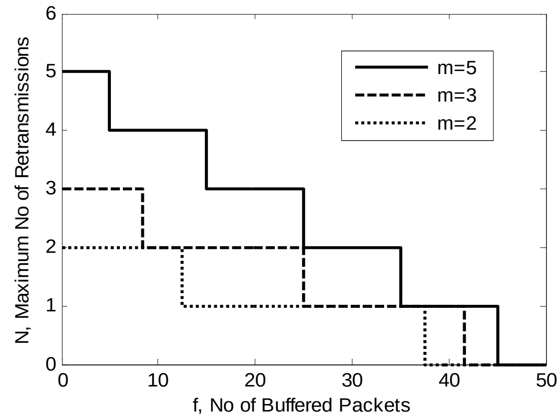

Figure 2. Maximum number of retransmissions, RTO according to sender buffer fullness.

The IEEE 802.11 Point Coordination Function (DCF) employs centralized polling, as in B/T. For example, [9] applied adaptive ARQ on an IEEE 802.11a network, but based on a noise level estimate, which does not reliably detect [2] volatile channel behavior. In [3], the less suitable for video but more common IEEE 802.11's Distributed CF (DCF) was used with a retry limit based on an encoder determined error propagation factor. The latter limits the flexibility, as it requires a specialist encoder.

METHODOLOGY

We assume Bluetooth version 2.0 with Enhanced Data Rate (EDR), which has gross air rates of 2.0 Mbps and 3.0 Mbps, compared to the basic rate of 1.0 Mbps of version 1. The research employed the University of Cincinatti B/T (UCBT) extension[1] to the well-known ns-2 network simulator (v. 2.28 used). All links were set at the maximum EDR 3.0 Mbps gross air rate. Simulation runs were each repeated twenty times and the results averaged to produce summary statistics.

A data frame across a B/T link in asymmetric mode consists of an Asynchronous Connection-Less (ACL) packet occupying one, three or five time slots. Unfortunately, if packetization takes place on a single MPEG-2 slice (one row of macro-blocks) per B/T packet the result is partially filled packets as well as many 1- or 3-slot packets, with a consequent drop in throughput. Therefore, in [10] fully filled B/T packets were formed, regardless of slice boundaries. While this results in some loss in error resilience, as each MPEG-2 slice contains a decoder synchronization

[1] A download is available from `http://www.ececs.uc.edu/~cdmc/ucbt`

marker, in [10] it is shown that the overall video performance is superior. In our experiments, the video B/T packet type was set to 3DH5, which corresponds to a five time-slot packet. The user payload is 1021 B with an asymmetric maximum bit rate of 2.1781 Mbps.

A Gilbert-Elliot (G.-E.) [4] two-state, discrete-time, ergodic Markov chain modeled the wireless channel error characteristics between a B/T master and slave node. By adopting this model it was possible to simulate non-independent burst errors of the kind that cause problems to an ARQ mechanism. Though B/T v.1.2 adopted an Adaptive Frequency Hopping scheme (AFH), G.-E. is still used herein to model the channel, because AFH is of limited benefit to audio/video applications [5], especially when interference occurs across the unlicensed 2.4 GHz ISM band. The mean duration of a good state, T_g, was set at 2 s and in a bad state, T_b, was set to 0.5 s. In units of the B/T time slot duration, T_g= 3200 and T_b= 800, which implies from:

$$T_g = \frac{1}{1 - Pgg}, T_b = \frac{1}{1 - Pbb} \quad (1)$$

that, given the current state is g, ¸Pgg, the probability that the next state is also g , is 0.9996875 and Pbb, given the current state is b¸ the probability that the next state is also b, is 0.99875. At 3.0 Mbps, the Bit Error Rate (BER) during a good state was set to 10^{-5} and in a bad state to 10^{-4}.

The simulations were carried out with input from an MPEG-2 encoded bit-stream for a 40 s video clip with moderate motion. The display rate was 25 frame/s, 1000 frames in all. The source video was Common Intermediate Format (CIF)-sized (366×288 pixels) with a GOP structure of N = 12, and M = 3 (M is the number of pictures from the I-picture to the first P-picture, i.e. including two B-pictures). To avoid consideration of a variety of decoder options, error concealment was confined to simple, previous frame substitution [7], which is a common assumption.

EXPERIMENTAL RESULTS

Two types of disruption were modelled in separate scenarios. In the first, a single master-to-slave interconnect was modelled with the G.-E. channel model already described. The display deadline formed by the B/T buffer was set at 0.2 s.

In the second scenario, a more severe form of disruption occurs while the B/T piconet reconfigures. In a B/T piconet [6], a master can communicate with up to seven slaves but no direct slave-to-slave communication is possible. Therefore, all such communication takes place over two hops via a master, effectively doubling the bit-rate over the shared channel. When the bit rate over the slave-to-slave connection passes a threshold it becomes worthwhile to reallocate the role of master to one of the communicating slaves. On request from a slave, a master broadcasts its intention to exchange its role with a slave, and after subsequent implemented exchanges the slave-to-slave communicating pair become master-to-slave. Reconfiguration takes between 63 and 200 ms [1], and is conservatively taken to be 200 ms in our experiments.

In Fig. 3, slave S1 sends the MPEG-2 video encoded at an average rate of 500 kbps to slave S2. A CBR traffic source (800 kbps, packet size of 800 B) at master M transmits data to slave S3. At a given time of 3.82 s the CBR rate decreases to 400 kbps, justifying dynamic role swapping. S1 becomes the new master directly connected to slave S2, while former master M becomes slave S1 communicating via the new master to slave S3. To avoid complicating the interpretation, in scenario 2, the channel is assumed to be error free. The display deadline was set to 0.15 s.

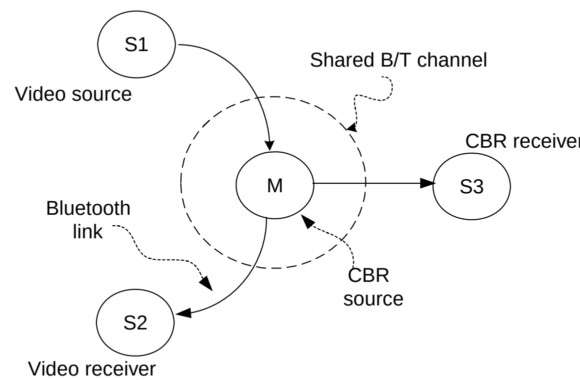

Figure 3. Scenario 2 with B/T piconet before dynamic reconfiguration.

Scenario 1 results

Figures 4, 5 and 6 compare the received video quality, Peak Signal to Noise Ratio (PSNR), resulting from the ARQ alternatives without and with DAB. Compared to Figs. 5 and 6, it is clear that it is unwise to turn off ARQ in the channel conditions resulting from the G.-E. model. While the PSNR level in Figs. 5 and 6 approaches 40 dB (high-quality video), there are repeated severe drops in quality, when the channel enters a ‘bad’ state. What is apparent is that the number and severity of the drops is reduced under the adaptive ARQ with DAB scheme.

Summary results are presented in Table 1. If the B/T channel is underutilized, as it is in this scenario, as will become apparent from a consideration of buffer fullness, then not using ARQ when a slot is available for this purpose is unwise. However, if the piconet were to be heavily loaded and/or the BER were to be high then ARQ with infinite retransmissions would also be unwise, as queues build up in send buffers, while a packet is repeatedly re-sent, causing extensive delays and buffer overflow. Adaptive ARQ is able to adjust towards either of these extremes. Fig. 7 compares the impact on B/T packet delay of each ARQ scheme. Though the no ARQ option results in limited delay, Table 1 shows that there are heavy packet losses over the channel. When infinite retry is applied, during a typical bad state, packets continue to be transmitted despite the expiration of their deadline, as the zoomed-in inset emphasizes.

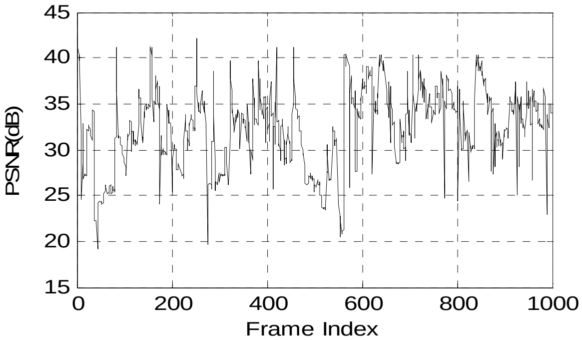

Figure 4. Scenario 1: PSNR with no ARQ.

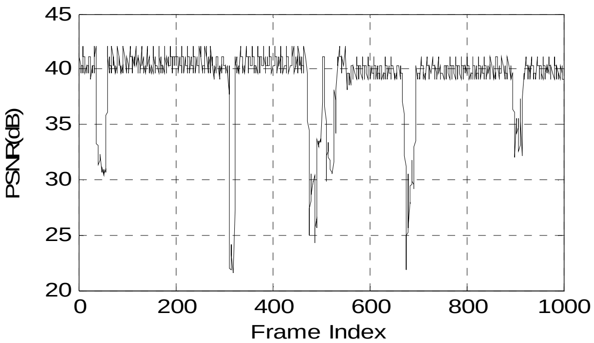

Figure 5. Scenario 1: PSNR with default ARQ.

However, with the DAB in place, then packets that pass their deadline are removed from the buffer, which explains why few packets have delays greater then 0.2 s. Those few that do are I- and P-picture packets. From Fig. 8, it is apparent that in scenario 1, there are few occasions when the send buffer becomes full, under the default B/T ARQ. However, these occasions lead to packet loss, while adaptive ARQ avoids that risk. Under favorable conditions, adaptive ARQ with DAB behaves similarly to default ARQ, but avoids packet loss through congestion.

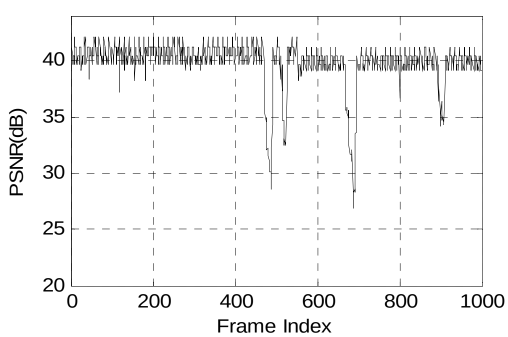

Figure 6. Scenario 1: PSNR with adaptive ARQ & DAB.

Scenario 2 results

During a role change the gain from including the DAB is very apparent. Of course, at other times, as the channel is error free, ARQ is not needed. In Fig. 9, because packets that have passed their deadlines are expelled from the send buffer, the scheme with DAB is able to restore video quality more quickly because it does not have to resend redundant packets. In Fig. 10, as in Fig. 7, few packets in the scheme with DAB experience delay beyond the deadline, here 0.15 s. The effect on buffer fullness is shown in Fig. 11, where it is apparent that only the default ARQ scheme will suffer from buffer overflow at the sender.

Table 1. Scenario 1: Video quality at the receiver according to ARQ scheme.

ARQ scheme	Mean PSNR	Packet Loss (%)
Adaptive ARQ /DAB	39.88	4.3
Infinite ARQ	38.74	7.4
No ARQ	33.24	20.7

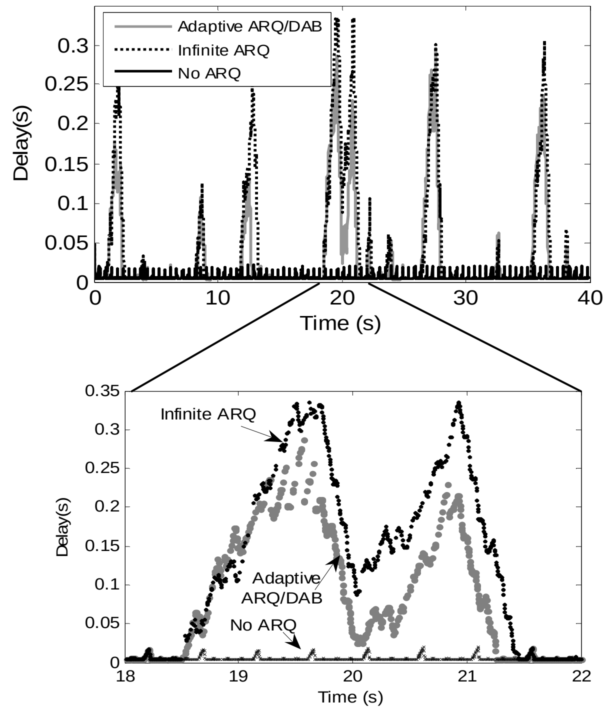

Figure 7. Scenario 1: B/T packet delay over time.

Consider the case of a longer period of disruption than occurs in Figs. 9-11. Then delay under the B/T default ARQ scheme would continue to grow. However, under the scheme with DAB scheme no alteration in behavior would occur. Moreover, had the piconet been more heavily loaded then buffer overflow and duration of video quality degradation would be more marked than occurs in Fig. 10. It would take longer for the sender to reduce its queue length to a level when deadlines were kept. The larger the send buffer size (beyond 50 packets herein), the longer this process would take.

CONCLUSION

This paper proposed a deadline-aware send buffer combined with adaptive ARQ to counter periods of network disruption. For video traffic, the scheme is packet content importance aware and multiple deadlines are applied. The example scenarios illustrate behavior under comparatively light piconet loading. The advantage would be even more marked had loading been higher or disruption been longer. In effect, delay is bounded by adaptive ARQ with DAB.

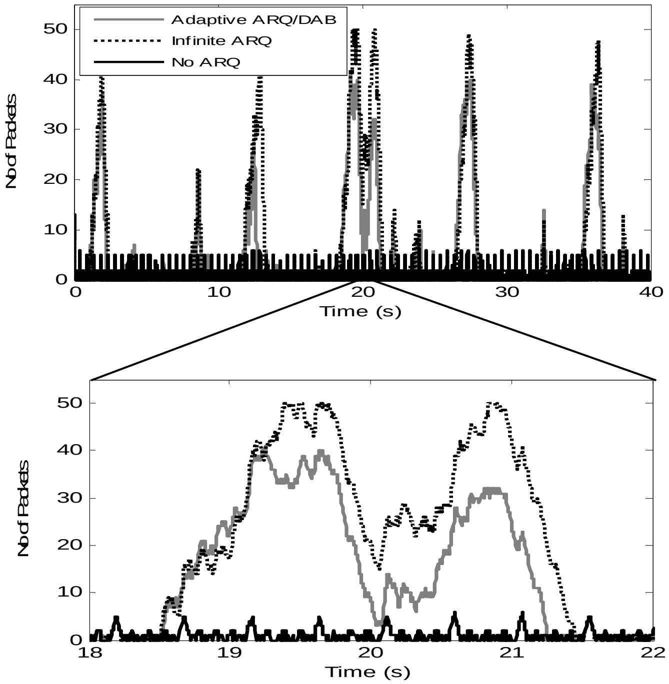

Figure 8. Scenario 1: Send buffer fullness over time.

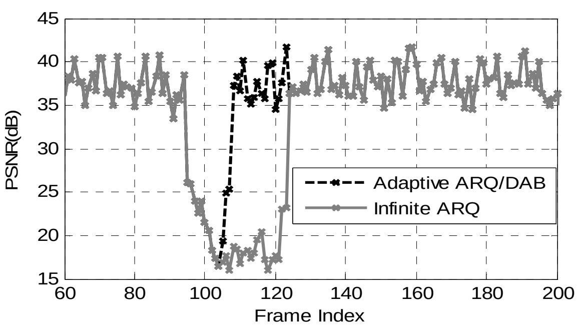

Figure 9. Scenario 2: PSNR with default ARQ and adaptive ARQ/DAB schemes.

REFERENCES

[1] Aghyankar, S., et al. "On the Application of Traffic Engineering over Bluetooth ad hoc networks", *ACM Int. Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, pp. 116-123, Sept. 2003.

[2] Chen, L.-J., Kapoor, R., Jung, S., Sanadidi, M.Y., and Gerla, M. "Audio Streaming over Bluetooth: An Adaptive ARQ Timeout Approach", 2005, *Inderscience Int. J. of Wireless and Mobile Comp.*, vol. 1, no. 5, 2005.

[3] Chen, C.-M.,Lin, C.-W., Chen, Y.-C., "Packet Scheduling for Video Streaming over Wireless with Content-Aware Packet Retry Limit", *IEEE Int. Workshop on Multimedia Signal Processing*, pp. 409-414, Oct. 2006.

[4] Elliott, E.O., "Estimates of Error Rates for Codes on Burst-Error Channels", *Bell Syst. Tech. J.*, vol. 42, pp. 1977-1997, 1963.

[5] Golmie, N., Chevrollier, N. and Rebala, O., "Bluetooth and WLAN Coexistence: Challenges and Solutions", *IEEE Wireless Comms.*, vol. 10, no. 6, pp. 22-29,2003.

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

[7] Kalman, M., Ramanathan, P., and Girod, B., "Rate-Distortion Optimized Video Streaming with Multiple Deadlines", *Int. Conf. on Image Processing*, vol. 3, pp. 662-664, Sept. 2003.

[8] Kang, S. H. and Zakhor, A., "Packet Scheduling Algorithm for Wireless Video Streaming", *PacketVideo Workshop*, Apr. 2002.

[9] Krishnmachari, S., Schaar, M., Choi, S. and Xu, X., "Video Streaming over Wireless LANs: A Cross-layer Approach", *PacketVideo Workshop*, Apr. 2003.

[10] 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.

[11] Razavi, R., Fleury, M. and Ghanbari, M., "Detecting Congestion within a Bluetooth Piconet: Video Streaming Response*", London Comms. Symp*., pp. 181-184, 2006.

[12] Zhang W., Zhu, H., and Cao, G., "On Improving the Performance of Bluetooth Networks through Dynamic Role Management", *Tech. report, CSE-01-018 Penn State University*, 2001.

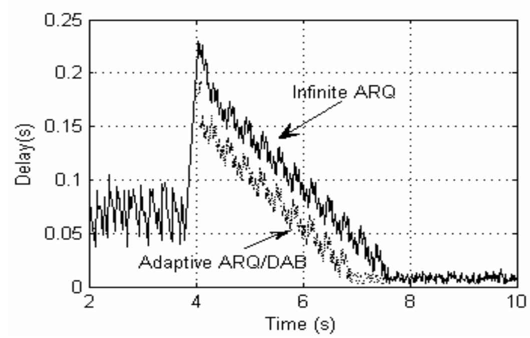

Figure 10. Scenario 2: Delay under default ARQ and adaptive ARQ/DAB schemes.

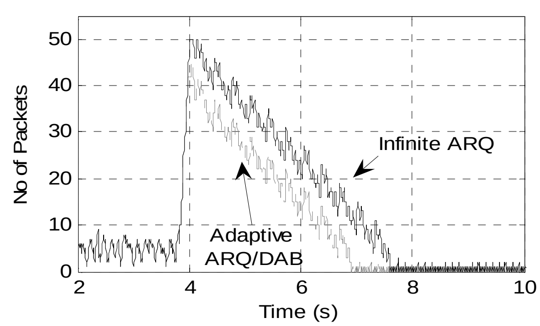

Figure 11. Scenario 2: Buffer fullness under default ARQ and adaptive ARQ/DAB schemes.