Transport Level Optimisations for Interactive Media Streaming Over Wide-area Wireless Networks

Julian Chesterfield[1], Rajiv Chakravorty[1], Suman Banerjee[2], Pablo Rodriguez[3], Ian Pratt[1] and Jon Crowcroft[1]

[1] Cambridge University Computer Laboratory, Cambridge CB3 0FD, UK
{firstname.lastname}@cl.cam.ac.uk,
[2] University of Wisconsin, Madison WI 53706, USA
suman@cs.wisc.edu
[3] Microsoft Research,
Cambridge CB3 0FD, UK
pablo@microsoft.com

Abstract. Wide-area cellular data networks (e.g. GPRS, CDMA 2000) are gaining popularity to provide "always-on" data connectivity to mobile users. However, the characteristics of the wide-area wireless links present several new challenges to different applications. In this paper we focus on optimisation techniques to improve the application performance of streaming media systems operating over wide-area wireless networks. Through detailed measurements in commercial wide-area cellular networks we generated traces to evaluate the performance of our proposed mechanisms. We make three contributions in the context of media streaming over wide-area wireless links: (1) perform a detailed analysis of the channel characteristics, (2) identify general optimisation techniques to reduce jitter for streaming applications, and (3) define a scheme to provide Unequal Error Protection (UEP) for multimedia streams. Due to deployment constraints inside the cellular service provider's network, our scheme defines an application-level inferencing technique that adapts itself to the characteristics of the underlying wireless link.

1 Introduction

Wide-area wireless networks (WWAN) are being deployed throughout the world to provide ubiquitous access to IP-based applications. Data technologies in this environment, e.g. GPRS, are currently being optimised for bursty data with strict reliability requirements such as HTTP traffic. Adaptations that work well for such bursty, reliable data, however lead to extreme variations in the packet inter-arrival times. For example, the link propagation delay over some of the commercial GPRS networks for the same size packet varies between 70 to 700 milliseconds.

Multimedia applications are generally not designed to cope with such extreme delay fluctuations and hence perform poorly in these environments. Unlike HTTP traffic, multimedia streams do not require perfect reliability. Instead they require predictable and bounded jitter on the end-to-end paths for an acceptable user experience which can be traded off with (some) data losses.

In this paper we define techniques to adapt the link layer characteristics leading to improved performance for multimedia streams. While these techniques can be most efficiently implemented in the link layer itself, such adaptation would require installing these mechanisms and network elements inside the cellular provider's network, and depends on the willingness of the latter to allow such changes. Therefore, in this paper we take an alternative approach whereby application-level inferencing and encoding mechanisms are used to effect similar changes to the link layer performance. Our optimisation mechanism first infers the packetisation structure used by the Radio Link Control (RLC) layer and subsequently uses this information to perform more intelligent packet framing. Additionally we carefully extract uncorrupted portions of individual data packets to improve the link-level performance in WWAN environments.

This paper, therefore, focuses on the following aspects of multimedia streams over WWAN links: (1) optimise commercial WWAN networks for improving the performance of multimedia streams, (2) identify techniques that reduce levels of jitter and at the same time maintain some usable levels of reliability, and (3) describe a deployment path that requires no changes to existing networks of the cellular service providers. This *end-to-end* solution works without assistance of proxies or any other interposed network elements. Hence this approach is a realistic solution to enable end-to-end interactive multimedia streaming where the last hop access is provided and controlled by commercial WWAN operators, and any change in the internal network elements is generally hard to instigate.

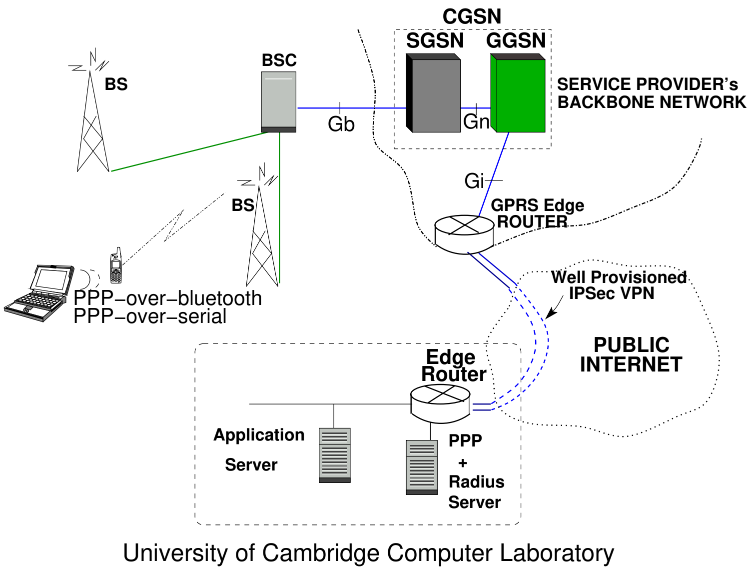

Figure 1. The experimental testbed for conducting the GPRS tests. A well provisioned, low latency tunnel transfers all Computer Laboratory GPRS device traffic directly to our PPP server, to which the application server is directly connected via a 100Mbit/s ethernet segment.

We begin with some link characterisation results in section 2 which demonstrate that by disabling reliability at the link layer we are able to reduce the packet inter-arrival jitter by a factor of three, at the expense of a marginal increase in packet losses due to errors. However, trace based analysis of the same link tests in section 3 indicates that by designing our encoding scheme to operate efficiently across the block structure of the RLC scheme we need to add a data redundancy of only 13% to recover more than 90% of corrupt frames.

2 Network Characterisation of GPRS

GPRS links use two different schemes at the link-layer to provide reliability over a wide range of channel noise conditions. These are FEC and ARQ that work aggressively to recover any data transfer losses and provide in-order, reliable abstraction to the higher layers (like IP). Thus, higher layers will detect losses only due to (1) deep fading that leads to bursty losses, or (2) cell-reselection due to the cell update procedure that leads to "black-outs."

Application data (IP) packets are fragmented into blocks of Radio Link Control (RLC) data and transmitted as GSM bursts in a *multiframe* construct [1]. A multiframe consists of 12 'blocks', each of which is further subdivided into 4 TDMA frames. Therefore, when a client is assigned a GPRS 'channel', it receives one TDMA slot (a burst of 456 encoded bits) in each block, or 12 slots in a multiframe. Note that this division of data transmission is important in our analysis as it impacts the packetisation approach we present in Section 3.

Due to the high occurrence of data errors over cellular links, current GPRS networks typically apply CS-1 or CS-2 encoding over the raw channel data. Additionally, RLC provides aggressive ARQ retransmission of the RLC blocks, typically up to 7 attempts, that are received with errors. The combined schemes (FEC and ARQ) provide a highly reliable link, but increase round trip times (RTT) making performance quite variable. To understand the effects of link level retransmissions, we conducted certain tests to measure the channel throughput.

2.1 Experimental Setup

Figure 1 indicates the experimental testbed used in the measurements. The mobile device connects to the provider GPRS service through the local Base Station Controller (BSC). All traffic from registered Cambridge Computer Laboratory devices is then tunnelled directly from the GGSN to the Lab PPP server via a well provisioned VPN link. We created a customised application that could measure the available throughput by sending UDP packets at a rate just higher than the link capacity. Back-to-back

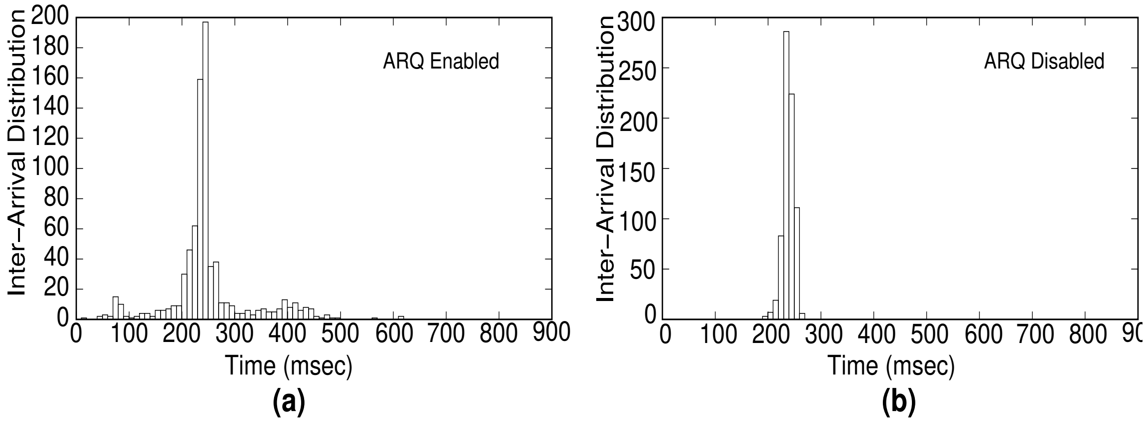

Figure 2. The inter-arrival time distribution of packets (1000 packet samples, 1371 bytes size) with (a) ARQ enabled and (b) ARQ disabled. The inter-arrival time distribution is much tighter when ARQ is disabled, when compared to the case when ARQ is enabled.

packets are buffered by Base Stations and forwarded to the mobile device at the wireless link rate until the buffer is depleted.

All tests were conducted via a stationary GPRS client in one location using a provider that typically assigns static channel resources per GPRS mobile device. Consequently we can be certain that the variability in inter-packet arrival times is caused by link conditions, such as ARQ, rather than dynamic channel assignment. The measurements of signal quality during the tests indicated that the values varied between -91dBm and -63dBm, with Bit Error Rate (BER) measurements from the mobile device ranging from 0 - 3%. Typically, readings below -77dBm are considered to indicate an increased probability of errors in the signal decoding causing a higher occurrence of retransmissions.

2.2 Experimental Results

Figure 2(a) presents a distribution of interpacket arrival times for a 1000 sample test. The results indicate that a significant number of retransmissions occurred, identified here by the high variation in inter-arrival times. The extended transmission times measured around the 400 to 600ms timeframe indicates that some RLC blocks were retransmitted during subsequent multiframe transmissions. The shorter transmission latencies identified around the 80ms value typically follow an extended retransmission. When errors occur causing retransmissions for portions of a packet, these retransmissions are interleaved in the subsequent portions of the next packet. This causes the inter-arrival period between two consecutive packets to appear shorter.

The variability in inter-arrival times of packets in these experiments is an order of magnitude larger than that typically seen over wired internet links. Figure 2(a) illustrates that inter-arrival latencies fluctuate anywhere between 70 - 610 milliseconds in this test. Hence a round trip journey would introduce an even larger degree of variability taking both directions of the GPRS link into account. Round trip times (RTT) play a critical role in real-time multimedia communications, and it is widely accepted that human quality perception is severely affected by both end-to-end latency and sample inter-arrival variability. The ITU-T [2]) recommendation for maximum round trip time targets in interactive voice communication is 400 milliseconds. Achieving this over GPRS links with no retransmission is clearly challenging, however the introduction of further packet latency in the form of inter-arrival jitter can cause the transmission latency to exceed this threshold by three or four times the recommended limit depending on the sample size. For less interactive applications, such as emerging *push-to-talk* services, slightly longer transmission periods can be handled, however it is desirable to maintain as consistent a channel condition as possible, particularly with respect to jitter. We therefore considered the effects of tuning link layer parameters for interactive multimedia.

Figure 2(b) presents results from a similar test with reliability disabled at the link layer. In this case, the inter-arrival distribution is much tighter, ranging only from 195 - 255 with a mean of 235. In disabling retransmissions we minimise the inter-arrival variance at the expense of a higher degree of packet loss (3.8% packets lost in this case). We consider that this is a worthwhile trade-off for interactive communication, and it is important to encode data at the application level accordingly to reflect the

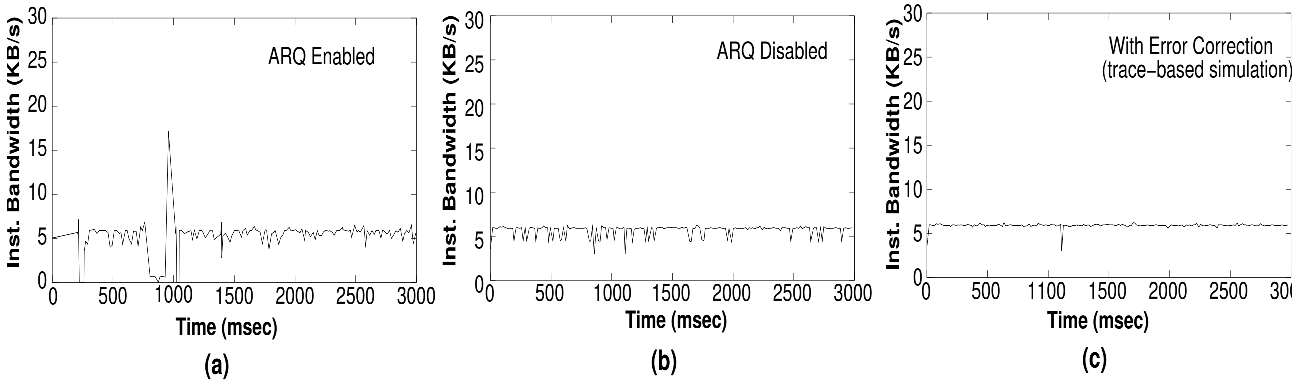

Figure 3. Figures shows the instantaneous throughput of the link with (a) ARQ enabled, (b) ARQ disabled and (c) with error correction applied (trace-based simulation).

channel conditions. Unlike traditional wired internet links where packet loss is typically always caused by congestion at routers, it is clear that losses over the WWAN link will occur randomly and there is no inter-dependency between link loss and the application bandwidth. We address new techniques to handle link loss in section 3.

Figures 3(a) and 3(b) also show the measured throughput of the link for both tests. The extreme drop in bandwidth followed by a peak in the reliable case (figure 3(a)) illustrates the effect of retransmissions of portions of a packet followed by a shorter consecutive interpacket arrival measurement.

Based on these observations, it is clear that for efficient real-time interactive communication over GPRS links, ARQ should be disabled in preference to avoiding packet loss. We also note that a non-interactive application can utilise link-layer reliability (ARQ) but should maintain a sufficient buffer to accommodate the variations in inter-arrival times. Our tests so far have not evaluated the impact of adjusting the value of the retransmission attempts variable for non-interactive streaming applications. It is apparent from our analysis, however that this would have an important impact on time sensitive streams, and would greatly benefit from some application or user-level influence. Our tests indicate that for both the ARQ-enabled and disabled case, occurrences of link blackouts and fading may occur. Where ARQ is enabled, recovery may take significantly longer and there is also an increased possibility of receiver buffer starvation where the link throughput is not sufficient to drain the base station buffer, consequently forcing data to become stale in the queue. It is essential to maintain an accurate flow control scheme in order to keep the amount of outstanding data in flight to a minimum.

Following these measurements we were able to conduct more in depth analysis of the packet data channel and retrieve data frames that had experienced corruption. The following section presents results on the actual impact of link layer corruption on the data stream.

3 End-to-end Transport Optimisations

Each GSM burst transmission provides a raw data rate of 456 bits which can be encoded in one of 4 ways named Coding Schemes (CS) 1 through 4. The difference in the schemes is the level of redundant FEC in the transmission. With each increase in redundancy, the usable throughput decreases, varying from 9.05 kbits/s (the highest level of FEC) through to 21.4 kbits/s (no FEC). In the following section we present the inference technique for aligning data to the coding scheme selected by the operator and consequently outline our Unequal Error Protection (UEP) scheme that leverages the inference capability. Our scheme relates closely to the UDPLite notion of corrupt frame utilisation [3–5], and in fact, we consider the two approaches complementary since UDPLite does not specifically mandate an error recovery and detection scheme for payload data.

3.1 Application-Level Inference of WWAN Links

Through tests with reliability and error checking disabled over GPRS we retrieved a number of samples of corrupt packets. We measured the length of burst errors which occur quite frequently. Studying the error patterns we noticed that burst lengths can always be expected to occur in multiples of the unencoded RLC

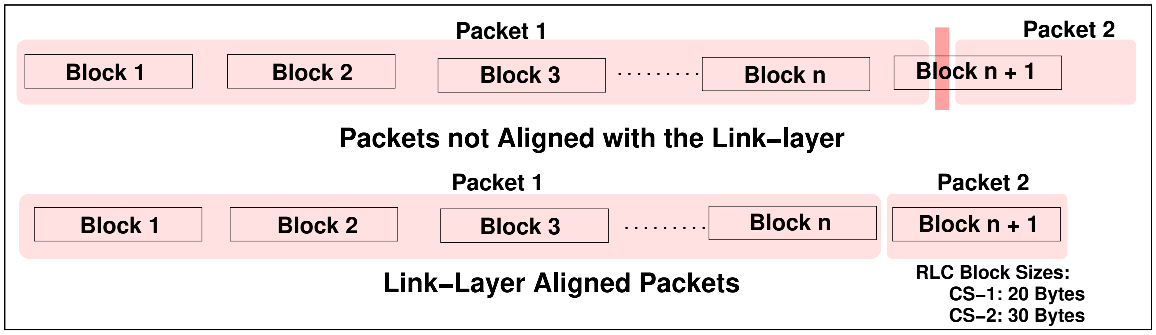

Figure 4. Aligning a packet onto RLC frames.

input data size. We also observed that packets can be structured such that the length and location of data exactly correspond to the underlying RLC blocks. For most cellular operators, we have observed fixed coding schemes being used, which appears to remain fixed while the context is active. Further analysis of GPRS devices indicates that in certain cases it may be possible to extract information about the link layer RLC encoding scheme based on the timed release of data, although this is not universally the case. The best solution would be to provide an application API to query the device during a connection and extract the relevant information. Since devices typically adopt a standard serial interface, however there is only a single channel over which to send either control information or data with limited flexibility to change between states. The solution that we have adopted is a cooperative end-to-end approach whereby the first packet error occurrence detected at the receiver is immediately returned to the source for analysis. Since the source can determine the original contents, it is an effective signalling approach for dynamically establishing the link conditions and adjusting the encoding accordingly.

Figure 4 demonstrates how the alignment of data packets over the RLC layer is applied. Our measurements indicate that if the size of the data packet is an exact multiple of the underlying coding scheme data block (taking account also of the lower level link headers and CRC bytes), the application can *infer prior to the transmission* where the potential burst loss boundaries may occur. This is because each packet will map directly onto an exact multiple of RLC frames. In the case of CS-1 coding we see block sizes of 20 usable bytes, for CS-2 coding this increases to 30 Bytes. Based on this, we can design a transmission mechanism that pro-actively encodes the data according to the potential loss patterns that would be expected, and consequently provides finer grained error recovery. Our scheme would also require an error detection mechanism that matches the block level granularity, and we therefore assume a 1 byte CRC hash is added to each block. Our experiments show that data received subsequent to the FEC decoding process, is either completely accurate, or completely destroyed impacting every byte across the block, indicating that the probability of the CRC failing to detect the error and consequently going unnoticed would be negligible. Given the ability to determine the underlying RLC block structure based on application level inference techniques, we now consider an encoding scheme that leverages this information.

3.2 Data block Encoding

The notion in generating Unequal Error Protection media, is that the encoded output can be classified into different priority blocks. It is generally acknowledged that providing Unequal Error Protection for layered quality encoded multimedia data can be very beneficial to the user perception of received quality [6–8]. Consequently a number of layered quality codecs are increasingly becoming available for Internet streaming, although typically prioritisation is still applied on a per-packet basis. An example of an audio codec designed for layering is the Adaptive Multi-Rate encoding (AMR) scheme with e.g. 3 levels of priority encoded data, which in this case would correspond to A, B and C frames, encoded in a typical sample ratio of 32%,42%,25% respectively of the total bandwidth. MPEG video is another example, where frames are classified as I, P or B frames at a typical ratio of 60%,30%,10% respectively. In terms of user-perceived quality, the highest priority frames are typically considered critical data in maintaining media comprehensibility, other levels provide enhancement data such as textural information that can be lost with little relative impact on the user-perception. For simplification, our model assumes that only the highest level would need to be protected, the other levels being open to corruption since they are not classified as critical elements of the stream.

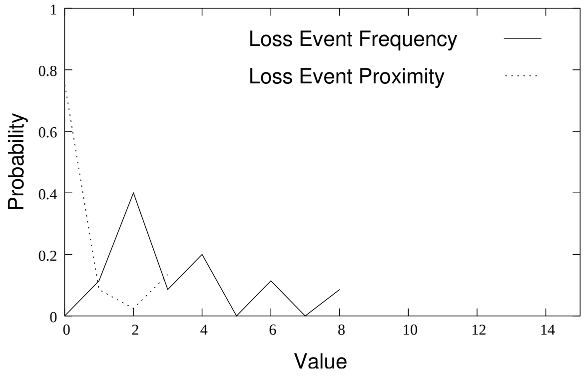

Figure 5. This graph shows the frequency of loss events within a single corrupt packet and the proximity of events to each other.

Since errors over the GPRS link appear in fixed size bursts, as our tests confirm, we want to be able to recover our protected data up to the predicted number of burst events within the packet. Hence we consider a typical Reed-Solomon coding approach [9] towards packetisation of the data. The source data would be encoded into n data chunks, the length of which is determined by the underlying coding scheme, creating an (n, k) code, where $(n > k)$, out of which any k blocks are required to recover the source data. The optimal value of $(n - k)$ would be determined by the number of predicted error events from which we wish to recover. Similar approaches for coding would include the newer 'tornado' type codes (e.g. [10, 11]) which provide a lower encoding and decoding complexity algorithm for large source data sets. One of the advantages held by the Reed-Solomon approach over Tornado codes is the inclusion of the unmodifed source data, useful for decoding where corruption has not occured. Tornado codes also introduce a nominal increase in source encoding size which we refer to as θ whereby the size of k encoded blocks is equal to θD where D refers to the original data size. Typical values for θ would be an increase of 3% of the original.

The encoding relationship can be expressed mathematically, next we discuss a theoretical analysis of the algorithm performance. We introduce the parameter l to indicate the underlying encoding bucket length. For D, the original data size, we refer to the division of protected data versus unprotected data as $D_p = p * D$ and $D_{1-p} = (1 - p) * D$ respectively. The value of k encoded blocks of data D_k can be expressed as θD_{1-p} where $\theta >= 1$. D_k can also be expressed in terms of l as $D_k = \theta l k$. The additional redundancy applied by the code is the number of extra blocks represented by $n - k$ which we refer to as ϕ. The size of ϕ, or D_ϕ can be expressed as $D_\phi = ((n - k)/k) * D_k$. We can therefore model the total encoded data size D' as a function of D, p, n and k:

$$D' = D_p + D_k + D_\phi \tag{1}$$

$$D' = pD + \theta(1 - p) * D + \frac{n - k}{k}\theta(1 - p) * D \tag{2}$$

or

$$D' = D(p + \theta(1 - p)(1 + \frac{n - k}{k})) \tag{3}$$

3.3 Burst Error Characteristics Analysis

Next we studied the characteristics of the loss bursts over the link to generate a clearer picture of the anticipated frequency and proximity of the loss events. Figure 5 indicates the results from the same experiment. As expected, the errors occur in very close proximity to each other, and in all cases where more than 1 burst occurred, the distance between the events never exceeds 3 slots. The results also indicate that more than one event occurring within a single packet is very common, the highest frequency value being 2, the largest occurrence being 8. These results therefore provide some insight into the potential design and performance of an interleaving pattern for critical versus non-critical 'padding' data. It is assumed that critical data will be surrounded by unprotected stream elements in order to minimise the impact of corruption and achieve the same level of protection with lower FEC redundant coding overhead.

Further analysis of the location and occurrence of bit errors over the link for larger data sets is the subject of our future research and will be most beneficial in determining efficient values for $(n-k)$ and interleaving patterns. Given these constraints, we evaluated the impact of the reliability scheme versus the overhead introduced by the FEC scheme through trace based analysis of the sample test introduced earlier in figure 3(b). The following results assume no unprotected data padding, and losses are assumed to always affect the critical data elements.

3.4 Trace Based Analysis

In order to understand the potential impact of this scheme, we present some trace-based analysis of the theoretical performance that might have been expected based on our new packetisation scheme. We retroactively analysed the impact of utilising corrupt packets, and used the regenerated trace to provide a channel simulation. In the sample test presented in Figure 3(b) we notice that out of the 3.8% or 38 lost packets, we were able to recover 35 of the packets, the remaining 3 being lost due to header corruption. Figure 3(c) demonstrates the improvement in bandwidth fluctuation that can be achieved. The key advantage as outlined by figures 3 (a),(b) and (c) is the much greater control of the instantaneous link bandwidth achieved by the optimised packetisation approach. The jitter is tightly controlled and the fluctuation in link throughput is minimal providing a much more amenable environment for multimedia traffic.

Given the improved channel conditions, we considered the impact of the scheme outlined above. For the coding scheme of CS-2 as presented in these tests, we considered the encoding of MPEG-4 packets up to a size of 1350 bytes. The encoded payload size (D')of 1350 was assumed since it is the closest multiple of 30 to 1371 (the actual payload used in these tests), and consequently $n = 45$. Divided into layers of 60%,30%,10% for I, P and B frames respectively, the protected data would include just the I frames, or 60% of the source data, generating a value of $p = 0.6$. For this sample test, we retroactively analysed the impact of variable degrees of FEC applied in advance of streaming. Table 1 demonstrates the trade-off in FEC overhead versus percentage of data packets recovered from the channel simulation presented in figure 3(c) with a value of $\theta = 1.03$ to simulate a standard tornado code.

(n - k)	D (Bytes)	% FEC Overhead	% Data Recovered
1	1308	3.1	10.53
2	1290	4.4	47.37
3	1271	5.85	55.26
4	1252	7.26	73.68
5	1233	8.6	73.68
6	1213	10.15	84.21
7	1193	11.63	84.21
8	1172	13.19	92.11

Table 1. The percentage of data recovered versus increased data overhead

As indicated in table 1, for an estimated increase of only 4.4% in packet overhead, we can recover up to 2 burst losses in each packet, or 44.74% of corrupt data in this example. The best theoretical recovery we could achieve from this test (3 of the packets were unrecoverable due to header corruption) was 92.1% introducing an FEC overhead of 16.3%.

4 Conclusions

The characteristics of WWAN links present challenges to different applications. In this paper we focused on the performance of multimedia streams across WWAN links, and demonstrated that the variability in the packet inter-arrival times can be too large for such applications. We investigated ways to adjust the link layer parameters to minimise the inter-arrival jitter of packets for such applications. Our tests confirmed that there is a significant improvement in performance at the expense of increased packet loss.

We further investigated the impact of loss events on the actual received data rate at the byte level in relation to the rate imposed through traditional packet checksum rules. Our tests indicated that there is a good deal of potential performance improvement through utilising good portions of corrupt frames, and in particular we identified a method for inferring the underlying encoding scheme and subsequently optimising the encoding of data within a packet to match the underlying RLC boundaries. As a consequence, applications can determine exactly what the potential locations of bursts may be in advance of transmitting a packet, and detect the extent of corruption down to the exact size of the burst length.

Ongoing research involves more in depth analysis of the burst characteristics, and the design of real systems to operate across the links. We intend to demonstrate that utilising corrupt data frames is very beneficial for cellular type links in particular due to the high occurrence of errors and the precisely defined RLC data lengths.

References

1. Christian Bettstetter, Hans-Jörg Vögel, and Jörg Eberspächer. GSM phase 2+ general packet radio service GPRS: Architecture, protocols, and air interface. *IEEE Communication Surveys*, 2(3), 1999.
2. Itu-t recommendation g.114, (05/00): One-way transmission time. Technical report, ITU-T, May 2000.
3. L.A. Larzon et al. The udp-lite protocol. Internet Draft, Internet Engineering Task Force, December 2002. Work in progress.
4. A. Singh, A. Konrad, and A. Joseph. Performance evaluation of udp lite for cellular video. In *Proceedings of NOSSDAV*, June 2001.
5. F. Hammer, P. Reichl, T. Nordstrom, and G. Kubin. Corrupted speech data considered useful. In *Proceedings of the First ISCA Tutorial and Research Workshop on Auditory Quality of Systems, Mont Cenis, Germany*, April 2003.
6. Y. Wang, A. Ahmaniemi, D. Isherwood, and W. Huang. Content-based uep: A new scheme for packet loss recovery in music. In *To Appear in Proceedings of ACM Multimedia 2003*, 2003.
7. T. Stockhammer and C. Buchner. Progressive texture video streaming for lossy packet networks. In *Packet Video Workshop, Korea*, page 57, 2001.
8. Steven McCanne, Martin Vetterli, and Van Jacobson. Low-complexity video coding for receiver-driven layered multicast. *IEEE Journal of Selected Areas in Communications*, 15(6):982–1001, 1997.
9. I.S. Reed and G. Solomon. Polynomial codes over certain finite fields. *J. Soc. Industrial Appl. Math, volume 8*, pages 300–304, 1960.
10. L. Rizzo. On the feasibility of software fec. In *DEIT Technical Report LR-970131. Available as http://www.iet.unipi.it/ luigi/softfec.ps*, January 1997.
11. John W. Byers, Michael Luby, Michael Mitzenmacher, and Ashutosh Rege. A digital fountain approach to reliable distribution of bulk data. In *Proceedings of SIGCOMM*, pages 56–67, 1998.