An Evaluation of Delay-Aware Receiver Playout Strategies for VoIP Applications

Hugh Melvin and Liam Murphy*

Dept. of Computer Science, UCD, Ireland

Abstract. Previous work by the authors confirm the feasibility of implementing a delay-aware receiver buffering strategy based on synchronised time within VoIP applications. Our objective in this paper is to extensively test this delay-aware approach, termed the hybrid playout strategy over diverse networks using a combination of simulation and delay emulation. We utilise both measured delay data and derived delay models and quantify the significant performance gains through use of the ITU-T E-model. We also analyse recent Internet delay studies to further assess the hybrid's wider applicability.

Keywords: Periodic Network Measurements, Simulations, Network Emulation, Synchronised Time.

1 Introduction

A recent review of Internet telephony in [1] indicates that despite significant research, its deployment has been very limited mainly due to quality of service (QoS) concerns. In this paper we examine receiver-based buffering approaches to improving VoIP QoS.Adaptive buffering approaches generally have no knowledge of actual delays and adapt to trends in delay [2] [3] [4]. Non-adaptive approaches implement a simpler fixed delay at the expense of inflexibility.

In previous work [5] [6] we propose a hybrid playout strategy based on synchronised time, provided via the Network Time Protocol (NTP) and the Real-time Transport Protocol Control Protocol (RTCP) Sender Reports (SR). This delay-aware approach enables end-to-end delays to be determined on a per-packet basis and used to select an optimum playout strategy i.e. adaptive or fixed. The extent of testing in [6] however was very limited and in effect proved the technical feasibility of the approach rather than its applicability. In this paper we evaluate a revised version of the hybrid over a wide range of network conditions using both measured delays and derived delay models as advocated by [7]. As a basis of comparison, we use two of the baseline adaptive strategies, detailed in [2].

To facilitate trace-driven testing, extensive delay data was captured in tests within Ireland and to the UK, mainland Europe and the US. The relative performance of the various playout strategies was evaluated against this test data

* H. Melvin is contactable at hugh.melvin@nuigalway.ie

N. Mitrou et al. (Eds.): NETWORKING 2004, LNCS 3042, pp. 320–331, 2004.

using two different approaches. The first imposed the trace delay data on a simulated voice stream, and evaluated through simulation, the various playout strategies. The second approach utilized pre-recorded voice streams and a LAN-based testbed incorporating a delay emulator. Although the former approach facilitated rapid testing, the latter is useful in evaluating the various strategies in a real implementation environment and lends itself to subjective testing though various implementation issues arose which degraded testbed effectiveness.

In addition to the trace data, more general delay models were also developed.These models were tuned to present more severe network conditions than those experienced from trace data and relative performance was assessed via the simulated approach. A simplified yet conservative adaption of the ITU-T E-model was used in all cases to quantify performance gains.

Results show that the hybrid playout strategy performed better than the baseline adaptive approaches in over 99% of tests. The extent of gain in E-model R-factor varied up to 50% but averaged at about 10-15% relative to the two conventional adaptive approaches.

In general, we propose that although jitter remains a serious issue in *best-effort* Internet, actual delays are often well within the 150 msec M2E delay limit. In this context, a delay-aware playout strategy such as the hybrid algorithm can deliver significant voice quality improvements across Internet Wide Area Network (WAN) paths. We review recent delay studies by [9] [10]that support this position. Adaptive approaches, though useful where precise delay information is not available, will often result in unnecessary late packet loss and inherent voice distortion.

The remainder of the paper is organized as follows. Section 2 reviews the rationale for the hybrid algorithm and describes the additional dynamic features introduced in this paper. Section 3 deals with various test design issues including delay measurement, network emulation, general Internet delay models and also outlines the use and limitations of the ITU-T E-model.Section 4 outlines implementation details, section 5 summarizes results and finally section 6 concludes by describing current and future work plans.

2 Receiver Buffer Strategies

In order to cope with the Internet's *best-effort* service level, a great deal of research has focused on the optimal implementation of application delay or buffering delay within receivers. We evaluated many such adaptive approaches such as [2] [3] [4] and [9] in [6] and identified Alg. 1 and 4 from [2] (labelled Alg. A and B respectively in [6]) as common candidates for baseline comparison.

The following points summarize the key issues relating to the design of receiver buffer strategies for best-effort networks:

- Adaptive algorithms such as A & B that utilize TCP-like formula tend to react too slowly yet overestimate playout delays where delay variation is significant.

- The selection of tuning parameters is a non-trivial matter though is required to suit network characteristics. Such characteristics may however change with time.
- Adaptive approaches result in some degree of late loss.They also result in distortion of the original audio signal.
- Adaptive algorithms are most useful when receivers have no knowledge of actual one-way delays. If actual delays are known, and are within G.114 [8] requirements, a fixed playout delay will avoid many of the problems associated with adaptive algorithms. However a fixed playout delay imposed at the start and fixed for the duration of a session may quickly become inappropriate.

2.1 Hybrid Adaptive-Fixed Algorithm

In [5] and [6], we propose, describe and test to a limited degree, the hybrid approach. It combines the useful characteristics of both adaptive and fixed buffer strategies and requires end-host system clocks to be synchronized which is achieved through use of NTP and RTCP SR packets.

The following briefly describes the hybrid strategy, refer to [6] for more detail:

Session commences implementing an adaptive buffer algorithm. Synchronized time enables one-way packet delays to be precisely determined once the first RTCP SR packet is received. Each receiver generates and maintains a rolling delay histogram. In the following equation, *est* is the delay estimate value extracted from the histogram to meet target loss requirements, Wf is a weight factor applied in determining the fixed playout delay and *Pkt* refers to the packetisation delay. If *est* is within the G.114 limit, playout switches to fixed mode and the situation thereafter is monitored:

$$playout = est + (150 - Pkt - est) * Wf$$

Note that for the limited results reported in [6], both Wf and the frequency of re-evaluation of playout mode were fixed for the duration of a session. In this paper we introduce self tuning features to the hybrid.

Tuned Hybrid Algorithm. In [6], the authors test the hybrid algorithm over the limited Irish academic and research network and report significant performance gains. In this paper, we tune the hybrid as follows:

- *Dynamic Wf:* The weight factor Wf provides an extra safety margin in determining the fixed playout delay. Wf is initialized to 0.33 as before but subsequent values are set at $variation/median$ with floor and ceiling values applied to maintain Wf in the range $0.1 - 1$. Essentially, this means that the safety margin applied is proportional to network jitter.
  As outlined by [9] and discussed in section 3.4, users are more sensitive to increased loss than to increased delay. This forms the basis for the development of the hybrid approach in that a higher (though fixed) delay is imposed, resulting in lower late losses and overall better quality.

– *Dynamic Estimation Interval:* The frequency at which playout mode is re-evaluated is set dynamically according to network conditions. A baseline value of 500 packets is used which equates to 16 sec with 32 msec packets but subsequent intervals are a function of $(150/median) * 500$. This ensures that where delays are running close to the G.114 limit, the operating mode is monitored more closely.

The core benefit of the hybrid algorithm is that an informed fixed playout is implemented whenever possible so that the integrity of speech both within and between talkspurts is maintained. Adaptive mode is thus used only when absolutely necessary.

3 Test Design Issues

As outlined in section 1, we utilised both measured delays and derived delay models for testing and regarding the former, used both a simulation-based and emulation-based approach. Model-driven testing was done through simulation only. Fig. 1 outlines the simulation model used and Fig. 2 illustrates the LAN-based emulator developed to evaluate the hybrid algorithm in a real implementation environment. The following sections briefly introduce and describe the network emulator, delay measurement issues, network delay and loss modelling and finally the ITU-T E-model.

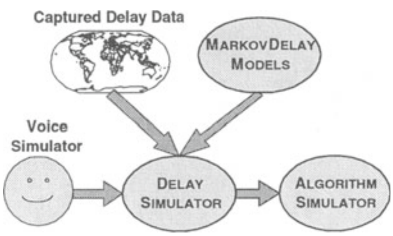

Fig. 1. Simulator Model

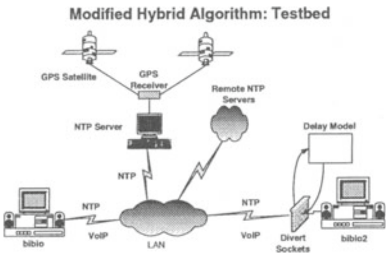

Fig. 2. LAN-based Emulator

3.1 Network Emulator

The network delay emulator is based on Linux Divert Sockets. A detailed description of its operation and use in testing applications over a variety of network models is given in [13]. Essentially it enables a user to intercept media packets at the socket level and divert them to a user process which reinjects them according to user specified rules of loss or/and delay.

3.2 Delay Measurement Issues

We gathered extensive delay data using the *ping* utility and estimated one-way delays assuming symmetric paths. Although one-way delay measurement is superior to round-trip-time (RTT) delays, RTT data is much easier to capture. Our objectives were to capture the general characteristics of diverse network paths and thus RTT data was considered sufficient. Bolot in his analysis [16] uses round-trip-times and Schulzrinne [14] though recording one-way delays suggests that round-trip-delays are generally sufficient.

Where precise one-way delay measurements are required, relying on RTT is unwise as outlined in RFC 2679 [11]. RFC 2330 [12] outlines that for delay measurements, periodic test packet generation can lead to synchronization issues or interference with network characteristics. For our tests, however the objective was somewhat different in that we wish to see the effect of simulated VoIP data streams on overall delay characteristics. In this regard, our objectives are similar to those of [16], [17], [14] detailed above and thus the more recent RFC 3432, on periodic sampling provided a more useful reference [15].

3.3 Internet Modelling

Internet loss and to a lesser degree delay modelling has attracted much research in recent years. In [7], the pitfalls of relying solely on trace-driven simulation are highlighted and a source-driven approach is also advised. As outlined in [14] and [16], network delay and loss both exhibit burstiness or temporal dependency and losses are often preceded by high delays which in a VoIP environment can lead to a final loss pattern of large loss bursts, comprising late loss and network loss. Our focus here is solely on delay modelling.

Schulzrinne et al in [17] and [14] examine the issue of delay burstiness and recommend the conditional CDF, defined in equation 1 as a useful metric to capture burstiness.

$$f(t) = P(d_i >= t | d_{i-l} >= t), l = 1, 2, 3, ... \tag{1}$$

[17] also examines the degree to which different queuing models reproduce the conditional CDF phenomenon. Bolot et al [16] model delay using a single server, 2 input stream queue where one stream represents periodic media packets and the other represents all other Internet traffic. They report good correlation between analytical and experimental results. In [13],a Markov Modulated Poisson Process (MMPP) model is used to simulate zero-loss, high delay-variance in a wireless channel with retransmissions.

In analysing trace data and developing general delay models, we draw from a number of the above sources. We applied the conditional CDF method to the trace data to determine the extent of delay burstiness. We then developed general bursty delay models, by applying some characteristics of actual traces to a 2-state Markov model. Unlike [13] which used a Poisson process to model interarrival times (in a wireless environment), packet interarrival times within each state were centered on the mean, with a wider spread of values for the BAD state than for the GOOD state.

3.4 ITU E-model

Although the use of Mean Opinion Score (MOS) tests are the preferred mechanism for evaluating speech quality, it requires significant resources to yield useful results. The recently developed ITU-T E-model is a rapid planning tool for predicting how the average user rates the voice quality of a phone call with known transmission parameters.The model returns a transmission rating factor R (scale 0-100), defined as:

$$R = (R_o - I_s) - I_d - I_e + A$$

Technical details are specified in [19] including a mapping from R to MOS scales. The factors of interest here are I_d (delay impairment) and I_e (loss impairment). I_e includes the distortion caused by low bit rate codec operation as well as the effect of packet loss (both network and late loss). Due to its ease of deployment, many other researchers such as [20] [21] and [18] have adopted it for assessing speech quality.

In the context of evaluating real voice sessions, the E-model is however limited in that it delivers an instantaneous rating based on singular loss and delay figures. Other research has thus examined issues such as bursty versus random loss, recency [22] [9], perceived versus instantaneous quality [22] and rating of entire voice calls rather than segments [9]. For the purposes of this paper, we apply the same simplified yet conservative E-model analysis developed in our previous work [6]. As such, our analysis is aimed at extracting approximate relative values rather than detailed absolute values.

As such, the I_d impairment can be approximated by 10 units per 100 msec [9] ([21] uses 3 units/100 msec up to 170 msec). For the I_e factor, we base impairment on that for G.711, with PLC implemented, a packet size of 10 msec and bursty rather than random loss. Packet loss in the range 0-5% can therefore be approximated by a dual-slope curve set at 10 units per % above 3% and at 3 units per % below 3% [see G.113 [23] for details]. A critical factor for the hybrid algorithm is the lower sensitivity of I_d to increased delay than that of I_e to packet loss. As outlined in section 2.1, this justifies the use of the weight factor Wf in calculating the fixed playout delay from the delay estimate *est*.

4 Implementation Details

Implementation details for capturing trace data, for the simulator and for the LAN-based emulator are presented separately.

4.1 Trace Capture

Trace delay data was gathered from tests to a number of remote Internet machines, in Ireland, UK, mainland Europe and the US. The *ping* utility was configured to mimic actual data streams (eg. 256 byte packets every 32 msec mimics G.711 data packets) and round-trip-times were recorded. In all, over 1000 tests were conducted. Tests lasted between 20 seconds and three minutes and were repeated at various times in order to capture any diurnal variation.

4.2 Simulator

For simulating voice input, the distribution of talkspurts and silence periods was generated from actual voice streams. This was mapped to the delay values to simulate real voice traffic which was fed to the various playout algorithms.In [9], a scale of 1 to 6 is used to distinguish between conversations involving short, rapid exchanges and slower, more relaxed speech. We chose a *medium* scale voice session for tests. Both Alg. A and B had default settings from [2] (Alg. A:α=0.998002 and β=4, Alg. B:α=0.875, β=4). Spike threshold values were chosen based on network conditions and varied from 10-30 msec. This simulation-based approach facilitated extensive and rapid testing and is similar to that used by [2].

4.3 Emulator

- Both end hosts were situated on the same LAN with the divert sockets application running on one host. See Fig. 2.
- Trace delay data files were used as source data for the emulator.
- Pre-recorded voice streams were delivered across the testbed. As with the simulator, a *medium* scale ON/OFF speech pattern was used.

5 Results

We firstly analyze the subset of trace delay data used for algorithm evaluation. The relative performance of the various algorithms when tested with the simulator approach using both the trace-driven delays and general delay models is then assessed. The emulator was then used to confirm the relative performance using a subset of the trace-driven delay tests in a real VoIP environment. Finally, we summarize recent delay studies such as [9] [10]that support the wider applicability of the hybrid approach.

5.1 Measured Delay Characteristics

We present a representative sample of the total trace delay data captured. More extensive results are available from [24]. This sample focuses on tests to four remote servers located in Dublin (www.ucd.ie), London (www.cs.ucl.ac.uk), Munich (www.lkn.ei.tum) and Berkeley, Calif.(www.icir.org). Tests to each were carried out in blocks of 30 where test numbers correspond to the following times of day (GMT/UTC): Test 1-6 (17:00-18:00),Test 7-12 (22:00-23:00), Test 13-18 (09:00-10:00), Test 19-24 (10:00-11:00) and Test 25-30 (13:00-14:00).

The following points summarise trace delay results:

- Mean Delay & Diurnal Variation: Although the *mean* delay to each location varied little over the course of a day, diurnal variation was evident with greatest burstiness evident in the 17:00-18:00 window.

– Unconditional/Conditional CDF: Although there was evidence to support some temporal dependence of delay values, it was most evident in the 17:00-18:00 window GMT.
– Network Loss: Network loss rates were generally very low, though interestingly were highest for the most local path to UCD. This contrasts with the more usual findings that international paths suffer from higher loss rates.

General Delay Models. A number of 2-state Markov models were developed based loosely on captured traces to both Germany (www.ei.tum.de) and the US (www.icir.org). The mean delay required as a priming delay value was taken directly from the traces. The required Markov parameters such as the average high-delay burst length (i.e. BAD state) and overall extent of high delays were exaggerated from traces to increase burstiness. A range of different models were developed and tested; results presented are based on a high delay state probability of 0.05 and a mean high-delay burst of 320msec (10 packets). As outlined in section 4.3, the GOOD and BAD states were modelled using two different ranges of packet jitter, significantly higher than those recorded in traces (Compare Fig. 3 with Fig. 8).

5.2 Hybrid Performance Results

Simulator-based results using both the actual delay traces and the 2-state Markov models are presented separately.

Trace-Driven Testing. Fig. 3 illustrates for LKN, a single test performance of the hybrid algorithm along with Algorithm A/B (Adaptive/Spike). The spike threshold for Alg.B was set to 10 msec. As evident from these figures, the hybrid resulted in higher playout delays but significantly lower late losses.

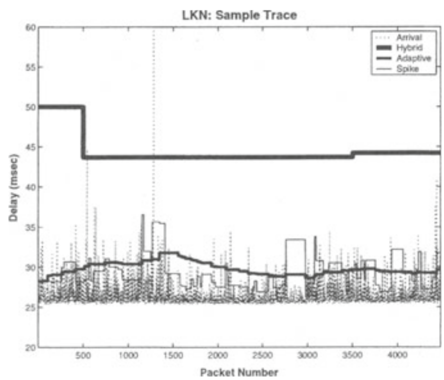

Fig. 3. Sample Trace Performance: LKN

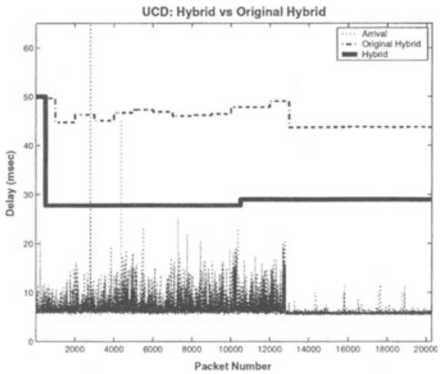

Fig. 4. Self-tuning Hybrid

Figures 5, 6, and 7 outline the performance of the hybrid along with adaptive algorithms Alg. A and B over a single but representative days tests to UCL, ICIR and LKN respectively. The plots within each figure show the average late loss rate for all three algorithms, the additional delay introduced by the hybrid relative to Alg.A/B, and the net gain in R-factor.

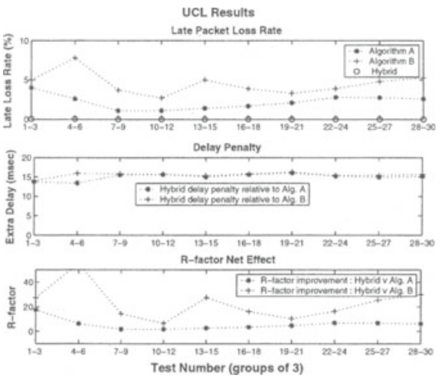

Fig. 5. Performance Comparison: UCL

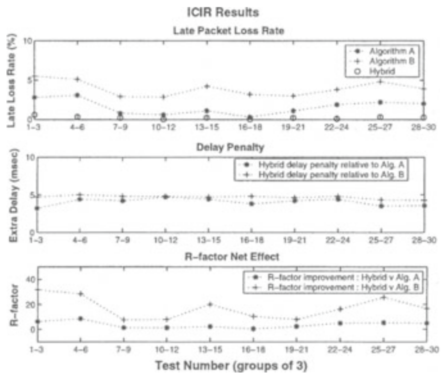

Fig. 6. Performance Comparison: ICIR

Of the 120 tests shown here, the hybrid returned a net gain in all but one. The extent of gain using our *conservative* E-model analysis varied up 50% and was highest in the test range 1-6 which corresponded to periods of highest jitter. Average gains relative to Alg. A/B were approximately 10/15%. Fig. 7 also shows that the R-factor gain is closely correlated with the standard deviation of packet delays. This reflects the fact, reported also by [9] that adaptive algorithms tend to track network conditions too closely resulting in unnecessary late loss when significant jitter occurs.

Although the primary objective of this paper was to assess the hybrid under more diverse network conditions, the performance improvements due to hybrid self-tuning features introduced in section 2.1 are evident from Fig. 4. This shows a decrease in playout delay (due to the variable W_f) with almost no increase in late loss along with a significant reduction in the frequency of playout mode re-evaluation (every 10000 versus every 500 packets).

General Delay Models. Fig. 8 shows the relative performance of the various algorithms when tested with the two-state Markov delay models developed for LKN from section 5.1. Relative to Alg. A, the hybrid resulted in a reduction of 2.8% (3.3 to 0.5) in late loss with an average increase in playout delay of 28 msec, resulting in a net R-factor gain of 8%. The corresponding gain for ICIR was higher at 35% (not shown). This was due to the much higher late losses in adaptive mode (6%) than in hybrid mode (2.4%). This reflects the fact that jitter which was set as a percentage of delays was significantly higher for ICIR,

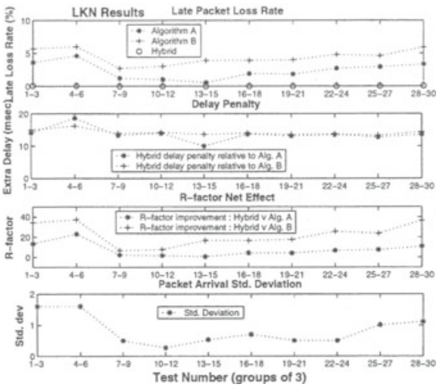

Fig. 7. Performance Comparison: LKN

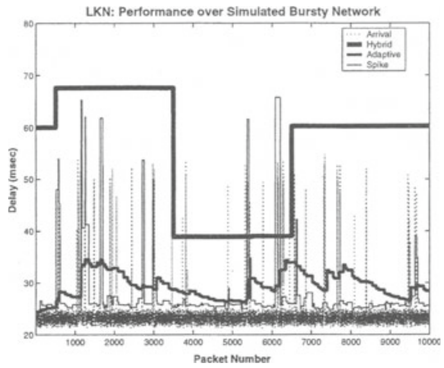

Fig. 8. LKN: Bursty Delay Model

leading to poor adaptive performance. The late loss rate was higher in hybrid mode for the ICIR than for LKN model due to the proximity to the G.114 limit. Alg. B performed marginally better than A in both cases, due to the high degree of bursty delays. More generally, the hybrid consistently outperformed both Alg. A and B, once overall delay values were within the G.114 limit. See [24] for more detail.

5.3 Testbed Performance

The emulator enabled simulator-based results to be repeated in a real implementation environment.Although results correlated well with those from the simulator-based approach, the delay emulator suffered from delay uncertainty, caused principally by timer granularity limitations.This resulted in actual imposed delays differing slightly from the measured trace delays, making precise comparisons impossible. On a related note, [25] carry out black box testing of various VoIP endpoints to measure, amongst other things, M2E delay introduced solely by the endpoints. Their results show that the combination of OS, packetisation, buffering, and miscellaneous hardware delays can often accumulate to 150 msec and more which is a serious cause for concern. As such, although a primary advantage of the emulator approach is that it facilitates subjective testing, in practice, due to the emulator delay jitter and endpoint uncertainties, our performance gains reported are based solely on E-model analysis.

5.4 Internet Delay Studies

Our results confirm that the hybrid approach can deliver significant gains over both short and long distance Internet paths. In [10] Roychoudhuri et al carry out extensive testing (using a smoothed RTT metric) of US and International paths and report that though jitter remains a serious problem, delay rarely exceeds the acceptable limits for telephony. Finally in [9], testing of forty three (43)

US backbone paths was carried out. These paths were classified into five types according to fixed and variable delay components and were continuously tested over a 2.5 day period. For 39 of these paths, 98% of packets experienced delays less than or equal to 120 msec. For the remaining four, classified as long haul with high delay burstiness, 98% of packets experienced delays less than or equal to 170 msec. The above brief analysis confirms that delay-aware playout strategies such as the hybrid offer a viable alternative to conventional delay-unaware adaptive approaches.

6 Conclusions and Future Work

In this paper we extensively test the hybrid algorithm using both a trace-driven and derived delay model approach. Conventional adaptive algorithms often result in unnecessary late packet loss and voice distortion. Where M2E delays are within G.114 requirements, an informed fixed playout delay, though increasing overall end-to-end delay, can significantly reduce late packet loss and fully preserve speech integrity. This trade-off can often result in improved quality as users are more sensitive to increased loss than delay. Extensive delay measurements carried out in this paper coupled with an analysis of third party studies suggest that delay characteristics within the Internet make the hybrid algorithm a viable alternative. Ongoing work is examining the timing limitations of the delay emulator in order to facilitate more effective evaluation, including subjective testing. Further analysis of Internet delay studies such as the Active Measurement Project [26] is being undertaken. Finally, more analysis is required of the uncertainties caused by hardware/software within VoIP endpoints. This has particular implications for the hybrid algorithm but applies more generally to any Internet multimedia application.

References

1. S. Bradner, "Internet Telephony-progress along the road," *IEEE Internet Computing*, vol. 6,no. 3,May-Jun. 2002.
2. R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, "Adaptive playout mechanisms for packetized audio applications in wide-area networks," *Proc. Conf. Comp. Comm. (IEEE Infocom)*, IEEE CS Press, Los Alamitos, Calif., Jun. 1994.
3. S. Moon, J. Kurose, and D. Towsley, "Packet audio playout delay adjustment:performance bounds and algorithms," *ACM/Springer Multimedia Systems*, vol. 6, pp. 17-28, Jan. 1998.
4. Y. Liang, N. Farber, and B. Girod, "Adaptive playout scheduling using time-scale modification in packet voice communications," *Proc. of ICASSP 2001.*
5. H. Melvin and L. Murphy, "Time synchronization for VoIP Quality of Service," *IEEE Internet Computing*, vol. 6,no. 3,May-Jun. 2002.
6. H. Melvin and L. Murphy, "An evaluation of the potential of synchronized time to improve VoIP quality", *IEEE Intl Conf. on Comm. (ICC 2003)*,Anchorage, May 2003.

7. S. Floyd, "Difficulties in Simulating the Internet." *IEEE/ACM Trans. on Networking*,vol. 9, no. 4, Aug. 2001.
8. Recommendation G.114, "One way transmission time," ITU, May 2000.
9. A. Markopoulou, F. Tobagi, and M. Karam, "Assessment of VoIP quality over Internet backbones," *IEEE Proc of Infocom 2002*, NY, Jun. 2002.
10. L.Roychoudhuri, E.Al-Shaer, H.Hamed, and G.Brewster, "Audio Transmissions over the Internet:Experiments and Observations," *IEEE Intl. Conf. on Comm. (ICC 2003)*,Anchorage, Alaska, May 2003.
11. G. Almes, S. Kalidindi, M. Zekauskas, "A One-way Delay Metric for IPPM," *IETF RFC 2679*, Sept. 1999.
12. V.Paxson et al, "Framework for IP Performance Metrics," *IETF RFC 2330*, May 1998.
13. W.Kellerer, E. Steinbach, P. Eisert, and B. Girod, "A Real-Time Internet Streaming Media Testbed,"*Proceedings of the IEEE Int'l Conf. on Multimedia and Expo.*,Lausanne, Aug. 2002.
14. W. Jiang and H. Schulzrinne, "Modeling of packet loss and delay and their effect on real-time multimedia service quality," *NOSSDAV 2000*, Chapel Hill,NC, Jun. 2000.
15. V.Raisanen, G.Grotefield, and A.Morton, "Network performance measurement with periodic streams" *IETF RFC 3432*, Nov.2002.
16. J. Bolot, "End-to-End Packet Delay and Loss Behaviour in the Internet," *SIGCOMM 1993*, NY, Sept. 1993.
17. W. Jiang and H. Schulzrinne, "QoS measurement of Internet real-time multimedia services," *tech. report CUCS-05-99m, Columbia Univ.*, NY, Dec. 1999.
18. C.Boutremans and J.Boudec, "Adaptive Joint Playout Buffer and FEC Adjustment for Internet Telephony," *IEEE Proc. of Infocom 2003*, San Fran., Mar. 2003.
19. Recommendation G.107, "The E-model, a computational model for use in transmission planning," ITU, May 2000.
20. J. Janssen, D. De Vleeschauwer, M. Buchli, and G. Petit, "Assessing voice quality in packet-based telephony," *IEEE Internet Computing*, vol. 6,no. 3,May-June 2002.
21. W. Jiang and H. Schulzrinne, "Comparison and Optimization of Packet Loss Repair Methods on VoIP Perceived Quality under Bursty Loss," *NOSSDAV'02*, Miami, May 2002.
22. A. Clarke, "Modeling the effects of burst packet loss and recency on subjective voice quality," *Proc. of IP Telephony Workshop*, Mar. 2001.
23. Recommendation G.113, "Transmission impairments due to speech processing," ITU, Feb. 2001.
24. H. Melvin and L. Murphy, "Trace Driven Testing of the Hybrid Playout Strategy", *Dept. of IT, NUI,G, Tech.Report 0106003-2*, Jun. 2003.
25. W. Jiang and H. Schulzrinne,"QoS Evaluation of VoIP End-points," *IEEE Intl. Conf. on Comm. (ICC 2003)*,Anchorage, Alaska, May 2003.
26. Active Measurement project. See http://www.watt.nlanr.net/.