A Survey of Packet-Loss Recovery Techniques for Streaming Audio

Colin Perkins Orion Hodson Vicky Hardman
Department of Computer Science
University College London
Gower Street
London WC1E 6BT

10 August 1998

Abstract

We survey a number of packet-loss recovery techniques for streaming audio applications operating using IP multicast. We begin with a discussion of the loss and delay characteristics of an IP multicast channel and from this show the need for packet loss recovery. Recovery techniques may be divided into two classes: sender- and receiver-based. We compare and contrast several sender-based recovery schemes: forward error correction (both media specific and media independent) interleaving and retransmission. In addition a number of error concealment schemes are discussed. We conclude with a series of recommendations for repair schemes to be used, based on application requirements and network conditions.

1 Introduction

The development of IP multicast and the Internet multicast backbone has led to be emergence of a new class of scalable audio/video conferencing applications. These are based on the lightweight sessions model [11,17] and provide efficient multi-way communication which scales from two to several thousand participants. The network model underlying these applications differs significantly from the tightly coupled approach in use for traditional conferencing systems. The advantage of this new, loosely-coupled approach to conferencing is scalability; the disadvantage is unusual channel characteristics which require significant work to achieve robust communication.

In this paper, we discuss the loss characteristics of such an IP multicast channel and how these affect voice communication. Following this, we examine a number of techniques for recovery from packet loss on the channel. These represent a broad cross-section of the range of applicable techniques, both sender-driven and receiver-based, and have been implemented in a wide range of conferencing applications, giving operational experience as to their behaviour. The paper concludes with an overview of the scope of applicability of these techniques and a series of recommendations for designers of packet based audio conferencing applications.

A number of surveys have previously been published in the area of reliable multicast and IP-based audio/video transport. The work by Obraczka [28] and Levine and Garcia-Luna-Aceves [20] is limited to the study of fully-reliable transport and does not consider real-time delivery. The survey by Carle and Biersack [6] discusses real-time IP-based audio/video applications and techniques for error recovery in this environment. However, that work neglects receiver-based error concealment techniques and focuses on sender-driven mechanisms for error correction.

Sender-driven and receiver-based repair are complementary techniques, and applications should use both methods to achieve the best possible performance. In contrast to previous work, we limit the focus of our paper to streaming audio applications and discuss *both* sender driven repair and receiver-based error concealment techniques.

2 Multicast Channel Characteristics

The concept of IP multicast was proposed by Deering [8] to provide a scalable and efficient means by which datagrams may be distributed to a group of receivers. This is achieved by imposing a level of indirection between

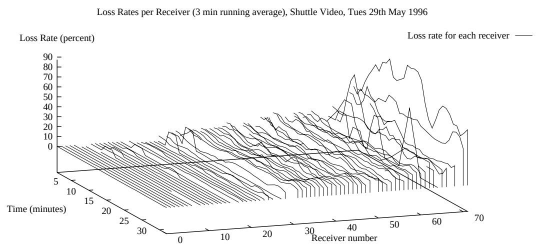

Figure 1: Observed loss rates in a large multicast conference (from [14]).

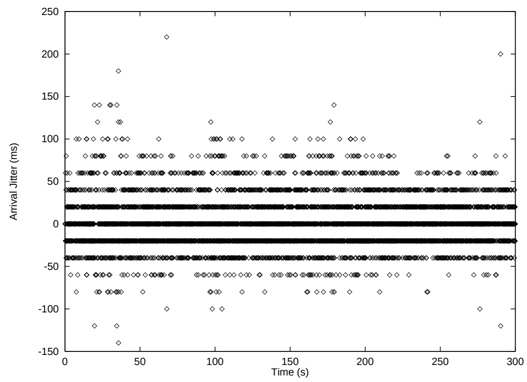

Figure 2: Observed variation in end-to-end delay as seen by an Mbone audio tool (20ms timing quantization).

senders and receivers: packets are sent to a group address, receivers listen on that same address and the network conspires to deliver packets. Unless provided by an application-level protocol, the senders and receivers are decoupled by the group address: a sender does *not* know the set of hosts which will receive a packet. This indirection is important: routing decisions and recovery from network outages are purely local choices which do not have to be communicated back to the source of packets, or to any of the receivers, enhancing scalability and robustness significantly.

Internet conferencing applications, based on IP multicast, typically employ an application-level protocol to provide *approximate* information as to the set of receivers and reception quality statistics. This protocol is the Real-time Transport Protocol, RTP [40].

The portion of the Internet which supports IP multicast is known as the multicast backbone (Mbone). Although some parts of the Mbone operate over dedicated links, the distinguishing feature is the presence of multicast routing support: multicast traffic typically shares links with other traffic. A number of attempts have been made to characterise the loss patterns seen on the Mbone [3, 4, 14, 47]. Although these results vary somewhat, the broad conclusion is clear: in a large conference it is inevitable that some receivers will experience packet loss. This is most clearly illustrated by the work of Handley [14] which tracks RTP reception report statistics for a large multicast session over several days. A typical portion of this trace is illustrated in figure 1. It can be seen that *most* receivers experience loss in the range of 2-5%, with some smaller number seeing significantly higher loss rates. The overwhelming cause of loss is due to congestion at routers. It is therefore not surprising that there is a correlation between the bandwidth used and the amount of loss experienced [14, 16] and the underlying loss rate varies during the day.

A multicast channel will typically have relatively high latency and the variation in end-to-end delay may be large. This is clearly illustrated in figure 2 which shows the inter-arrival jitter for a series of packets sent from the University of Oregon to University College London on 10 August 1998. This delay variation is a reason for concern when developing loss tolerant, real-time applications, since packets delayed too long will have to be discarded in order to meet the application's timing requirements, leading to the appearance of loss. This problem is more acute for interactive applications: if interactivity is unimportant a large playout delay may be inserted to allow for these delayed packets. This problem and algorithms for playout buffer adaption are studied further in [2, 25, 32].

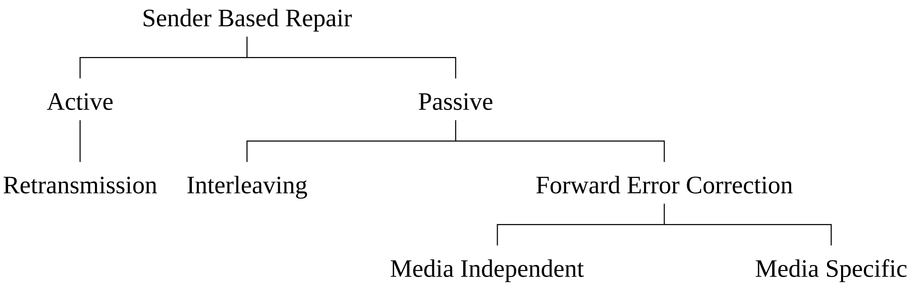

Figure 3: A Taxonomy of Sender Based Repair Techniques

Unlike other communications media, IP multicast allows for the trade-off between quality and interactivity to be made independently for each receiver in a session, since this is a local choice only and is not communicated to the source of the data. A session may exist with most participants acting as passive observers (high latency, low loss), but with some active participants (low latency, higher loss).

It should be noted that the characteristics of an IP multicast channel are significantly different from those of an ATM or ISDN channel. The techniques discussed herein do not necessarily generalise to conferencing applications built upon such network technologies.

The majority of these techniques are applicable to unicast IP, although the scaling and heterogeneity issues are clearly simpler in this case.

3 Sender Based Repair

We discuss a number of techniques which require the participation of the sender of an audio stream to achieve recovery from packet loss. These techniques may be split into two major classes: active retransmission and passive channel coding. It is further possible to subdivide the set of channel coding techniques, with traditional forward error correction (FEC) and interleaving-based schemes being used. Forward error correction data may be either media independent, typically based upon exclusive-or operations, or media specific based on the properties of an audio signal. This taxonomy is summarised in figure 3.

In order to simplify the following discussion we distinguish a *unit* of data from a *packet*. A unit is an interval of audio data, as stored internally in an audio tool. A packet comprises one or more units, encapsulated for transmission over the network.

3.1 Forward Error Correction

A number of forward error correction techniques have been developed to repair losses of data during transmission. These schemes rely on the addition of repair data to a stream, from which the contents of lost packets may be recovered. There are two classes of repair data which may be added to a stream: those which are independent of the contents of that stream and those which use knowledge of the stream to improve the repair process.

3.1.1 Media Independent FEC

There has been much interest in the provision of media independent FEC using block, or algebraic, codes to produce additional packets for transmission to aid the correction of losses. Each code takes a codeword of k data packets and generates $n - k$ additional check packets for the transmission of n packets over the network.

A large number of block coding schemes exists and we discuss only two cases, parity coding and Reed-Solomon coding, as these are currently proposed as an RTP payload[37]. These block coding schemes were originally designed for the detection and correction of errors within a stream of transmitted bits and so the check bits were generated from a stream of data bits. In packet streams we are concerned with the loss of entire packets and

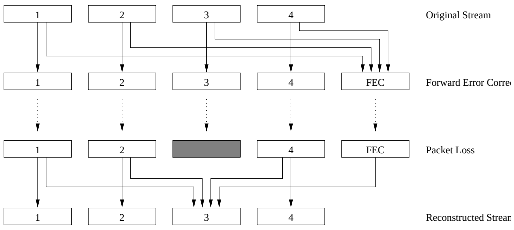

Figure 4: Repair using parity FEC

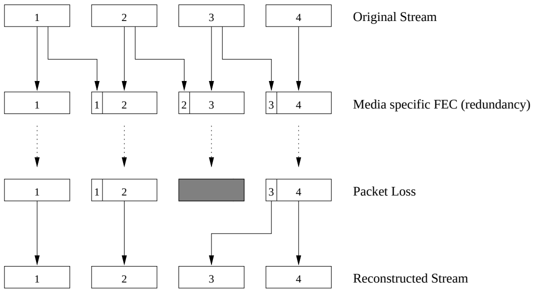

Figure 5: Repair using media specific FEC

so we apply block coding schemes across the corresponding bits in blocks of packets. Hence i'th bit in a check packet is generated from the i'th bit of each of the associated data packets.

In parity coding, the exclusive-or (XOR) operation is applied across groups of packets to generate corresponding parity packets. An example of this has been implemented by Rosenberg [36]. In this scheme, 1 parity packet is transmitted after every $n-1$ data packets. Provided there is just one loss in every n packets, then that loss is recoverable. This is illustrated in figure 4. Many different parity codes may be derived by XOR'ing different combinations of packets, a number of these were proposed by Budge *et al.* and summarised by Rosenberg & Schulzrinne [37].

Reed-Solomon codes [22, 34] are renowned for their excellent error correcting properties. Encoding is based upon the properties of polynomials over particular number bases. Essentially RS encoders take a set of codewords and use these as coefficients of a polynomial, $f(x)$. The transmitted codeword is determined by evaluating the polynomial for all non-zero values of x over the number base. Whilst this may sound complicated, the encoding procedure is relatively straightforward and optimised decoding procedures such as the Berlekamp-Massey algorithm [1, 21] are available. In the absence of packet losses decoding carries the same computational cost as encoding, but when losses occur it is significantly more expensive.

There are several advantages to forward error correction schemes. The first is that they are media independent: the operation of the forward error correction does not depend on the contents of the packets and the repair is an exact replacement for a lost packet. In addition, the computation required to derive the error correction packets is relatively small and simple to implement. The disadvantages of these schemes are the additional delay imposed, increased bandwidth and difficult decoder implementation.

3.1.2 Media Specific FEC

A simple means to protect against packet loss is to transmit each unit of audio in multiple packets. If a packet is lost then another packet containing the same unit will be able to cover the loss. The principle is illustrated in figure 5. This approach has been advocated by Hardman *et al.* [15] and Bolot *et al.* [3] for use on the Mbone and extensively simulated by Podolsky *et al.* [31].

The first transmitted copy of the audio data is referred to as the primary encoding and subsequent transmissions as secondary encodings. It is the sender's decision whether the secondary audio encodings should be the same coding scheme as the primary, although usually the secondary is encoded using a lower-bandwidth, lower-quality, encoding than the primary.

The choice of encodings is a difficult problem and depends on both the bandwidth requirements and the computational complexity of the encodings. Erdöl *et al.* [9] consider using short term energy and zero crossing measurements as their secondary scheme. When loss occurs the receiver then interpolates an audio signal about the crossings using the short-term energy measurements. The advantage of this scheme is that is uses computationally cheap measures and can be coded compactly. However, it can only cover short periods of loss due to the crude nature of the measures. Hardman *et al.* [15] and Bolot *et al.* [3] advocate the use of low bit-rate analysis-by-synthesis codecs, such as LPC (2.4-5.6kbps) and GSM (13.2kbps), which although computationally more demanding can tolerably cover the loss periods experienced on the Internet.

If the primary encoding consumes considerable processing power, but has sufficient quality and low bandwidth, then the secondary encodings may be the same as the primary. An example of this is the ITU G.723.1 [41] codec which consumes a considerable fraction of today's desktop processing power, but has a low bandwidth (5.3/6.3kbps).

The use of media specific FEC incurs an overhead in terms of packet size. For example the use of 8kHz PCM μ-law (64kbps) as the primary compression scheme and GSM [26] (13kbps) as the secondary results in a 20% increase in the size of the data portion of each packet. Like media independent FEC schemes, the overhead of media specific FEC is variable. However, unlike those schemes, the overhead of media specific FEC may be reduced without affecting the number of losses which may be repaired, instead the quality of the repair varies with the overhead. To reduce the overhead, approximate repair is used, which is acceptable for audio applications.

It should be noted that it may often not be necessary to transmit media specific FEC for every packet. Speech signals have transient stationary states that can cover 80ms. Viswanathan *et al.* [43] describe LPC codecs where units of speech are only transmitted if the parameters of the codec are deemed to have changed sufficiently and achieve a 30% saving bandwidth for the same quality. A similar decision could be made about whether to transmit the FEC data, although this is likely to be codec specific.

Unlike many of the other sender-based techniques discussed, the use of media specific FEC has the advantage of low-latency, with only a single-packet delay being added. This makes it suitable for interactive applications, where large end-to-end delays cannot be tolerated. If large end-to-end delay can be tolerated, it is possible to delay the redundant copy of a packet, achieving improved performance in the presence of burst losses [19].

At the time of writing, media specific FEC is supported by a number of Mbone audio conferencing tools. The standard RTP payload format for media specific FEC is described in [30].

3.1.3 Congestion Control

The addition of FEC repair data to a media stream is an effective means by which that stream may be protected against packet loss. However, application designers should be aware that the addition of large amounts of repair data when loss is detected will increase network congestion and hence packet loss, leading to a worsening of the problem which the use of FEC was intended to solve.

This is particularly important when sending to large multicast groups, since network heterogeneity causes different sets of receivers to observe widely varying loss rates: low-capacity regions of the network suffer congestion, whilst high-capacity regions are under-utilised.

At the time of writing, there is no standard solution to this problem. There have been a number of contributions which show the likely form the solution will take [23, 35, 42]. These typically use some form of layered encoding of data sent at different rates over multiple multicast groups, with receivers joining and leaving groups in response to long-term congestion and with FEC being employed to overcome short-term transient congestion.

Such a scheme pushes the burden of adaption from the sender of a stream to the receivers, which choose the number of layers (groups) they join based on the packet loss rate they observe. Since the different layers contain data sent at different rates, receivers will receive different quality of service depending on the number of layers they are able to join. The precise details of these schemes are beyond the scope of this paper, the reader is referred to the above references for further details.

Layered encoding schemes are expected to provide a congestion control solution suitable for streaming audio applications. However, this work is not yet complete and it is important to give some advice to authors of streaming audio tools as to the behaviour which is acceptable, until such congestion control mechanisms can be deployed.

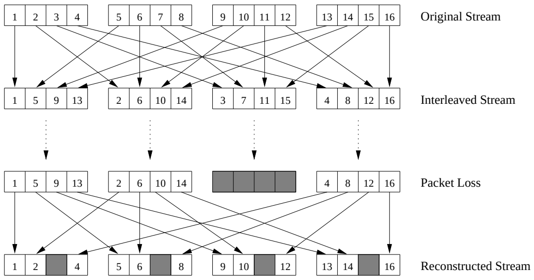

Figure 6: Interleaving units across multiple packets.

It has been suggested [29] that one heuristic suitable for determining reasonable behaviour for unicast streaming media tools is to adapt the transmission rate to the approximate throughput a TCP/IP stream would achieve over the same path. Since TCP/IP flows are the dominant form of traffic in the Internet, this would be roughly fair to existing traffic. Clearly such a scheme would not work for a multicast flow (although a worst case or average throughput to the set of receivers could be derived and used as the basis for adaption) and clearly it does not capture the dynamic behaviour of the connection, merely the average behaviour, but it does provide one definition of reasonable behaviour in the absence of real congestion control. In the long-term, effective congestion control must be developed.

Note that the need for congestion control is *not* specific to FEC encoded audio streams. It should be considered for all streaming media.

3.2 Interleaving

When the unit size is smaller than the packet size and end-to-end delay is unimportant, interleaving is a useful technique for reducing the effects of loss [33]. Units are resequenced before transmission, so that originally adjacent units are separated by a guaranteed distance in the transmitted stream and returned to their original order at the receiver. Interleaving disperses the effect of packet losses. If, for example, units are 5ms in length and packets 20ms (ie: 4 units per packet), then the first packet would contain units 1, 5, 9, 13; the second packet would contain units 2, 6, 10, 14; and so on, as illustrated in figure 6.

It can be seen that the loss of a single packet from an interleaved stream results in multiple small gaps in the reconstructed stream, as opposed to the single large gap which would occur in a non-interleaved stream. This spreading of the loss is important for two similar reasons: firstly, Mbone audio tools typically transmit packets which are similar in length to phonemes in human speech. Loss of a single packet will therefore have a large effect on the intelligibility of speech. If the loss is spread out so that small parts of several phonemes are lost, it becomes easier for people to mentally patch-over this loss [24] resulting in improved perceived quality for a given loss rate. In a somewhat similar manner, error concealment techniques perform significantly better with small gaps, since the amount of change in the signal's characteristics is likely to be smaller.

The majority of speech and audio coding schemes can have their output interleaved and may be modified to improve the effectiveness of interleaving. The disadvantage of interleaving is that it increases latency. This limits the use of this technique for interactive applications, although it performs well for non-interactive use. The major advantage of interleaving is that it does not increase the bandwidth requirements of a stream.

3.3 Retransmission

Interactive audio applications have tight latency bounds and end-to-end delays need to be less than 250ms [5]. For this reason such applications do not typically employ retransmission based recovery for lost packets. If larger end-to-end delays can be tolerated the use of retransmission to recover from loss becomes a possibility.

A widely deployed reliable multicast scheme based on the retransmission of lost packets is Scalable Reliable Multicast [11]. When a member of an SRM session detects loss, it will wait a random amount of time, determined by its distance from the original source of the lost data and then multicast a repair request. The retransmission timer is calculated such that, although a number of hosts may miss the same packet, the host closest to the point of failure will most likely timeout first and issue the retransmission request. Other hosts which also see the loss, but receive the retransmission request message, suppress their own request to avoid message implosion.[1] On receiving a retransmission request, any host with the requested data may reply: once again a timeout is used based on the distance of that host from the sender of the retransmit request, to prevent reply implosion. The timers are calculated such that typically only one request and one retransmission will occur for each lost packet.

Whilst SRM and related protocols are well suited for reliable multicast of data objects, they are not generally suitable for streaming media such as audio. This is because they do not bound the transmission delay and, in the presence of packet loss, may take an arbitrary amount of time. A large number of reliable multicast protocols have been defined (see [20] for a survey) which are similarly unsuitable for streaming media and hence are not studied here. For similar reasons, TCP is not appropriate for unicast streaming audio.

That is not to say that retransmission based schemes cannot be used for streaming media, in some circumstances. In particular, protocols which use retransmission but bound the number of retransmission requests allowed for a given unit of data may be appropriate. Such retransmission based schemes work best when loss rates are relatively small. As loss rates increase, the overhead due to retransmission request packets increases. Eventually a cross-over point is reached, beyond which the use of FEC becomes more effective. It has been observed in large Mbone sessions that *most* packets are lost by at least one receiver [14]. Indeed, in their implementation of an SRM-like protocol for streaming audio [46], Xu *et al.* note that "In the worst case, for every multicast packet, at least one receiver does not receive the packet, which means that *every* packet needs to be transmitted to the whole group at least twice". In cases such as this, it is clear that the use of retransmission is probably only appropriate as a secondary technique to repair losses which are not repaired by FEC.

An alternative combination of FEC and retransmission has been studied by Nonnenmacher *et al.* [27]. This work takes the approach of using parity FEC packets to repair multiple losses with a single retransmission, achieving substantial bandwidth savings relative to pure retransmission.

Further, the retransmission of a unit of audio does not need to be identical to the original transmission: the unit can be recoded to a lower bandwidth, if the overhead of retransmission is thought to be problematic. There is a natural synchrony with redundant transmission and a protocol may be derived in which both redundant and retransmitted units may be accommodated. This allows receivers which cannot participate in the retransmission process to benefit from retransmitted units, if they are operating with a sufficiently large playout delay.

The use of retransmission allows for an interesting trade-off between the desired playback quality and the desired degree of latency inherent in the stream. Within a large session, the amount of latency which can be tolerated varies greatly for different participants: some users desire to participate closely in a session and hence require very low latency, whereas others are content to observe and can tolerate much higher latency. Those participants who require low-latency must receive the media stream without the benefit of retransmission-based repair (but may use FEC). Others gain the benefit of the repair, but at the expense of increased delay.

4 Error Concealment

We consider a number of techniques for error concealment which may be initiated by the receiver of an audio stream and which do not require assistance from the sender. These techniques are of use when sender based recovery schemes fail to correct all loss, or when the sender of a stream is unable to participate in the recovery.

Error concealment schemes rely on producing a replacement for a lost packet which is similar to the original. This is possible since audio signals, and in particular speech, exhibit large amounts of short-term self similarity. As such, these techniques work for relatively small loss rates ($\lesssim 15\%$) and for small packets (4-40ms). When the loss length approaches the length of a phoneme (5-100ms) these techniques break down, since whole phonemes may be missed by the listener.

It is clear that error concealment schemes are not a substitute for sender-based repair, but rather work in tandem with it. A sender-based scheme is used to repair most losses, leaving a small number of isolated gaps to be

[1] The SRM protocol is designed to scale to very large groups. If request suppression was not used, a lost packet near the source would trigger simultaneous retransmission requests from many group members which could overwhelm the sender (consider the effects in a group with many hundreds, or thousands, of members).

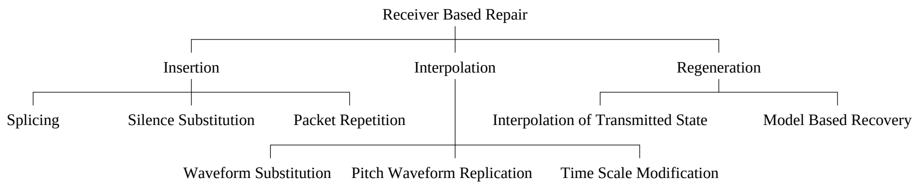

Figure 7: A Taxonomy of Error Concealment Techniques

repaired. Once the effective loss rate has been reduced in this way, error concealment forms a cheap and effective means of patching over the remaining loss.

A taxonomy of various receiver-based recovery techniques is given in figure 7. It can be seen that these techniques split into three categories:

Insertion based schemes repair losses by inserting a fill-in packet. This fill-in is usually very simple: silence or noise are common, as is repetition of the previous packet. Such techniques are easy to implement but, with the exception of repetition, have poor performance.

Interpolation based schemes use some form of pattern matching and interpolation to derive a replacement packet which is expected to be similar to the lost packet. These techniques are more difficult to implement and require more processing when compared with insertion based schemes. Typically performance is better.

Regeneration based schemes derive the decoder state from packets surrounding the loss and generate a replacement for the lost packet from that. This process is expensive to implement but can give good results.

The following sections discuss each of these categories in turn. This is followed, in section 4.4, by a summary of the range of applicability of these techniques.

4.1 Insertion-Based Repair

Insertion-based repair schemes derive a replacement for a lost packet by inserting a simple fill-in. The simplest case is splicing where a zero-length fill-in is used; an alternative is silence substitution where a fill-in with the duration of the lost packet is substituted, to maintain the timing of the stream. Better results are obtained by using noise or a repeat of the previous packet as the replacement.

The distinguishing feature of insertion based repair techniques is that the characteristics of the signal are not used to aid reconstruction. This makes these methods simple to implement but results in generally poor performance.

4.1.1 Splicing

Lost units can be concealed by splicing together the audio on either side of the loss; no gap is left due to a missing packet, but the timing of the stream is disrupted. This technique has been evaluated by Gruber and Strawczynski [13] and shown to perform poorly. Low loss rates and short clipping lengths (4-16ms) faired best, but the results were intolerable for losses above 3%.

The use of splicing can also interfere with the adaptive playout buffer required in a packet audio system, because it makes a step reduction in the amount of data available to buffer. The adaptive playout buffer is used to allow for the reordering of misordered packets and removal of network timing jitter and poor performance of this buffer can adversely affect the quality of the entire system.

It is clear, therefore, that splicing together audio on either side of a lost unit is not an acceptable repair technique.

4.1.2 Silence Substitution

Silence substitution fills the gap left by a lost packet with silence in order to maintain the timing relationship between the surrounding packets. It is only effective with short packet lengths ($\lesssim$ 4ms) and low loss rates ($\lesssim$ 2%) [18], making it suitable for interleaved audio over low loss paths.

The performance of silence substitution degrades rapidly as packet sizes increase and quality is unacceptably bad for the 40ms packet size in common use in network audio conferencing tools. Despite this, the use of silence substitution is widespread, primarily because it is simple to implement.

4.1.3 Noise Substitution

Since silence substitution has been shown to perform poorly an obvious next choice is noise substitution, where, instead of filling in the gap left by a lost packet with silence, background noise is inserted instead.

A number of studies of the human perception of interrupted speech have been conducted, for example that by Warren [44]. These have shown that *phonemic restoration*, the ability of the human brain to subconsciously repair the missing segment of speech with the correct sound, occurs for speech repair using noise substitution but not for silence substitution.

In addition, when compared to silence, the use of white noise has been shown to give both subjectively better quality [24] and improved intelligibility [44]. It is, therefore, recommended as a replacement for silence substitution.

As an extension for this, a proposed future revision of the RTP profile for audio/video conferences [39] allows for the transmission of *comfort noise* indicator packets. This allows the communication of the loudness level of the background noise to be played, allowing for better fill-in information to be generated.

4.1.4 Repetition

Repetition replaces lost units with copies of the unit that arrived immediately before the loss. It has low computational complexity and performs reasonably well. The subjective quality of repetition can be improved by gradually fading repeated units. The GSM system, for example, advocates the repetition of the first 20ms with the same amplitude and followed by fading the repeated signal to zero amplitude over the next 320ms [10].

The use of repetition with fading is a good compromise between the other poorly performing insertion-based concealment techniques and the more complex interpolation-based and regenerative concealment methods.

4.2 Interpolation Based Repair

A number of error concealment techniques exist which attempt to interpolate from packets surrounding a loss to produce a replacement for that lost packet. The advantage of interpolation based schemes over insertion based techniques is that they account for the changing characteristics of a signal.

4.2.1 Waveform Substitution

Waveform substitution uses audio before, and optionally after, the loss to find a suitable signal to cover the loss. Goodman *et al.* [12] studied the use of waveform substitution in packet voice systems. They examined both one- and two-sided techniques that use templates to locate suitable pitch patterns either side of the loss. In the one sided scheme the pattern is repeated across the gap but with the two-sided schemes interpolation occurs. The two-sided schemes generally performed better than one sided schemes and both work better than silence substitution and packet repetition.

4.2.2 Pitch Waveform Replication

Wasem *et al.* [45] present a refinement on waveform substitution by using a pitch detection algorithm either side of the loss. Losses during unvoiced speech segments are repaired using packet repetition and voiced losses repeat a waveform of appropriate pitch length. The technique, known as pitch waveform replication, was found to work marginally better than waveform substitution.

4.2.3 Time Scale Modification

Time scale modification allows the audio either side of the loss to be stretched across the loss. Sanneck *et al.* [38] present a scheme that finds overlapping vectors of pitch cycles either side of the loss, offsets them to cover the loss and averages them where they overlap. Although computationally demanding, the technique appears to work better than both waveform substitution and pitch waveform replication.

4.3 Regeneration Based Repair

Regenerative repair techniques use knowledge of the audio compression algorithm to derive codec parameters, such that audio in a lost packet can be synthesised. These techniques are necessarily codec dependent but perform well because of the large amount of state information used in the repair. Typically, they are also somewhat computationally intensive.

4.3.1 Interpolation of Transmitted State

For codecs based on transform coding or linear prediction, it is possible that the decoder can interpolate between states. For example, the ITU G.723.1 speech coder [41] interpolates the state of the linear predictor coefficients either side of short losses and uses either a periodic excitation the same as the previous frame, or gain matched random number generator, depending on whether the signal was voiced or unvoiced. For longer losses, the reproduced signal is gradually faded. The advantages of codecs that can interpolate state rather than recoding the audio either side of the loss is that there is are no boundary effects due to changing codecs and the computational load remains approximately constant. However, it should be noted that codecs where interpolation maybe applied typically have high processing demands.

4.3.2 Model-Based Recovery

In model-based recovery the speech on one, or both, sides of the loss is fitted to a model that is used to generate speech to cover the period loss. In recent work by Chen and Chen [7], interleaved μ-law encoded speech is repaired by combining the results of autoregressive analysis on the last received set of samples with an estimate of the excitation made for the loss period. The technique works well for two reasons: the size of the interleaved blocks (8/16ms) is short enough to ensure the speech characteristics of the last received block have a high probability of being relevant. The majority of low bit-rate speech codecs use an autoregressive model in conjunction with an excitation signal.

4.4 Summary

It is difficult to obtain an accurate characterisation of the performance and complexity of error concealment techniques since the measurements which may be performed are, due to the nature of the repair, subjective. However, based on our experience, we believe that figure 8 provides a reasonable illustration of the quality/complexity tradeoff for the different repair techniques discussed.

The computation required to perform the more advanced repair techniques increases greatly, relative to the simpler repair options. However, the improvement in quality achieved by these schemes is incremental, at best. For this reason, the use of packet repetition with fading is recommended as offering a good compromise between

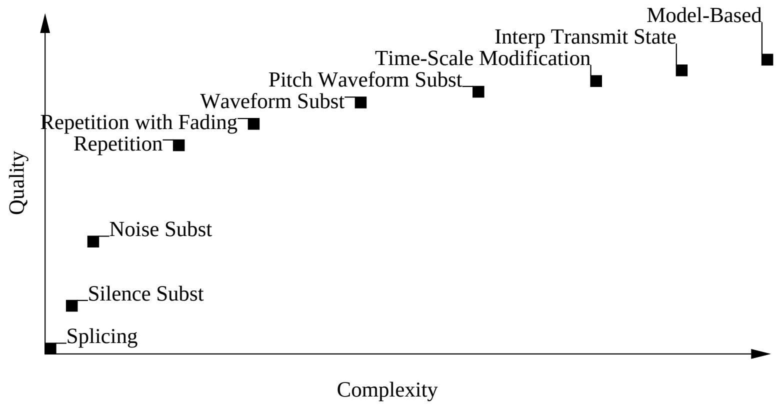

Figure 8: Rough quality/complexity tradeoff for error concealment

achieved quality and excessive complexity. For comparison, an example using packet repetition and waveform substitution can be seen in figure 9.

Several of these techniques can be applied using data from one or both sides of the loss. Many audio and speech coders assume continuity of the decoder state. When a loss occurs, it may not be possible to decode audio data on both sides of the loss for use in the repair since the decoded audio after the loss may be starting from an inappropriate state. In addition, two-sided operations incur greater processing overhead and usually represent a marginal improvement. In the majority of cases one-sided repair is sufficient.

5 Recommendations

In this final section, we suggest which of these techniques should be considered for IP multicast applications in some common scenarios. We discuss the trade-off between achieving good performance with acceptable cost/complexity.

5.1 Non-Interactive Applications

For one-to-many transmissions, in the style of radio broadcasts, latency is of considerably less importance than quality. In addition, bandwidth efficiency is a concern since the receiver set is likely to be diverse and the group may include members behind low-speed links. The use of interleaving is compatible with both of these requirements and is strongly recommended.

Although interleaving drastically reduces the audible effects of lost packets, some form of error concealment will still be needed to compensate. In this case the use of a simple repair scheme, such as repetition with fading, is acceptable and will give good quality.

Retransmission based repair is not appropriate for a multicast session, since the receiver set is likely to be heterogeneous. This leads to many retransmission requests for different packets and a large bandwidth overhead due to control traffic. For unicast sessions, retransmission is more acceptable, particularly in low-loss scenarios.

A media independent FEC scheme will perform better than a retransmission based repair scheme, since a single FEC packet can correct many different losses and there is no control traffic overhead. The overhead due to the FEC data itself still persists, although this may be acceptable. In particular, FEC protected streams allow for exact repair, whilst repair of interleaved streams is only approximate.

5.2 Interactive Applications

For interactive applications, such as IP telephony, the principle concern is minimising end-to-end delay. It is acceptable to sacrifice some quality to meet delay requirements, provided that the result is intelligible.

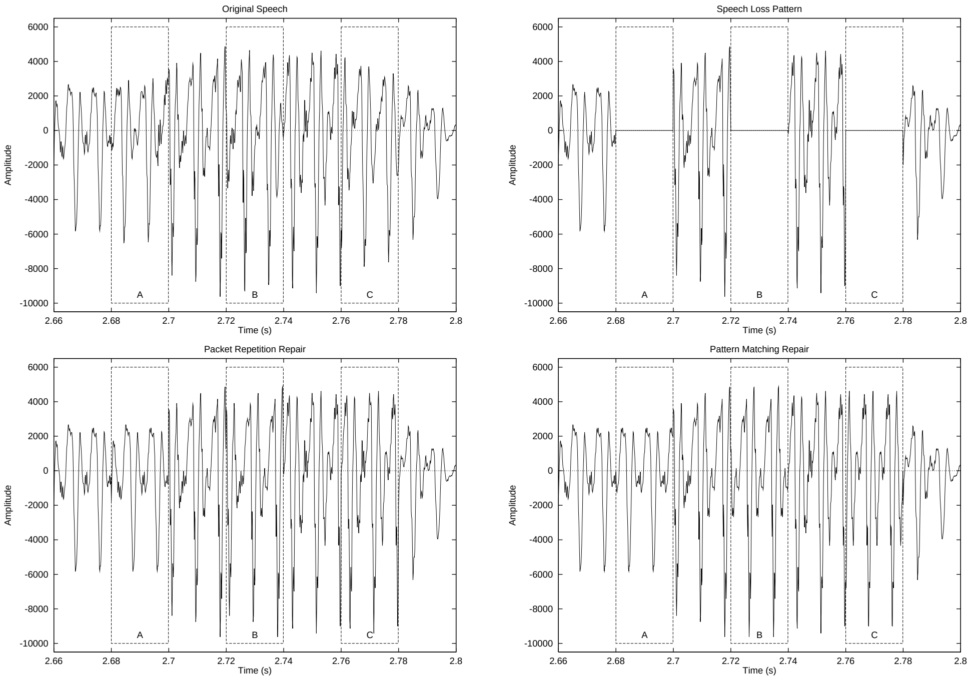

Figure 9: Sample error concealment techniques. Original audio signal*(top left)*, the loss pattern*(top right)*, packet repetition*(bottom left)* and one sided waveform substitution*(bottom right)*. Packet repetition has minor discontinuities at the loss boundaries. In this instance, the waveform substitution is continuous at boundaries on both sides because the pitch is close to constant in the interval. As both of these techniques are one-sided they have inappropriate amplitudes for packet loss C.

The delay imposed by the use of interleaving, retransmission and media independent FEC is not acceptable for these applications. Whilst media-independent FEC schemes do exist that satisfy the delay requirements, these typically have high bandwidth overhead and are likely to be inappropriate for this reason.

Our recommendation for interactive conferencing applications is that media-specific FEC is employed, since this has low-latency and tunable bandwidth overhead. Repair is approximate, due to the use of low-rate secondary encodings, but this is acceptable for this class of application when used in conjunction with receiver based error concealment.

5.3 Error concealment

Receivers must be prepared to accept some loss in an audio stream. The overhead involved in ensuring that all packets are received correctly, both in time and bandwidth, is such that some loss is unavoidable. Once this is accepted, the need for error concealment becomes apparent. Many current conferencing applications use silence substitution to fill the gaps left by packet loss, yet it has been shown that this does not provide acceptable quality. A significant improvement is achieved by the use of packet repetition, which also has the advantages of being simple to implement and having low computational overhead. The other error concealment schemes discussed provide incremental improvements, with significantly greater complexity. Accordingly, we recommend the use packet repetition since it is a simple and effective means of recovering from the low-level, random, packet loss inherent in the Mbone.

6 Acknowledgements

The authors are supported by the UK EPSRC project RAT (GR/K72780), the EU Telematics for research project MERCI (#1007) and British Telecommunications plc (ML72254).

This work has benefited from discussion with members of the networked multimedia research group at UCL. In particular, we wish to thank Jon Crowcroft and Roy Bennett for their helpful comments. We are grateful to Mark Handley for permission to use figure 1.

References

[1] E.R. Berlekamp. *Algebraic Coding Theory*. McGraw-Hill, 1968.

[2] J.-C. Bolot. End-to-end packet delay and loss behavior in the internet. In *Proceedings of ACM SIGCOMM'93*, pages 289–298, San Francisco, September 1993.

[3] J.-C. Bolot and A. Vega-García. The case for FEC based error control for packet audio in the Internet. To appear in ACM Multimedia Systems.

[4] J.-C. Bolot and A. Vega-García. Control mechanisms for packet audio in the Internet. In *Proceedings IEEE INFOCOM'96*, 1996.

[5] P. T. Brady. Effects of transmission delay on conversational behavior on echo-free telephone circuits. *Bell System Technical Journal*, 50:115–134, January 1971.

[6] G. Carle and E. W. Biersack. Survey of error recovery techniques for IP-based audio-visual multicast applications. *IEEE Network*, 11(6):24–36, November/December 1997.

[7] Y. L. Chen and B. S. Chen. Model-based multirate representation of speech signals and its application to recovery of missing speech packets. *IEEE Transactions on Speech and Audio Processing*, 15(3):220–231, May 1997.

[8] S. Deering. *Multicast Routing in a Datagram Internetwork*. PhD thesis, Stanford University, Palo Alto, California, December 1991.

[9] N. Erdöl, C. Castelluccia, and A. Zilouchian. Recovery of missing speech packets using the short-time energy and zero-crossing measurements. *Transactions on Speech and Audio Processing*, 1(3):295–303, July 1993.

[10] ETSI. Recommendation GSM 6.11 - Substitution and muting of lost frames for full rate speech channels, 1992.

[11] S. Floyd, V. Jacobson, S. McCanne, C.-G. Liu, and L. Zhang. A reliable multicast framework for light-weight sessions and applications level framing. *IEEE/ACM Transactions on Networking*, December 1997.

[12] D. J. Goodman, G. B. Lockhart, O. J. Wasem, and W.-C. Wong. Waveform substitution techniques for recovering missing speech segments in packet voice communications. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-34(6):1440–1448, December 1986.

[13] J. G. Gruber and L. Strawczynski. Subjective effects of variable delay and clipping in dynamically managed voice systems. *IEEE Transactions on Communications*, COM-33(8):801–808, August 1985.

[14] M. Handley. An examination of Mbone performance. USC/ISI Research Report: ISI/RR-97-450, April 1997.

[15] V. Hardman, M. A. Sasse, M. Handley, and A. Watson. Reliable audio for use over the Internet. In *Proceedings of INET'95*, 1995.

[16] O. Hermanns and M. Schuba. Performance investigations of the IP multicast architecture. *Computer Networks and ISDN Systems*, 28:429–439, 1996.

[17] V. Jacobson. Multimedia conferencing on the Internet. In SIGCOMM Symposium on Communications Architectures and Protocols, August 1994. Tutorial slides.

[18] N. S. Jayant and S. W. Christenssen. Effects of packet losses in waveform coded speech and improvements due to an odd-even sample-interpolation procedure. *IEEE Transactions on Communications*, COM-29(2):101–109, February 1981.

[19] I. Kouvelas, O. Hodson, V. Hardman, and J. Crowcroft. Redundancy control in real-time Internet audio conferencing. In *Proceedings of AVSPN'97*, Aberdeen, Scotland, September 1997.

[20] B. N. Levine and J. J. Garcia-Luna-Aceves. A comparison of reliable multicast protocols. *ACM Multimedia Systems*, August 1998.

[21] J.L. Massey. Shift-register synthesis and BCH decoding. *IEEE Transactions on Information Theory*, IT-15:122–127, 1969.

[22] H.F. Mattson and G. Solomon. A new treatment of bose-chaudhuri codes. *Journal of the Society of Industrial and Applied Mathematics (SIAM)*, 9(4):654–669, December 1961.

[23] S. McCanne, V. Jacobson, and M. Vetterli. Receiver-driven layered multicast. In *Proceedings ACM SIGCOMM'96*, Stanford, CA., August 1996.

[24] G. A. Miller and J. C. R. Licklider. The intelligibility of interrupted speech. *Journal of the Acoustical Society of America*, 22(2):167–173, 1950.

[25] S. B. Moon, Jim Kurose, and Don Towsley. Packet audio playout delay adjustment algorithms: performance bounds and algorithms. Research report, Department of Computer Science, University of Massachusetts at Amherst, Amherst, Massachusetts, August 1995.

[26] M. Mouly and M.-B. Pautet. *The GSM system for mobile communications.* Europe Media Duplication, Lassay-les-Chateaux, France, 1993.

[27] J. Nonnenmacher, E. Biersack, and D. Towsley. Parity-based loss recovery for reliable multicast transmission. In *Proceedings ACM SIGCOMM'97*, Cannes, France, September 1997.

[28] K. Obraczka. Multicast transport mechanism: A survey and taxonomy. *To appear in IEEE Communications Magazine*, 1998.

[29] C. S. Perkins and O. Hodson. Options for repair of streaming media. IETF Audio/Video Transport Working Group, June 1998. RFC2354.

[30] C. S. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J.-C. Bolot, A. Vega-García, and S. Fosse-Parisis. RTP Payload for redundant audio data. IETF Audio/Video Transport Working Group, 1997. RFC2198.

[31] M. Podolsky, C. Romer, and S. McCanne. Simulation of FEC-based error control for packet audio on the Internet. In *Proceedings IEEE INFOCOM'98*, San Francisco, April 1998.

[32] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne. Adaptive playout mechanisms for packetized audio applications in wide-area networks. In *Proceedings IEEE INFOCOM*, Toronto, Canada, June 1994.

[33] J.L. Ramsey. Realization of optimum interleavers. *IEEE Transactions on Information Theory*, IT-16:338–345, May 1970.

[34] I.S. Reed and G. Solomon. Polynomial codes over certain finite fields. *Journal of the Society of Industrial and Applied Mathematics (SIAM)*, 8(2):300–304, June 1960.

[35] L. Rizzo and V. Vicisano. A reliable multicast data distribution protocol based on software fec techniques. In *Proceedings of the Fourth IEEE Workshop on the Architecture and Implementation of High Performance Communication Systems (HPCS'97)*, 1997.

[36] J. Rosenberg. Reliability enhancements to NeVoT, December 1996.

[37] J. Rosenberg and H. Schulzrinne. An RTP payload format for generic forward error correction. IETF Audio/Video Transport working group, July 1998. Work in progress (Internet-draft).

[38] H. Sanneck, A. Stenger, K. Bem Younes, and B. Girod. A new technique for audio packet loss concealment. In *IEEE Global Internet 1996*, pages 48–52. IEEE, December 1996.

[39] H. Schulzrinne. RTP profile for audio and video conferences with minimal control. IETF Audio/Video Transport Working Group, March 1997. Work in progress.

[40] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. IETF Audio/Video Transport Working Group, January 1996. RFC1889.

[41] International Telecommunications Union. Recommendation G.723.1: Dual rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s, March 1996.

[42] L. Vicisano, L. Rizzo, and Crowcroft J. TCP-like congestion control for layered multicast data transfer. In *Proceedings IEEE INFOCOM'98*, 1998.

[43] V. R. Viswanathan, J. Makhoul, R. M. Schwartz, and A. W. F. Higgins. Variable frame rate transmission: A review of methodology and application to narrow-band LPC speech coding. *IEEE Transactions on Communications*, COM-30(4):674–687, April 1982.

[44] R. M. Warren. *Auditory Perception.* Pergamon Press, Inc., 1982.

[45] O. J. Wasem, D. J. Goodman, C. A. Dvorak, and H. G. Page. The effect of waveform substitution on the quality of PCM packet communications. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 36(3):342–348, March 1988.

[46] R. X. Xu, A. C. Myers, H. Zhang, and R. Yavatkar. Resilient multicast support for continuous media applications. In *Proceedings of the 7th International Workshop on Network and Operating Systems Support for Digital Audio and Video (NOSSDAV'97)*, Washington University in St. Louis, Missouri, May 1997.

[47] M. Yajnik, J. Kurose, and D. Towsley. Packet loss correlation in the Mbone multicast network. In *Proceedings IEEE Global Internet Conference*, November 1996.