

Colin Perkins
University College London

Options for Repair of Streaming Media
draft-ietf-avt-info-repair-00

Status of this memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as ``work in progress''. To learn the current status of any Internet-Draft, please check the ``lids-abstracts.txt'' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), ftp.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited.

Comments are solicited and should be addressed to the author(s) and/or the IETF Audio/Video Transport working group's mailing list at rem-conf@es.net.

Abstract

This document summarizes a range of possible techniques for the repair of continuous media streams subject to packet loss. The techniques discussed include redundant transmission, retransmission, interleaving and forward error correction. The range of applicability of these techniques is noted, together with the protocol requirements and dependencies.

1 Introduction

A number of applications have emerged which use IP multicast to deliver continuous media streams. Due to the unreliable nature of IP multicast transport, the quality of the received stream will be adversely affected by packet loss. A number of techniques exist by which the effects of packet loss may be repaired. These techniques have a wide range of applicability and require varying degrees of protocol support.

In this document, four such techniques (redundancy, interleaving, retransmission, and FEC) are discussed, and recommendations for their applicability made.

The author's experience is with the development of a loss-resilient multicast audio conferencing application. This document has, therefore, been prepared with the underlying assumption that the media is streaming audio. The techniques discussed are, however, expected to generalize to other media types in many cases.

2 Terminology and Protocol Framework

A unit is defined to be a timed interval of media data, typically derived from the workings of the media coder. A packet comprises one or more units, encapsulated for transmission over the network. For example, many audio coders operate on 20ms units, which are typically combined to produce 40ms or 80ms packets for transmission.

The framework of RTP [10] is assumed. This implies that packets have a sequence number and timestamp. The sequence number denotes the order in which packets are transmitted, and is used to detect losses. The timestamp is used to determine the playout order of units. Most loss recovery schemes rely on units being sent out of order, so an application must use the RTP timestamp to schedule playout. The use of RTP allows for several different media coders, with a payload type field being used to distinguish between these at the receiver. Some loss recovery schemes send some units multiple times, using different encoding schemes. A receiver is assumed to have a 'quality' ranking of the differing encodings, and so is capable of choosing the 'best' unit for playout, given multiple options.

3 Network Loss Characteristics

If it is desired to repair a media stream subject to packet loss, it is useful to have some knowledge of the loss characteristics which are likely to be encountered. A number of studies have been conducted on the loss characteristics of the Mbone [8,9] and although the results vary somewhat, the broad conclusion is clear: in a large conference it is inevitable that some receivers will experience packet loss. Packet traces taken by Handley [8] show a session in which most receivers experience loss in the range 2-5%, with a somewhat smaller number seeing significantly higher loss rates. Other studies have presented broadly similar results.

It has also been shown that the vast majority of losses are of single packets. Burst losses of two or more packets are around an order of magnitude less frequent than single packet loss, although they

do occur more often than would be expected from a purely random process. Longer burst losses (of the order of tens of packets) occur infrequently.

These results are consistent with a network where small amounts of transient congestion cause the majority of packet loss. In a few cases, a network link is found to be severely overloaded, and large amount of loss results.

The primary focus of a packet loss repair scheme must, therefore, be to correct single packet loss, since this is by far the most frequent occurrence. It is desirable that losses of a relatively small number of consecutive packets may also be repaired, since such losses represent a small but noticeable fraction of observed losses. The correction of large bursts of loss is of considerably less importance.

4 Loss Repair Schemes

In the following sections, four loss repair schemes are discussed. These schemes have been discussed in the literature a number of times, and found to be of use in a number of scenarios. Each technique is briefly described, and its advantages and disadvantages noted. A summary and comparison follows.

4.1 Redundant Transmission

The case for redundant transmission of audio data has been made in [5,6]. Each unit is coded multiple times, and sent in several packets. If a packet is lost, a subsequent packet contains a copy of the unit which may be used as a replacement. By recoding the redundant unit(s) with a low bit-rate compression scheme the overhead of this technique may be reduced, at the expense of a reduction in quality (but note that even an LPC encoded fill-in sounds better than silence). Unlike the other techniques discussed, the use of redundancy 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. In a broadcast-style environment, it is possible to delay the redundant copy of a packet, achieving improved performance in the presence of burst losses [7], at the expense of additional latency.

If the redundant copies of a unit are recoded with a low-bandwidth compression scheme, the bandwidth overhead of this technique is small. This does, however, result in an increased processor load which may make this technique infeasible on low power workstations, particularly if other media types are also being coded.

An RTP payload format for redundant data is defined in [1]. This has been implemented in a number of audio tools, and has been shown

to perform well.

4.2 Retransmission

Retransmission of lost packets is an obvious means by which loss may be repaired. It is clearly of value in broadcast style applications, with relaxed delay bounds, but many authors have discounted the use of retransmission for interactive applications, due to the potentially large delay imposed. A recent paper [4] challenges this: in that paper it is noted that ``the desired degree of interactivity typically varies from one participant to another'', and that this leads to an interesting tradeoff between quality (reliability in delivery, due to retransmission of lost packets) and interactivity (latency in delivery).

In addition to the possibly high latency, there is a potentially large bandwidth overhead to the use of retransmission. Not only are units of data sent multiple times, but additional control traffic must flow to request the retransmission. It has been shown [8] that, in a large Mbone session, most packets are lost by at least one receiver. In this case the overhead of requesting retransmission for most packets may be such that redundant transmission is more acceptable. This leads to a natural synergy between the two mechanisms, with a redundant transmission being used to repair all single packet losses, and those receivers experiencing burst losses, and willing to accept the additional latency, using retransmission based repair as an additional recovery mechanism.

In order to reduce the overhead of retransmission, the retransmitted units may be piggy-backed onto the ongoing transmission. This also allows for the retransmission to be recoded in a different format, to further reduce the bandwidth overhead.

Note that the choice of a retransmission request algorithm which is both timely and network friendly is an area worthy of further study.

4.3 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. 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 could contain units 1, 5, 9, 13; the second packet would contain units 2, 6, 10, 14; and so

on. 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 results in a noticeable increase in the perceived quality of an audio stream, for example.

The obvious disadvantage of interleaving is that it increases latency. This limits the use of this technique for interactive applications, although it performs well for broadcast use. The major advantage of interleaving is that it does not increase the bandwidth requirements of a stream.

A potential RTP payload format for interleaved data is a simple extension of the redundant audio payload [1]. That payload requires that the redundant copy of a unit is sent after the primary. If this restriction is removed, it is possible to transmit arbitrary interleaving-s of units with this payload format.

4.4 Forward Error Correction

Forward error correction (FEC) schemes rely on the addition of repair data to a media stream, from which lost packets may be recovered. That repair data takes the form of 'parity' packets, calculated from the exclusive-or (XOR) of a number of data packets. A lost packet may be regenerated by XOR'ing the received data with the repair data. A number of FEC schemes have been proposed for use with continuous media streams by Budge et al [3]. These vary the bandwidth, latency and repair capabilities by XOR'ing different combinations of packets to generate the parity packets.

FEC based techniques have a significant advantage in that they are media independent, and provide exact repair for lost packets. In addition, the processing requirements are relatively light, especially when compared with some redundancy schemes which use very low bandwidth redundant encodings.

Disadvantages of FEC include high latency (in some cases), and potentially high bandwidth overhead. It is possible to reduce the bandwidth used by the FEC data, but this can only be achieved at the expense of reduced repair capability. If the bandwidth is available, FEC does, however, provide very good error recovery capabilities. Two RTP payload formats have been proposed for FEC protected data: the original by Budge et al [3], and an alternative from Rosenberg and Schulzrinne [2] who generalize the protocol somewhat.

	Latency	Bandwidth Overhead	Processing Overhead
Redundancy	Small	Variable	Variable, may be large
Retransmission	Medium	Variable	High
Interleaving	High	None	Low
FEC	High	High	Low

Table 1: Overheads of different repair schemes

4.5 Summary

A comparison of the relative overheads of the four schemes discussed is provided in table 1. It can be seen that the latency overhead is such that the use of redundant transmission is preferable for interactive use, whereas interleaved streams or FEC are preferable for broadcast style applications. The use of retransmission together with redundant transmission offers an interesting trade-off between the two approaches, with participants requiring interactivity relying on the redundant data only, and other participants using retransmission to correct losses at the expense of additional delay.

In terms of error recovery capability, the clear winner must be the use of retransmission, since this will eventually recovery all lost packets (the time required to achieve this may be large, however). Of the other schemes, the use of FEC as proposed by Budge et al [3], is typically the most effective repair mechanism. The use of multiple redundant encodings can achieve similar repair capability, although the processing requirements are likely to be excessive if differing encodings are used for the multiple redundant units.

5 Open Issues

Of the four techniques discussed, only redundant transmission has a well defined, standard, protocol framework (although this may clearly be reused for the retransmission of media data). A simple extension to this protocol provides a possibility for transporting interleaved media streams.

The choice of a retransmission algorithm which is both timely and network friendly, together with a suitable control protocol, is an area worthy of further study.

Two conflicting proposals exist for the transport of FEC protected data. This must clearly be resolved.

Experience with redundant audio (using a single, low bandwidth, redundant encoding) has shown that this is sufficient to protect against 30% packet loss in many cases. It is possible to protect against much higher packet loss rates, but this may not be desirable. Many current

media streaming applications do not employ congestion control, and the widespread use of techniques which allow operation of these tools in the presence of high levels of congestive packet loss is dubious, at best. It would clearly be useful if guidelines on this issue could be derived before widespread deployment occurs.

6 Acknowledgments

The author wishes to thank Orion Hodson for his helpful comments on an early version of this document.

7 Author's Address

Colin Perkins
Department of Computer Science
University College London
Gower Street
London WC1E 6BT
United Kingdom

Email: c.perkins@cs.ucl.ac.uk

8 References

- [1] C. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J.-C. Bolot, A. Vega-Garcia and S. Fosse-Parisis, ``RTP Payload for Redundant Audio Data'', Internet draft, IETF Audio/Video Transport working group, July 1997, draft-ietf-avt-rtp-redundancy-01.txt.
- [2] J. Rosenberg and H. Schulzrinne, ``An A/V Profile Extension for Generic Forward Error Correction in RTP'', Internet draft, IETF Audio/Video Transport working group, July 1997, draft-ietf-avt-fec-00.txt
- [3] D. Budge, R. McKenzie, W. Mills and P. Long, ``Media-Independent Error Correction using RTP'', May 1997, draft-budge-media-error-correction-00.txt
- [4] X. Rex Xu, A. C. Myers, H. Zhang, R. Yavatker, ``Resilient Multicast Support for Continuous-Media Applications'', Proceedings NOSSDAV'97.
- [5] J.-C. Bolot and A. Vega-Garcia; The case for FEC-based error control for packet audio in the Internet; ACM Multimedia Systems, 1997
- [6] V. J. Hardman, M. A. Sasse, M. Handley and A. Watson, ``Reliable Audio for Use over the Internet'', Proceedings INET'95, Honolulu, Oahu, Hawaii, September 1995. <http://www.isoc.org/in95prc/>

- [7] I. Kouvelas, O. Hodson, V. Hardman and J. Crowcroft, ``Redundancy Control in Real-Time Internet Audio Conferencing'', Proceedings of AVSPN'97, September 1997.
- [8] M. Handley, ``An Examination of Mbone performance'', USC/ISI Research Report: ISI/RR-97-450, <http://buttle.lcs.mit.edu/mjh/mbone.ps>
- [9] M. Yajnik, J. Kurose and D. Towsley, ``Packet loss correlation in the Mbone multicast network'', Proceedings of IEEE Globecom'96, November 1996.
- [10] 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, RFC 1889.