Exploiting Diversity to Enhance Multimedia Streaming Over Cellular Links

Julian Chesterfield[†], Rajiv Chakravorty[†], Ian Pratt[†], Suman Banerjee[‡], Pablo Rodriguez[§]

[†]University of Cambridge Computer Laboratory, U.K., [‡]University of Wisconsin, Madison, U.S.A

[§] Microsoft Research Cambridge, U.K.

Abstract— **Wireless Wide Area Networks (WWANs) are becoming ubiquitous across most geographic regions, enabling simultaneous coverage from multiple providers. WWAN channels exhibit both uncorrelated and correlated behaviour on a variety of levels. In this paper we examine the statistical properties of WWAN links, and illustrate the benefits in heterogeneity that can be exploited to improve statistical throughput and Multimedia quality. Our results are based on real network measurements. We describe the design and implementation of a high quality multimedia streaming application that implements WWAN streaming optimisations utilising Unequal Error Protection coding techniques, and we evaluate the performance over an operational WWAN network.**

Index Terms— **Multimedia, Cellular, UEP, GPRS, Networks, WWAN, Streaming, Vorbis**

I. INTRODUCTION

Wireless wide-area cellular networks such as GPRS and UMTS 3G networks in Europe and CDMA 2000 networks in America and Asia are being widely deployed, promising ubiquitous access to IP-based multimedia applications for end-users. The characteristics of the cellular wireless medium, however pose several new challenges to audio and video multimedia streaming applications. In this paper we consider the performance of such interactive multimedia streaming over cellular environments.

In contrast to the relatively stable indoor 802.11b WLANs, wide-area cellular links are severely plagued by the vagaries of the cellular environment such as multipath fading, interference, cell re-selection etc.. The net effect is that cellular links generally suffer from high and variable round trip times, rate fluctuations, link outages and occasional burst losses e.g. during deep fading and hand-overs etc. that can significantly impair efficient transmission of multimedia data.

In addition to the residual problems of the cellular wireless medium, the end-user experience of such multimedia applications depends heavily on a users perceptible interaction and tolerance to delay. Unlike 'staple' Internet applications (e.g. web-browsing, e-mail etc.), where user tolerance is generally quite flexible, for multimedia applications, a user can not tolerate *any* flexibility beyond certain interactivity constraints.

In video conferencing applications, for example, the delay constraints are tightly coupled with the human perception of interactive communication, whilst for one-way broadcast-style applications the delay tolerance is typically based on the willingness of the listener to trade-off initial start-up delay versus long-term performance improvement through smart buffering and data retransmission. Thus, we find many challenges at different levels for utilising such applications in the cellular environment.

In this paper we present our practical experiences of streaming multimedia over production cellular networks, and our attempts to build a system that improves end-user experience. After a brief introduction to WWAN characteristics in section II, we highlight the relevant performance characteristics of WWANs for multimedia streams in section III. In section IV we present the design of a multimedia streaming tool, *vorbistreamer* that implements a variety of optimisation techniques for the cellular environment, and in section V we evaluate the relative benefits of exploiting different levels of diversity across WWAN links.

We briefly summarise the main contributions of our work as follows:

- We present the first detailed evaluation of multimedia applications in wide-area cellular environments,
- We demonstrate the importance of application interactivity requirements on configuration parameters of the wireless channel through real experiments, and quantify the *potential for diversity in cellular environments from a variety of naturally occurring phenomena of random and non-random events*,
- We evaluate the advantages of correlated and uncorrelated behaviour in cellular environments to exploit diversity to significantly improve multimedia streaming performance.

Our work shows that we can significantly improve end-user streaming experience by exploiting diversity.

The collective benefits of the schemes proposed show an improvement in goodput of the link of around 5%, and a decrease in error bursts through channel interleaving of 14%.

II. WWAN Technology

A. Cellular Technology Overview

Cellular networks offer lower date-rates but often span longer distances, accommodating more users in a given cell than short area networks such as WLANs. The links have to cope with harsh outdoor conditions such as *multi-path fading* caused by buildings, hills, or other obstructions. To overcome the *multi-path* problem in wireless – in the physical layer – TDMA based systems like GSM/GPRS use diversity reception techniques like *Viterbi equalisation*, while CDMA based systems (IS-95/B) make use of *rake* receivers. Table II-A illustrates the comparative characteristics of some of the most commonly deployed wireless networks. These technologies range widely in frequency spectrum allocation, transmission range, Access control and modulation techniques, but while each of the WWAN systems utilises different approaches toward error detection and recovery, they all experience similar path fading and interference phenomena.

Characteristics	GSM/GPRS	UMTS-3G (WCDMA)	IEEE 802.11b
Frequency band	900MHz	2GHz(WCDMA)	2.4 GHz
Carrier Rate	200KHz	1.25MHz	5.5 Mb/s
Typical range (m)	Cellular	Cellular	50-100m
Medium Access Control	slotted aloha TDMA	DS/CDMA	CSMA/CA (DCF)
Modulation	GMSK	8-PSK/QPSK	CCK

TABLE I

WIRELESS CHARACTERISTICS - A COMPARISON.

In GPRS, the Radio Link Control(RLC) layer is used to transmit data blocks by a selective repeat ARQ technique. The sender transmits blocks within a window of 64, acknowledged periodically by receiver side ACK/NACK messages. The sender re-transmits any erroneous RLC data blocks, and then further slides the sending window. This happens at the expense of high and variable delay due to such retransmissions.

The client data packets (IP) are fragmented into blocks of RLC data and transmitted as GSM bursts in a *multi-frame* construct or 'channel' [1]. The number of channels that a device can support simultaneously depends on the hardware itself, although typically a GPRS capable device such as a cellphone will handle 4 channels downstream and 1 up, referred to as a '4+1' device.

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
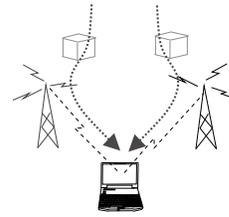

Fig. 1. A multi-homed wireless device capable of media striping across multiple channels.

difference in the schemes is the level of redundant FEC in the transmission, thereby impacting the throughput. It is generally acknowledged that CS-2 provides the best trade-off in throughput to error resilience, and is therefore the most commonly assigned encoding rate [2], [3].

B. Multimedia Streaming Protocols

There are standards being defined to relax the rules controlling the strict integrity of data across layers through the use of packet level CRCs in the form of UDPLite [4], [5]. This new protocol provides a flexible UDP error checksum for network streaming applications that may benefit from utilising partial portions of corrupt data packets. Various implementations and evaluations of UDPLite [6], [7] have been described, demonstrating that in certain circumstances (typically networks that are latency constrained), a great deal of benefit can be derived from utilising corrupt data. To complement the advances in partial error detection, there has been a similar interest in the development of Unequal Error Protection (UEP) encoding of multimedia data. This involves prioritising the encoded data into separate layers, enabling successful decoding of the data with only partial information, the quality of decoded data improving as more of the original sample is retrieved. We build upon this research through the design of our own UEP support scheme in this paper.

C. Streaming Media Architecture

MAR (the Mobile Access Router) is a real mobile system designed to aggregate wide area wireless links. By leveraging the inherent diversity between parallel WWAN links, MAR is able to statistically improve the channel performance and present local system users with a more reliable and consistent combined channel. In [8] we demonstrated that the diversity of the links can also benefit streaming applications in supporting a lower buffer for real-time streams. We evaluate the options for exploiting diversity in multi-homed systems such as MAR for multimedia streaming in greater detail in this
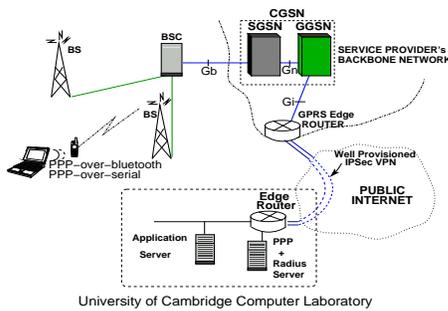
Fig. 2. The experimental testbed used for conducting the GPRS tests. A well provisioned, low latency tunnel transfers all Computer Laboratory GPRS device traffic directly to our PPP server.

Fig. 3. The inter-arrival time CDF of packets (1000 packet samples, 1371 bytes size) with ARQ enabled and ARQ disabled.

paper. We also extend our notion of diversity to encompass *intra*-channel uncorrelated behaviour occurring as a result of error characteristics of the link as well as *inter*-channel diversity.

Figure 1 illustrates the architecture of our streaming environment. A multi-homed mobile client exploits channel diversity by striping data across multiple WWAN interfaces, connected either to different cellular providers, or the same provider. The application itself operates transparently to the network below, and is designed to be robust to failure by adapting to the environment through inference techniques rather than explicit link knowledge.

III. CHANNEL AND DIVERSITY CHARACTERISATION

Through measurements of the qualitative aspects of WWAN channel performance, we can quantify the typical constraints imposed upon a multimedia streaming application. Parameters such as packet propagation latency, inter-arrival jitter and loss can have a significant impact on its' design and operation. In this section we present characterisation results for GPRS, and additionally evaluate performance diversity in the WWAN environment, looking at error characteristics and parallel channel diversity.

For all tests we utilise a GPRS network testbed that is linked to our lab in Cambridge (see figure 2), providing us with actual data results from a live network. The traffic from each of our registered mobile devices travels directly from the provider network GGSN to our lab router via a well provisioned VPN tunnel. Since the performance of the VPN is well understood and highly consistent, we are able to accurately infer the effects of data transmission across the wireless link and measure network performance metrics such as jitter and loss.

GPRS channel performance varies considerably depending upon the proximity to the base station and quality of the signal. In previous work [8] we attempted to characterise some of the variation in performance
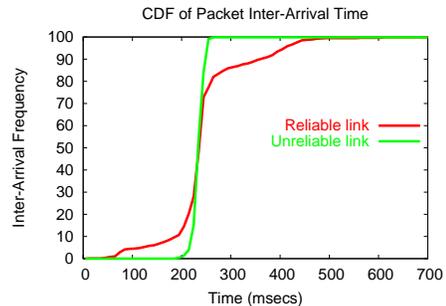
as measured during static and mobile tests in multiple locations providing a variety of signal qualities. Our tests demonstrated that there is a significant amount of diversity in WWAN networks which can be leveraged to provide better application performance. We exploit channel diversity further in this paper. For our multimedia streaming experiments, we selected an area with average signal quality using a stationary device for the duration of the tests, we do not address the impact of mobility in this paper. For measuring general link performance we used our own bandwidth measurement tool which utilises consecutive packet propagation times to accurately measure the capacity of the link.

A. Channel Characterisation

GPRS provides three useful channel conditions that must be negotiated prior to link activation. The first mode provides full reliability with retransmission and error checking, the second mode provides no reliability with error detection enabled, and the third mode provides no reliability with error checking disabled.

1) Latency: To measure the packet propagation time across the wireless link we calculated the inter-arrival time between consecutive packets transmitted in a burst to the GGSN. Figure 3 indicates the cumulative distribution of inter packet arrival times for a 1000 sample test. It is apparent from the results that when reliability is enabled there is a significant variation in propagation times across the link, the reasons for which are attributed to the aggressive ARQ applied by the link. The evenly spread distribution can be attributed to hardware buffering of packets which causes some packets to be buffered awaiting RLC retransmissions, and then subsequent packets to be released early. In general, however the average propagation time for a packet when no retransmissions occur is a factor of the packet length itself and the raw channel capacity. For a $4 + 1$ device utilising CS-2 encoded channels with a raw throughput of 53.6Kbits/s,
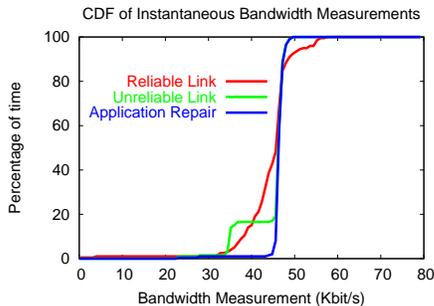
Fig. 5. The CDF of the measured capacity of the link, demonstrating unreliable, reliable and application repaired measurements.

Fig. 6. The challenge of trying to maintain a sufficient buffer in advance of the media playout over a reliable GPRS link compared to transmitting data over an unreliable link.

it takes around 209 milliseconds (ms) to transmit 1400 bytes. This observation matches our measured results in figure 3, demonstrating that the most significant delay in the end-to-end path is caused by the wireless link.

Given the average propagation delay of the link, and the slower up link channel, only 13.4kbit/s, it is clear that the total Round-trip time is substantial. 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 [9] recommendation for maximum round trip time targets in interactive voice communication is 400 ms. 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.

2) Network Jitter: The variation in inter-arrival times of packets over GPRS links is an order of magnitude larger than that typically seen over wired Internet links (figure 3 illustrates measurements between 70 - 610 ms). 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 consider the effects of adjusting reliability of the link.

Figure 3 also presents results from a similar test with reliability disabled. Through disabling retransmission, the inter-arrival distribution is much tighter (ranging only from 195 - 255 ms). The inter-arrival variance is minimised at the expense of a higher degree of packet loss (3.8% packets lost in this case). We also note that losses over the WWAN link occur randomly and there is no clear inter-dependency between link loss and the application bandwidth.
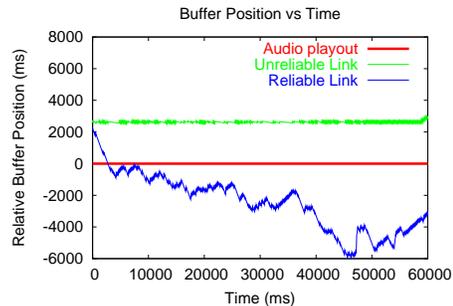
3) Channel Capacity Fluctuation: Based on these measurements of network jitter variation for both reliable and unreliable channels, we consider the resulting channel capacity as measured by the mobile client for each test. Figures 4 and 5 illustrate the fluctuation of channel capacity over time for stationary tests with average signal quality. For the reliable channel, we find that the capacity ranges quite widely, typically under utilising the physical bandwidth of the channel. The unreliable cases however illustrate that for a higher level of packet loss (in this case 3.8% versus 0%) our channel capacity can remain much more constant, providing us with a smooth arrival of data over time. The third CDF plot indicates the benefits of intra-packet error detection (to be described further in section IV-C) demonstrating that the loss effects on the unreliable channel causing a significant number of measurements around 35 - 45Kbit/s can be removed to produce a very smooth and predictable channel. We therefore consider the use of partial packet error detection at the UDP level as advocated by UDPLite adopters, to utilise as much of the channel bandwidth as possible.

4) Buffer Performance measurement: The impact of variability on a streaming media application depends upon the constraints of the application itself. It is clear that a complete and reliable stream is always preferable for a receiver if there is sufficient flexibility to support the variation in inter-arrival times and fluctuating channel capacity. We therefore can model the relationship between the length of the media that is to be transmitted and played, the target bit rate of the application and the tolerance of the user to an initial startup delay.

Figure 6 demonstrates the challenges associated with buffer management for real-time media streaming over wide area wireless links. These tests were conducted using a 2 second initial playout buffer for a 60 second music clip. Figure 6 does not address the ability of the application to pause the audio playout stream as it
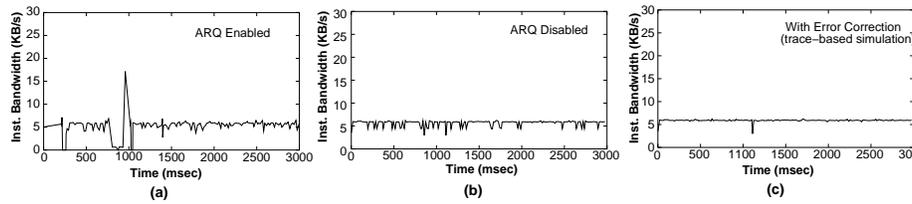
Fig. 4. The instantaneous bandwidth of the link showing (a) reliable mode, (b) unreliable mode and (c) a reconstructed stream based on application based on fine-grained error detection.
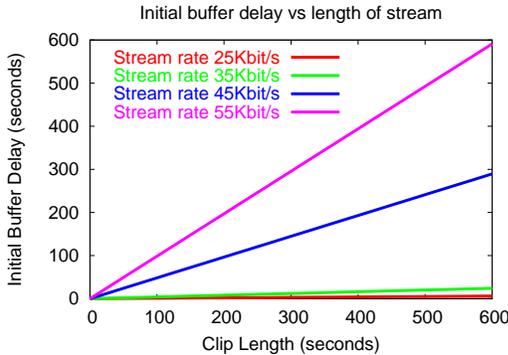

Fig. 7. The initial buffer delay that should be inserted prior to media playout in order to support various streaming rates.

is intended to highlight the significance of the initial buffer delay to support the streaming rate. It is clear that when reliability is enabled, we cannot guarantee that the link will not go stale in the event of a severe fade or interference and cause the application buffer to under run. We consider the theoretical buffer limit of an application based on average link measurements over time to support a target bit rate and clip length without experiencing buffer under run.

Based on our earlier results presented in figure 5 we evaluate percentage of capacity available over time and consequently the ability of the link, under these circumstances, to support a minimum target streaming rate. Figure 7 indicates the theoretical buffer requirements required of our application to target various streaming rates over different periods of time based on the media clip length. We see that in order to support a higher streaming rate closer to the link bandwidth limit, our theoretical buffer requirement grows exponentially, requiring significantly more relative buffer delay as the bit rate increases.

In analysing the performance of the application buffer, we do not take account of the ability of the application to adjust the streaming rate over time to more closely match the capacity of the channel. As indicated in the introduction, multimedia applications, while often capable of adjusting bit rate within certain bounds, have a fixed minimum target streaming rate which must be supported over the length of the stream by the link. In this analysis we consider the minimum target rate as an absolute threshold. For pre-encoded media it is always beneficial to attempt to stream as much data as possible in advance to the receiver whenever excess capacity exists, thereby increasing the data buffer resilience to throughput fluctuation. For many applications that generate media in real-time, such as radio or television type streams, such pre-streaming of content is not possible, and we therefore consider all content generation to also be time constrained for the purposes of these tests.

5) Flow Control Accuracy: In order to avoid data in flight getting buffered at the Gn interface, going stale and causing subsequent packets to also be delayed it is important to maintain a tight flow control between the source (which might be a proxy device located in the providers network) and receiver. Detecting available capacity over a GPRS link is difficult to accomplish since the channel assignment from the base station can be highly dynamic, causing a data flow to slow down with potentially no data loss for a long period of time.

One of the advantages of utilising an unreliable mode for the link is that the actual channel capacity is easier to measure. Since the received data rate when operating in unreliable mode with application based error checking always matches the capacity of the link or the application data rate, whichever is lower, it is clear that our detection time for link capacity fluctuation is higher than the reliable case. In figure 4(a) the severe fluctuation around the 1000 ms time frame demonstrates the instantaneous measured throughput reaction of the channel where our capacity can appear to fluctuate very fast, although it does not necessarily reflect the capacity at that point in time since it is apparent that such a capacity does not physically exist across our GPRS channel (a 4 + 1 device experiencing CS-2 channel coding can only achieve a bit rate of 56Kbit/s or 7KByte/s). Such fluctuations occur due to the in-order delivery requirement of the link and RLC retransmissions. The solution for gathering
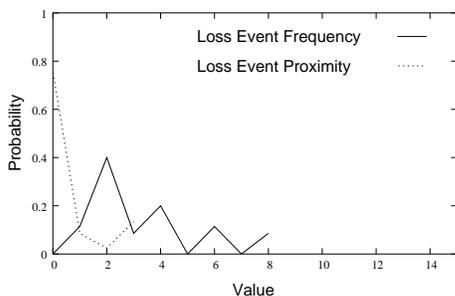
Fig. 8. This graph shows the correlated behaviour of loss bursts within a single loss event, e.g. within a corrupt packet, highlighting the proximity of events to each other.

Fig. 9. GPRS correlation between links as evaluated under MAR, static conditions.

a more accurate reading must involve increasing the measurement period, thereby decreasing the reaction time of any flow control algorithm.

B. Diversity Characterisation

Diversity in WWAN links can be measured in various ways. In the following section we measure both correlated and uncorrelated behaviour in the WWAN environment, focusing on channel errors and diversity between parallel links.

1) Channel Error Characteristics: Channel errors exhibit both correlated performance, errors typically occur in bursts, and uncorrelated performance, burst events occur randomly. When operating in unreliable mode, a mobile host can cause all data to be received from the link without error detection. By disabling packet level checksums at the UDP level, either through UDPLite partial header coverage or otherwise, the application can extract errored data arriving over the WWAN interface. Due to the nature of the RLC layer, transmitted data is typically subject to an FEC coding process that attempts to recover a certain percentage of errors in hardware, the impact of which on the application is that errors appear in the bit stream as fixed size blocks equivalent to the residual data size provided by the RLC Coding Scheme (CS). Section II-A explains the CS schemes in more detail. The error rate measured at the application level therefore is not the absolute BER as measured over the link, but a post-FEC decoding rate that operates on an RLC residual coding rate granularity.

We classify errors as events which occur on a packet level, and we look at the frequency of events, and the characteristics of bursts within an event. We conducted a series of experiments to measure losses in the data stream as introduced by the link. Figure 8 characterises the nature and frequency of bursts within a single event, demonstrating that errors occur in close proximity to each other, and in all cases where more than 1 error
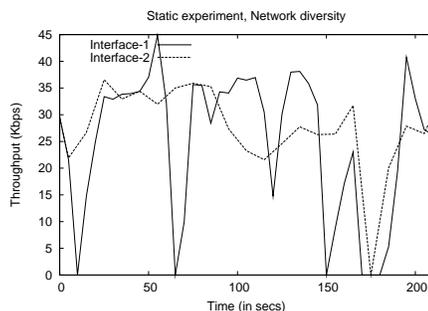
occurred, the distance between the subsequent errors never exceeds 3 slots. We also find that there is a high probability (the calculated probability is 0.9) of more than one error occurring.

2) Parallel Channel Diversity: In the design and evaluation of MAR [8], we measured the relative benefits of utilising *Link Diversity* between different wireless technologies, different devices but with the same operator and between different operators. It was demonstrated that due to the nature of errors in the WWAN environment, there is a great deal of diversity that can be exploited. Technology diversity provides the most uncorrelated behaviour although it is not always the most practical solution particularly where network performance such as round trip times are an order of magnitude different, e.g. in the case of 802.11 links and Wide Area Wireless GPRS links. The best uncorrelated performance for similar technologies occurs between separate providers where different portions of spectrum are utilised. In this paper we evaluate the benefits of network diversity for a multi-homed device capable of intelligently receiving data across multiple links.

In both static and mobile scenarios, see figure 9, the statistical diversity between interfaces helps to provide a smoother overall throughput, and lower packet inter-arrival times for data traffic. Our measurements indicated that there is a higher degree of correlation between interfaces under mobile situations, or as a result of extreme environmental conditions, however on average we observe significant potential to exploit different kinds of diversity in the WWAN environment.

IV. DESIGN OF A ROBUST STREAMING TOOL FOR WWAN LINKS

Based on our WWAN measurements we have designed a streaming tool called *vorbistreamer* that is adaptable to the WWAN environment, and in particular we consider the case of unreliable link streaming. Given the

benefits of utilising application based error recovery for unreliable multimedia streams as outlined in section III, we have designed an application that implements various recovery techniques, is optimised for cellular links and robust to failure.

A. Media Characterisation

In order to rate the resultant quality of Multimedia streaming over WWAN links, it is useful to characterise the content of the media itself. Multimedia encoding techniques of audio and video data vary considerably. Traditionally, physical audio and video signals are transformed into a constant digital representation over time through sampling. Since certain frequency components in a signal have higher importance than others in terms of human perception, a common approach toward encoding is to order data by importance such that easier application processing, whether it be to provide variable bit rate streaming or progressive decoding of a signal, is facilitated. The data rate produced by current Media encoders is typically fairly constant over time, but not exact, i.e. samples are not a fixed size. The reason for this is that signal analysis can evaluate the importance of a sample, such that certain samples benefit more than others from greater data representation, and a more perceptually pleasing result can be produced by maintaining flexibility with respect to the exact rate generated by the encoder. In all cases, however we find that the bandwidth produced by the encoder is not unrestricted, and will always range between a minimum and a maximum target. Any channel that cannot support the minimum data rate for constant periods up to the bounds of the playout buffer delay will cause decoder failure, and hence we attempt to maintain as predictable a channel performance as possible.

The Vorbis codec [10] provides high quality encoding of audio data, allowing flexibility with respect to the characteristic of the bit stream produced. Upon initialisation, various parameters can be specified such as Average, Constant or Variable rate coding with a quality or bandwidth setting to indicate the quality level required. The encoder initially generates a base layer 'spectral floor' which defines the basic 'shape' of the signal, followed by residual higher frequency components. The codec has the property that although residual layers do have an order of importance, ultimately any combination of residual layers can be combined to improve the quality of the signal. The encoded data therefore lends itself well to UEP formatting with a single base layer data stream, and any combination of residual layers received over a 'best effort' channel. For the purposes of these tests, a hybrid version of the codec was produced which allows
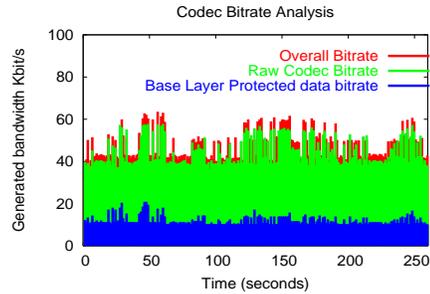

Fig. 10. The division of protected to unprotected data generated by *Vorbistreamer*, and the amount of packet and payload header overhead.

data to be directly categorised as it comes out of the encoder, in essence producing a UEP-ready version of the data. This allowed intelligent packetisation of the data prior to streaming. The codec also benefits from the optimised design of pre-streaming codebooks which must be transmitted reliably to the decoder, allowing a higher degree of signal compression for all components of the bit stream.

B. Raw Codec Performance

We selected the vorbis encoding format as a suitable high quality audio codec, and instrumented the encoder to our requirements in order to generate UEP content. Figure 10 illustrates the typical ratio of essential/protected data to higher level residue unprotected data. The ratio of data rates varies depending upon the maximum target bit rate of the encoder, with the rates closer to equal as the overall bit rate drops to its lowest threshold. The lowest overall bit rate achieved from the codec in separate tests has been around 30 - 35kbit/s including packetisation overhead, at which point the residue data rate and the base layer data rate are approximately equal. The codec ratio identified in figure 10, is representative of vorbis streaming rates in general as the overall data rate increases above approximately 40Kbit/s. The maximum data rate generated by the encoder is around 500kbit/s, although no cellular link in the wide area is currently capable of supporting this rate.

Each vorbis sample, as generated by the encoder, includes a header which specifies, among other things, the length of the sample and the position of the sample in the continuous audio stream. This allows any sample to be decoded independently of another, without being impacted by loss of earlier or subsequent data. This is an important property for our streaming environment since we anticipate that samples will be lost on occasions. Decoded vorbis sample sizes are always a factor of 2, and range between 64 to 8192 bytes. Typically, however
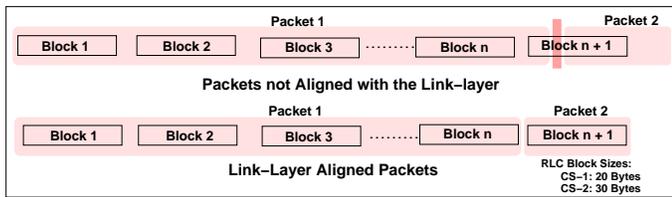
Fig. 11. The bucket alignment technique which optimises the location of CRC checks within a packet to minimise the amount of data lost through bit errors.

for lower bit rate streaming, as required over GPRS, the sample sizes tend to be equivalent to 32 millisecond segments of audio, ensuring that any single gap occurring in the decoder stream as a result of loss is not too significant.

C. Error Detection

In order to utilise as much of the channel capacity as possible we consider the use of partial packet checksumming in order to detect sub-packet errors. IP transport protocols typically apply packet-level checksums to data, preferring the simplicity of data retransmission to the complexity of forward error correction and sophisticated CRC algorithms. For wired links this is a very reasonable assumption since errors only occur in very isolated cases, many of which can be immediately repaired using fast local area ARQ schemes. For time constrained data streaming across WWAN links we cannot retransmit data for reasons outlined in section III, and errors occur at a much higher frequency than typically experienced. Figure 4(c) highlighted the benefits of discarding only partially corrupted data segments from errored packets, demonstrating that substantial improvements in goodput can be achieved. Such capability, however requires a fine-grained error detection scheme that is sufficiently advanced enough to isolate the corrupt data segments and allow processing of the rest of the data.

In order to enable the isolation of RLC sized errors within the data stream, we designed an error detection scheme that strategically placed 1 byte CRC hashes throughout the packet. Each hash has the capability of detecting errors occurring within a block of data of the RLC data length including itself. For CS-2 coding therefore we detect an RLC size of 30 bytes, assigning CRC hash markers for every 29 bytes of data throughout the packet. The error detection scheme therefore adds an additional 3% data overhead to the stream.

In the simplest case, due to non-alignment of CRCs with the RLC frames we find that any single error occurring at the RLC layer will typically impact 2 CRC

checks, however we can further optimise the detection and isolation of errors to the exact length of the errors themselves through opportunistic packetisation at the source. After conducting a series of tests with packet lengths of different sizes it became apparent that through the alignment of packets, an application is capable of *inferring prior to transmission* where the potential burst loss boundaries may occur and consequently to manipulate the accuracy of the detection scheme. This process is referred to as *Bucket Alignment* and has been utilised to great effect in optimising the error detection scheme outlined above. From the application level it is possible to build data packets in multiples of the RLC encoding size, also accounting for the lower level packet headers included by the UDP, IP, PPP and RLC layers, in order to ensure that the start of an RLC frame will always align exactly over the first byte of the packet. The impact of this scheme is to optimise the error detection, reducing the amount of data discarded, and thereby improving the recovery performance of any additional FEC scheme. Figure 11 illustrates the *Bucket Alignment* process that is implemented in *vorbistreamer*.

The method for determining the actual RLC block length is currently dynamic, requiring the receiver to periodically send compressed versions of corrupt data packets back to the sender for analysis against the original to determine the exact length of the errors. A future solution might be for the network provider to supply an Internet based query server that would directly supply the link information based on source IP address, or to locate a transparent proxy server that re-encodes data on the fly based on local knowledge of the link. Any such solutions, however would introduce commercial and security complications that are outside the scope of this paper.

While we have not evaluated the accuracy of the CRC scheme (there is a chance that errors may go undetected), our experience has been that an unsuccessful RLC decode process impacts all the data within a block significantly. The resulting data set is a completely random bit stream that permeates across all the bytes, decreasing any probability of systematic errors occurring within the block and thereby improving the detection likelihood of the hash value.

D. Error Recovery techniques

We now consider the ability of the technology to support varying levels of QoS for multimedia data. We increase the resilience to loss of the base layer data through partial intra-packet FEC redundant encoding, and statistically decrease the probability of errors occurring over protected portions of data through internal
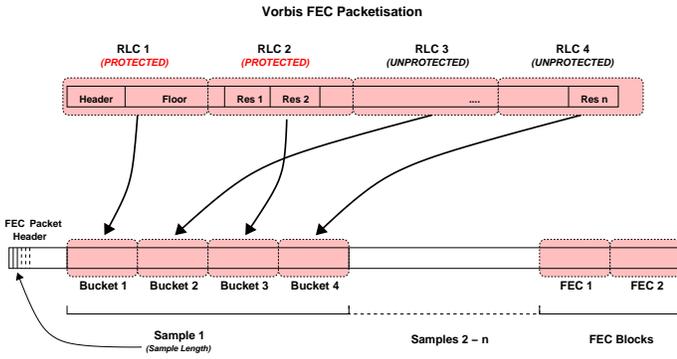
Vorbis FEC Packetisation

Fig. 12. The packetisation approach for utilising intra-packet error checking, and generating Unequal Error Protection for media such as Ogg Vorbis encoding.

interleaving of protected and unprotected data. Additionally we interleave samples across packets within the stream in order to lower the impact of unrecoverable burst errors in the decoded audio, and consider the possibility for post-decoder sample regeneration of lost audio data through interpolation between neighbouring audio segments.

1) Intra-packet Redundancy using Reed-Solomon encoding: The output from the vorbis encoder is in UEP layers of base and residual data to which we would like to be able to assign different levels of priority. We divide our base layer into multiples of the RLC coding size minus the CRC hash, padding where necessary with residual data, and then apply a reed-Reed-Solomon FEC code [11] to generate parity data which is included in the packet. Reed Solomon(RS) codes have the property that for k source data symbols, an (n, k) code can be constructed where any k symbols out of the n will reconstruct the original data. Figure 12 illustrates the UEP scheme adopted by *vorbistreamer* depicting a sample with 4 RLC bucket lengths worth of data. The first 2 blocks are protected by the RS codes, and the second 2 blocks are unprotected. The figure also illustrates that multiple samples are stacked into one packet in order to minimise the amount of header overhead, the RS encoding process is then applied over the k protected data segments throughout the packet, generating $(n-k)$ parity symbols of data appended to the packet. Normally in wireless environments, it is beneficial to maintain a lower packet length in order to keep retransmission times, and consequently network jitter low, however where retransmission and error checking is disabled this has no impact, and therefore we minimise the header and FEC overhead by generating larger packets.

At the head of every packet, a small header is included that specifies the length of each sample appended to the packet. This header must be received intact, otherwise packet decoding is not possible.

2) Bucket interleaving: In section III-B.1 we demonstrated that errors statistically occur in bursts. In order to further optimise the recovery properties of the RS coding, we interleave the protected data with the unprotected data with the intention that the unprotected data will absorb some of the errors, thereby minimising the number of corrupt protected data segments. Figure 12 also illustrates the interleaving process, demonstrating the algorithmic approach toward the insertion of data. Each sample begins with a protected segment, and is interspersed with unprotected data until either type runs out, filling the rest of the sample with the remaining data. This process enables simple packet decoding, and in the event that certain samples are corrupted it ensures that only a single sample or a single data segment is affected. In practise, it was found that base layer data varies very little in size, and a fixed protected bucket size of 2 RLC blocks was utilised in order to simplify encoding further. Therefore in all network measurements presented in the next section, a fixed size of 58 bytes + 2 bytes of CRC hashes was used.

3) Vorbis Sample repair and interleaving across packets: For protected samples that are unrecoverable through the application level RS decoder, a gap in the audio stream is created since the sample must be discarded. In order to mask the effects of the loss, receiver based repair techniques can be applied. The easiest, and most effective methods are packet repetition (the previous sample is repeated), packet interpolation (a sample is derived through inference based on the two samples on either side) or noise insertion. The effectiveness of any repair technique is significantly increased by minimising the length of the gap created as a result of data loss. Therefore, we also consider the effects of packet level interleaving. Since error events do not statistically occur close together, even though bursts of errors occur within an event, we find that distributing the sample stream in time makes an important difference to the audio quality perception. At the expense of increased buffer delay to recover from sample re-ordering, the proximity of packet losses is a parameter that is minimised as much as possible within the buffer delay bounds.

V. EXPLOITING DIVERSITY IN WWAN ENVIRONMENTS

We consider the benefits of exploiting diversity in WWANs, beginning with an evaluation of the effectiveness of intra-packet redundancy to overcome burst errors within a single channel. For multi-channel network capacity, such as that provided by MAR, we further
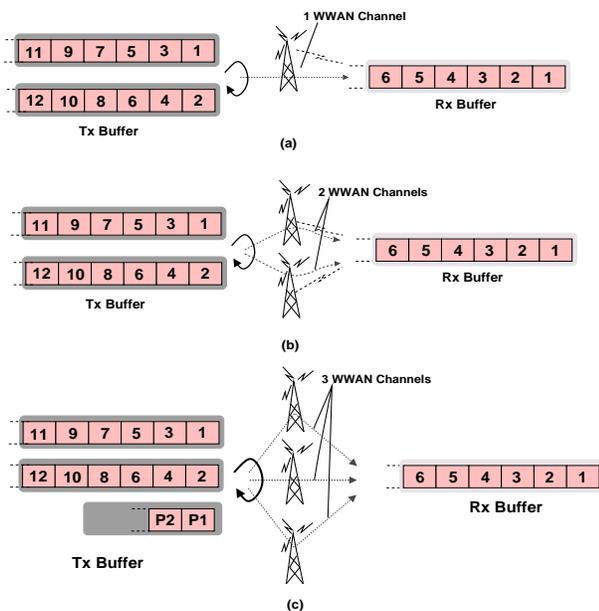
Fig. 13. The effects of interleaving over (a) 1 channel, (b) 2 channels and (c) 2 channels + 1 parity channel

FEC Scheme	Recovery (%)	Overhead (%)	Goodput (%)
1 Channel FEC 0	0	0	96.2
1 Channel FEC 1	39.5	2.84	97.7
1 Channel FEC 2	82.75	4.99	99.34
1 Channel FEC 3	92.17	6.59	99.7
1 Channel FEC 4	94.06	10.18	99.77
1 Channel FEC 5	99.7	12.93	99.9
cross-channel FEC	99.9	12.39	99.9

TABLE II

REDUNDANCY SCHEME PERFORMANCE

Interleaving Scheme	Average Interleave Gap	Goodput (%)
Single Channel, 1 buffer	1.2	98.93
Single Channel, 2 buffers	1.08	97.86
Round-robin Scheduler	1.04	97.0

TABLE III

INTERLEAVING SCHEME PERFORMANCE

consider three additional methods of exploiting channel diversity; utilising simple round-robin scheduling, applying cross-channel redundancy and utilising channel based Quality of Service. For each case we evaluate the relevant performance enhancements, primarily channel recovery and goodput (the residual sample rate after FEC decoding) and the burst length of lost samples (the sample stream after decoding). In the following results, all tests which utilise FEC redundant encoding and interleaving operate across unreliable channels. The only exception is the channel based Quality of Service approach which demonstrates the advantage of streaming part of the codec data (the base layer) across a reliable channel, and the rest over unreliable, but low jitter channels. For the purposes of these tests, the goodput of a sample is considered to be 100% if all protected segments are received or repaired, we do not measure the effect of corruption on the residue data.

The results presented are real network measurements derived from GPRS streaming tests utilising *vorbistreamer* to generate UEP multimedia content. We demonstrate that exploiting uncorrelated performance over separate channels has the most significant effect, however for single-channel devices we show that very reasonable performance can be derived through intelligent FEC and sample interleaving.

A. Single Channel Performance

We begin with diversity evaluation of a single channel, considering both the correlated and uncorrelated benefits

of the channel that can be exploited. We evaluated the effects of burst errors over the packets, and consequently measured the effectiveness of the intra-packet redundancy scheme. Our previous tests demonstrated that error events occur randomly and, within events, in bursts. We therefore try to overcome the effects of errors through adding redundancy to the packet and block interleaving. We also evaluated the effects of sample interleaving across packets in order to minimise the length of sample loss at the decoder.

Table II illustrates the trade-off in FEC overhead versus recovery performance of the algorithm. Some data is inevitably lost due to packet header corruption, hence we do not anticipate that the recovery performance would ever reach 100%. The results indicate that for an increase of only 6.59%, we achieve around 92% recovery.

The interleaving results presented in table III demonstrate that the average burst length can be reduced through interleaving by 10%, thereby lowering the perceptual impact of losses in the stream. By minimising the burst length, we can more effectively utilise packet repair techniques such as sample based interpolation. Figure 13(a) illustrates the benefit of sample interleaving across packets. We can see that the impact of a burst of errors over a single packet will be spread out in the receiver buffer.

B. Basic Multimedia Striping

Next we consider the benefits of channel diversity. In it's most basic form, MAR can be utilised to stripe data packets in a round robin fashion across interfaces in order to distribute the effects of sustained loss over the network. Our characterisation results in III-B.1 demonstrated that we should expect errors to occur in

random time periods (classified as an event), but in bursts within each event. We therefore anticipate that when an event occurs, there is a high probability that errors may cross packet boundaries, affecting more than one packet at a time. Since separate links are not expected to show correlation with respect to errors occurring at the same time, we therefore would anticipate that whilst striping sequential packets across links will not statistically improve the actual bit error rate, we would expect the average burst length of lost samples after decoding to be lower. Figure 13 illustrates the impact of interleaving on the system. In figure 13(a) we see a double output buffer being transmitted over a single wireless channel. Each sample is interleaved through the packetisation process, however the correlation between subsequent packets being lost is higher than the separate channel case. Figure 13(b) illustrates the impact of channel diversity, which lowers the probability of consecutive packets being lost due to the diversity properties of the separate channels.

Table III indicates that the interleaving over separate channels is indeed more effective than interleaving across a single channel. The results include measurements of a single channel without interleaving, the base case performance, sending data across a single channel with packet interleaving which is marginally better, and striping data across 2 channels with sample interleaving which produces the best results.

The goodput of the channel does not change significantly in either case. In these experiments we measured a slightly lower goodput from the dual channel tests, quite feasibly due to under-performance of one or other of the channels for some period of time during one of the tests, however even with poor goodput we see that the interleaving gap is lower, demonstrating that there is still a significant benefit in utilising more than 1 channel.

C. Cross-channel Redundancy

In order to leverage the diversity of the link in improving the resilience of the redundancy algorithm, we consider applying parity codes across channels rather than across packets. The motivation for this approach is that we would anticipate the probability of errors occurring simultaneously across links to be lower than the probability of errors occurring within a single packet. Here we reconsider the packetisation approach outlined in figure 12. Instead of adding redundancy blocks equivalent to an anticipated burst error rate within a packet, we assume that redundancy will be applied through cross-channel parity, so a single channel is designated as the parity channel for every set of packets. This is not a static

assignment for all parity data, however the principle is that for any n channels, the set of packets for which parity will be applied is $n-1$, e.g. for 3 channels, every 2 packets will also receive a parity packet. Due to the nature of the coding, however we only need to protect the base layer data, thereby lowering our parity rate. Figure 13(c) illustrates the parity coding channel which carries redundant information capable of repairing any losses that may occur on either of the other 2 channels. The channel selection is not important, the significant element is that the parity data is transmitted over a separate WWAN path.

Table II also illustrates the performance of the cross-channel redundancy scheme that we have measured from our network tests. The results demonstrate that utilising cross-channel diversity is highly effective compared to the intra-packet redundancy scheme on it's own. In all our 3 channel tests we have identified very few occurrences of simultaneous RLC bucket errors on 2 channels causing a sample to be lost. One of the significant advantages of the scheme over intra-channel redundancy is that in the event of total packet loss due to header corruption, we are still able to regenerate base layer samples through parity data, thereby providing a 100% recovery of data when errors do not impact the other channels. In our tests, only 1.8% of the error events occurred simultaneously across channels, and in 99.9% of cases the RLC blocks affected in any of the packets were different, thereby enabling full recovery of the data.

There is a trade-off for the cross-channel redundancy scheme which can be witnessed in the data overhead generated by the parity codes. As the number of channels increases, this value could be reduced since the parity to data ratio would be lower, however we have not been able to measure the effectiveness of the parity recovery for larger numbers of channels. Even in the 3 channel case however it does still compare favourably against the strongest intra-packet redundancy scheme, generating an overhead of around 12% for the highest recovery rate.

D. Channel Based QoS

The final case that we consider utilises link reliability controls to provide different channel quality of service. In this case we distribute a low bandwidth protected data stream over a link operating in reliable mode, and the higher bandwidth unprotected data over unreliable channels. In the simplest case at least 2 channels must be available. Based on figure 7 it is clear the key streaming constraint will be the amount of initial buffering required to support the target bit rate of the protected channel. Figure 10 demonstrates the bit rate for the protected

data stream as a function of the overall bandwidth, approximately one third of the combined rate, thereby enabling us to model the constraints of the system for different target bit rates and clip lengths. An example average streaming rate of 45 Kbit/s with a clip length of around 300 seconds would therefore require an initial buffer delay for the protected channel of around 3 seconds, in contrast to the original requirement for the same overall streaming rate of 144 seconds.

The diversity that can be exploited in this case is the jitter and the bandwidth of the reliable versus unreliable channel. The higher rate stream generated by the residual data from the codec does not impose any delay constraints on the protected data stream since the link is operating in an unreliable mode, and we find that packets experiencing loss do not significantly impact the stream due to the UEP nature of the data. Since the protected stream is low bandwidth, we also note that the packet size can also be reduced at the expense of some header overhead to lower the propagation latency of the data. This presents an extremely attractive option for supporting high data rates with reasonably good quality and low stream initialisation delay. Assuming the buffer does not under run, the goodput for this channel would be 100% where we assume that all base-layer data has been received. The unreliable data would be subject to the same conditions as the other 2 channel diversity schemes.

VI. Conclusions

In this paper we demonstrated that WWAN links are highly variable due to the strict reliability constraints that are configured by default. We measured the variation imposed by ARQ at the link layer, and assessed the buffer requirements for multimedia applications. For more interactive applications that require less buffering constraints in exchange for higher data loss, we evaluated the benefits of application level forward error correction to provide unequal error protection of the media stream.

We designed a multimedia streaming application that implements UEP coding for WWAN streaming, and conducted real network measurements of the streams over a GPRS network. We demonstrated that application level error detection is an effective means to better utilise channel resources by only discarding portions of data packets that are corrupt, thereby creating the ability to provide UEP support at the application level.

We considered the impact of diversity in the wide area and evaluated the benefits on multimedia streaming. In particular, we demonstrate that *intra*-channel diversity is useful, and we presented results to show that for a small

overall increase in bit rate (6.59%) we can achieve a channel goodput of 99.7% of the original stream.

We further considered the benefits of exploiting *inter*-channel redundancy through MAR, the Mobile Access Router through three techniques. The first technique of basic multimedia striping demonstrated that providing packet interleaving across parallel channels reduces the average burst loss length by 14% over the basic single channel case without impacting overall bandwidth or goodput of the channel. The second technique of cross-channel redundancy provides the most significant results of the diversity experiments. By sending parity data on a separate channel than the data, rather than appended to the packet in the same channel, we achieved 99.9% recovery for an overhead of 12.39%. This is due to the low probability that parallel channels will exhibit correlated behaviour, and simultaneous errors occur in the same place. The result is that sufficient redundant information is statistically more likely to be available to repair any blocks that are affected. Thirdly, we presented a channel-based QoS approach towards unequal error protection of the data stream. By sending only the base layer information across a reliable channel, and the higher layer data over an unreliable channel, we significantly reduce the buffering constraints of the link, and can statistically support a higher data rate.

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*, vol. 2, no. 3, 1999.
[2] R. Kalden, I. Meirick and M. Meyer, "Wireless Internet Access based on GPRS," in *IEEE Personal Communications*, 2000.
[3] "ETSI TR 101 115 digital cellular telecommunications system (phase 2+); background for radio frequency requirements," Tech. Rep., ITU-T, May 2000.
[4] L.A. Larzon et al, "The udp-lite protocol," Internet Draft, Internet Engineering Task Force, Dec. 2002, Work in progress.
[5] G. Pelletier, "Robust header compression for udp-lite," Internet Draft, Internet Engineering Task Force, Apr. 2003, Work in progress.
[6] R. Reine and G. Fairhurst, "Mpeg-4 and udp-lite for multimedia transmission," in *PGNet 2003, Liverpool, UK*, June 2003.
[7] A. Singh, A. Konrad, and A. Joseph, "Performance evaluation of udp lite for cellular video," in *Proceedings of NOSSDAV*, June 2001.
[8] P. Rodriguez, R.Chakravorty, J. Chesterfield, Ian Pratt, and S. Banerjee, "Mar: A commuter router infrastructure for the mobile internet," in *MOBISYS - 2nd International Conference on Mobile Systems, Applications and Services on Network Computing and Applications*, June 2004.
[9] "ITU-T recommendation g.114, (05/00): One-way transmission time," Tech. Rep., ITU-T, May 2000.
[10] The Ogg Vorbis multimedia coding format, http://www.xiph.org/ogg/vorbis, ," .
[11] I.S. Reed and G. Solomon, "Polynomial codes over certain finite fields," 1960, pp. 300–304.