Adaptive VoIP Playout Scheduling: Assessing User Satisfaction

Miroslaw Narbutt
Technological University Dublin, miroslaw.narbutt@tudublin.ie

Andrew Kelly
Dublin City University, kellya@eeng.dcu.ie

Liam Murphy
University College Dublin, liam.murphy@ucd.ie

See next page for additional authors

Authors

Miroslaw Narbutt, Andrew Kelly, Liam Murphy, and Philip Perry

7-1-2006

Adaptive VoIP Playout Scheduling: Assessing User Satisfaction

Miroslaw Narbutt
Dublin Institute of Technology, narbutt@cnri.dit.ie

Andrew Kelly
Dublin City University, kellya@eeng.dcu.ie

Liam Murphy
University College Dublin, liam.murphy@ucd.ie

Philip Perry
University College Dublin, philip@perryradio.com

Adaptive VoIP Playout Scheduling: Assessing User Satisfaction

Delay and packet loss dramatically affect the quality of a voice-over-IP (VoIP) call and depend on the playout buffer scheme implemented at the receiver. The choice of playout algorithm can't be based on statistical metrics without considering the perceived end-to-end conversational speech quality. The authors present a method for evaluating various playout algorithms that extends the E-model concept by estimating user satisfaction from time-varying transmission impairments. This article evaluates several playout algorithms and shows a correspondence between their results and those obtained via statistical loss and delay metrics.

Miroslaw Narbutt
Dublin Institute of Technology

Andrew Kelly
Dublin City University

**Liam Murphy
and Philip Perry**
University College Dublin

Unlike conventional telephony, transmission impairments such as packet loss and delay can affect speech quality or a conversation's interactivity (conversational dynamics between speakers decreases as the delay increases, because when one-way end-to-end delay gets too great, talkers and listeners often fall out of sync, speaking at the same time or each waiting for the other to speak), thus affecting overall voice-over-IP (VoIP) communication. Such impairments depend largely on the playout buffer scheme implemented at the receiver, yet this buffer is vendor-specific and not governed by standards. Moreover, information about implementing playout buffers in commercial applications is practically nonexistent (such information holds a strategic value to the vendor), resulting in many different adaptive and fixed playout schemes from which to choose, each with a different parameter set. Given so many options, VoIP equipment designers and providers need a way to evaluate them.

Traditional subjective "listening-only" tests in which subjects listen to speech samples and grade their quality according

to an opinion scale don't take into account delay impairments, thus they can't assess a conversation's interactivity.[1] The Perceptual Evaluation of Speech Quality (PESQ) method[2] considers playout adaptation, but it doesn't include the absolute delay in its ratings and isn't recommended to assess speech-transmission quality.[3] Speech codec designers typically use both methods (listening-only tests and PESQ) for assessing narrow-band speech quality, but network planners must deal with delay-sensitive VoIP transmission. Thus, they have to rely on the International Telecommunications Union-Telecommunications (ITU-T) computation model for use in transmission planning, known as the E-model,[4] which takes into account static impairments (such as average mouth-to-ear delay and average packet loss) that don't consider the dynamics of adaptive playout buffering. The combination of PESQ and the E-model[5,6] could enhance the accuracy and efficiency of conversational speech quality evaluation, but it would require a reference signal and it doesn't work in real time.

In this article, we present a new scheme for evaluating various playout algorithms, which can be described as a short-time speech-transmission quality assessment (STSTQA).[7] Essentially, it extends the E-model concept, providing a direct link to the speech-transmission quality by estimating user satisfaction.

Adaptive Playout Scheduling

Large delay variations in IP networks can complicate the proper reconstruction of the speech signal at the receiver. To compensate for jitter, a typical VoIP application buffers incoming packets before playing them out, letting slower packets arrive on time and play out at their sender-generated rate. The optimal delay for this de-jitter buffer should be equal to the total variable delay along the connection. Unfortunately, it's impossible to find an optimal, fixed de-jitter buffer size when network conditions vary in time; thus, de-jitter buffers with a dynamic size allocation — so-called *adaptive playout buffers* — are more appropriate.

Compressing or expanding the silent periods between consecutive talk spurts adjusts the playout delay. With this type of "per-talkspurt" mechanism, the playout buffer module calculates the playout time for just the incoming talk spurt's first packet. Any variation in playout delay introduces artificially elongated or reduced silent periods between two consecutive talk spurts. The effectiveness of the per-talkspurt mechanisms is limit-ed when talk spurts are long and the network delay variation within them is high. Thus, some algorithms adjust the voice packets' playout time during voice activity by scaling individual voice packets. These packets can be scaled from 50 percent to 200 percent of their original size without degrading sound quality.[8]

A good playout algorithm should be able to keep the buffering delay as short as possible while minimizing the number of packets that arrive too late to be played out. These two conflicting goals have led to various adaptive playout algorithms, which we can group into four categories:

- algorithms that continuously estimate network delays and jitter to calculate playout deadlines (*reactive algorithms*);
- algorithms that maintain a histogram of packet delays and choose the optimal playout delay from it (*histogram-based algorithms*);
- algorithms that monitor the packet-loss ratio or buffer occupancy and adjust playout delay accordingly; and
- algorithms that aim to maximize user satisfaction.

Traditionally, the choice of a buffer algorithm was based purely on the trade-off between buffering delay and the resulting late-packet loss. Given that the purpose of playout buffering is to improve conversational speech quality, a more informed choice of algorithm can be made by considering its effect on user satisfaction.

A New Method for Assessing User Satisfaction

The E-model is a useful tool for estimating speech-transmission quality (in terms of a rating factor R) under various impairments. The rating factor R is defined as a linear combination of the individual impairments and is given by

$$R = (R_0 - I_S) - I_d - I_e + A. \qquad (1)$$

In the context of this work, delay impairment I_d (which captures the effect of delay and echo) and equipment impairment I_e (which captures the effect of information loss caused by encoding scheme and packet loss) are the most interesting.

Because other impairments — such as loud connection and quantization impairment I_S, the basic signal-to-noise ratio R_0, and the "advantage factor" (user willingness to accept some quality degrada-

R	User satisfaction	Speech-transmission quality
90 to 100	Very satisfied	Best
80 to 90	Satisfied	High
70 to 80	Some users dissatisfied	Medium
60 to 70	Many users dissatisfied	Low
50 to 60	Nearly all users dissatisfied	Poor
0 to 50	Not recommended	

tion in return for ease of access) A — are irrelevant for assessing speech-transmission quality, we can reduce the expression for the R rating to

$$R = 93.2 - I_d - I_e. \qquad (2)$$

Based on the R rating, ITU-T Recommendation G.109 also introduces categories of speech transmission quality and corresponding user satisfaction.[9] Table 1 defines those categories in terms of ranges of R.

Using Equation 2 and the categories of user satisfaction defined in Table 1,[9] we created contours of quality as a function of mouth-to-ear delay (assuming one echo level) and the packet-loss ratio (assuming a given encoding scheme). Quality contours are a crucial part of assessing overall user satisfaction. We first use a playout buffer module to calculate the playout delays and resulting packet loss with a specific playout algorithm for a given time interval (for example, 10 seconds). We can then map these playout delays and packet losses on a loss–delay plane that already has quality contours on it. The distribution of loss–delay points on the contours provides a direct link to perceived conversational speech quality.

Quality contours determine the rating factor R for all possible combinations of loss and delay; impairments I_d and I_e determine the contours' shapes. Figure 1 shows the quality contours for the G.711 encoding scheme (assuming bursty loss of packets) and for five different echo-loss levels (talker echo loudness rating [TELR] = 45, 50, 55, 60, and 65 decibels [dB]).

Algorithm Performance Comparison

To prove our method's effectiveness, we evaluated the performance of five different playout algorithms in an IEEE 802.11b WLAN — three reactive algorithms [10-12] and two histogram-based ones[13,14] — taking into account both the effect of various buffering schemes on the statistical loss–delay trade-off and the effect the schemes had on user satisfaction.

We started by establishing a VoIP call between two wireless hosts. To increase the delay and jitter in the WLAN cell, we used several wireless stations to generate background User Datagram Protocol traffic. The stations generated intermittent 1,500-byte packets at 500 Kbits per second. We used the simplest G.711 A-Law encoding scheme pulse code modulation (PCM); the terminal encoder sent one frame of audio (240 bytes) every 30 milliseconds (ms). We used a sequence of alternating audio signals and silent periods as an input signal (following the ITU-T P.59 recommendation[15]); no audio packets were generated during silent periods. For an hour of transmission, we collected all experimental data (packet arrival times, timestamps, sequence numbers, and marker bits) at the receiving terminal and processed it later (offline) with a program that simulates various playout algorithms' behavior. Figure 2 shows the influence of background traffic on delay and delay variation during the call.

Trade-Off between Loss and Delay

A good playout algorithm should be able to minimize buffering delay and late packet loss, thus improving the loss–delay trade-off. We can get a trade-off curve for a given playout algorithm by plotting the average buffering delays and late-loss percentages for the entire range of values of the algorithm's control parameter.[14] Once we have this curve, we can better judge which algorithm performs better: if the curve achieved with one algorithm falls below the curve achieved with a second one, then the first algorithm performs better.

In reactive algorithms (we used Ramjee's, Bolot's, and dynamic α), the β parameter with values between two and four controls the loss–delay trade-off. In histogram-based algorithms (we used Moon's and Concord), we can control the trade-off by specifying the desired packet loss rate (for example, from 0 percent to 10 percent). Figure 3 shows the trade-off between average buffering delay and the average late-packet-loss rate for the playout schemes we evaluated. In Figure 3a, the solid lines represent the basic Ramjee's algorithm's performance with fixed α (0.8, 0.9, and 0.998002), the line with triangles represents the Bolot algorithm's performance, and the line with circles rep-
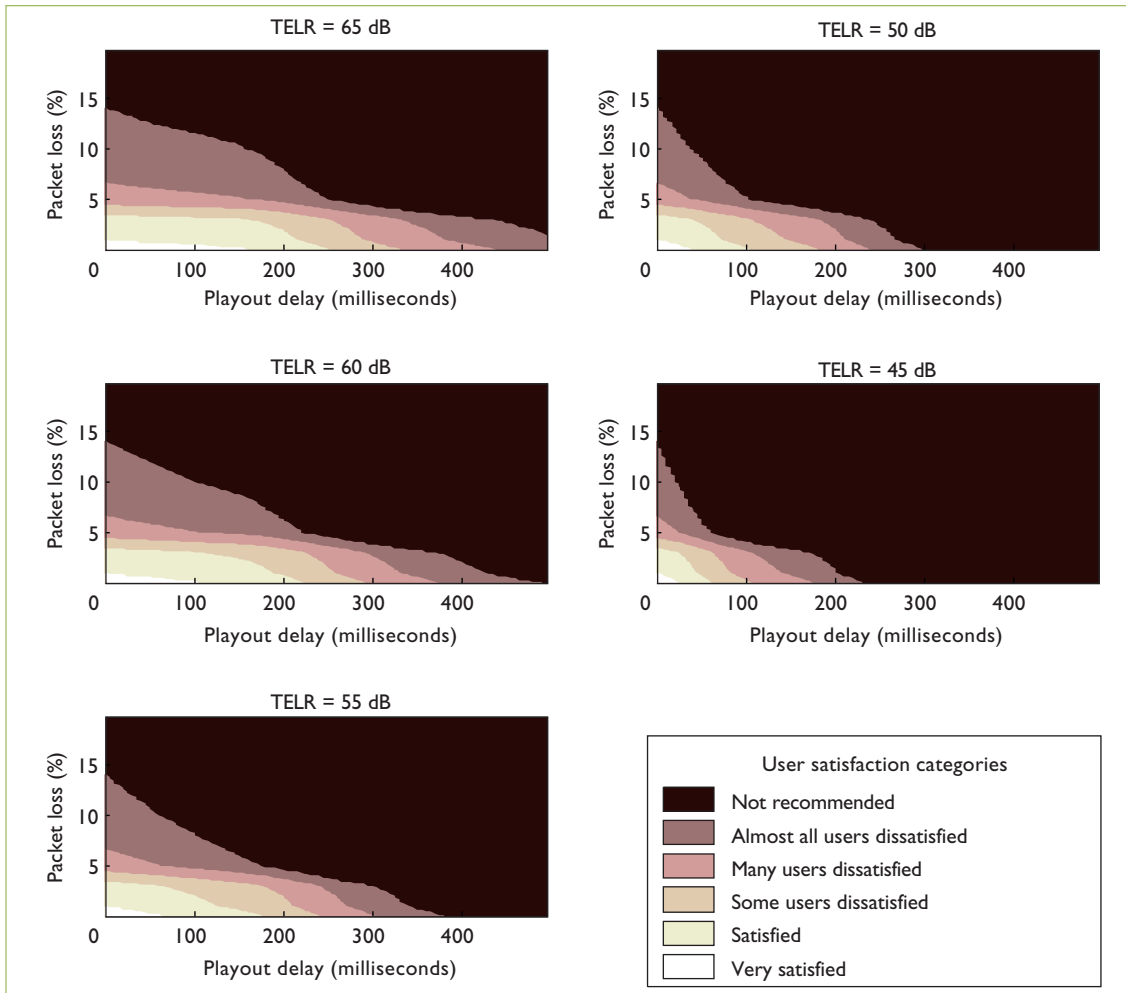
Figure 1. Quality contours. Assuming G.711 encoding (with codec's built-in packet-loss concealment and bursty packet loss), we can see that tolerable mouth-to-ear delay depends strongly on echo cancellation. Of particular interest here is the ability to find different combinations of loss and delay that result in the same user satisfaction. TELR stands for the talker echo loudness rating.

resents the dynamic α algorithm's performance. As we can see, the algorithm with dynamic α achieves a better loss–delay trade-off than the reactive algorithms do for the full range of β values.

In Figure 3b (next page), the solid lines represent the Moon's algorithm's performance, the line with triangles represents the Concord algorithm's performance, and the line with circles represents the algorithm with dynamic α's performance. (The number of samples in the Moon's histogram was 100, 200, 400, and 1,000.) Again, the dynamic α algorithm achieves a better loss–delay trade-off than the histogram-based algorithms.

User Satisfaction

Although a trade-off curve is useful from a statistical viewpoint, we can make a more informed choice of buffer algorithm by considering its effect on perceived speech quality.

Accordingly, we used our method to assess overall user satisfaction with these five buffering schemes. Assuming G.711 encoding with packet loss concealment (PLC), random loss, and echo cancellation implemented (TELR = 65 dB), we created quality contours that could determine the rating factor R for all possible combinations of loss and delay. We then calculated the average playout delay (that is, mouth to ear) and average packet loss for 10-second periods over the transmission. Figure 4 shows the loss–delay distribution on the quality contours and the resulting overall user satisfaction.

The figure shows that the dynamic α algorithm gave excellent user satisfaction 76 percent of the
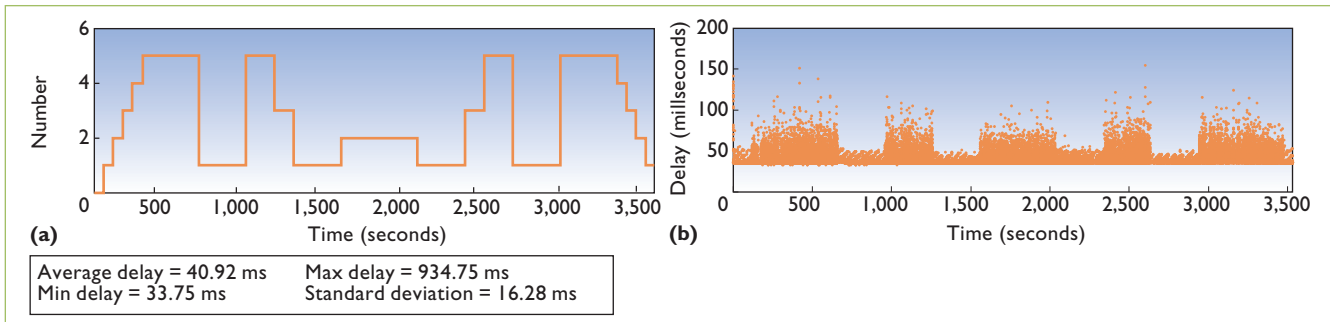
Figure 2. Background traffic. With (a) the level of background traffic during the one-hour VoIP call, (b) mouth-to-ear delay and jitter rises.
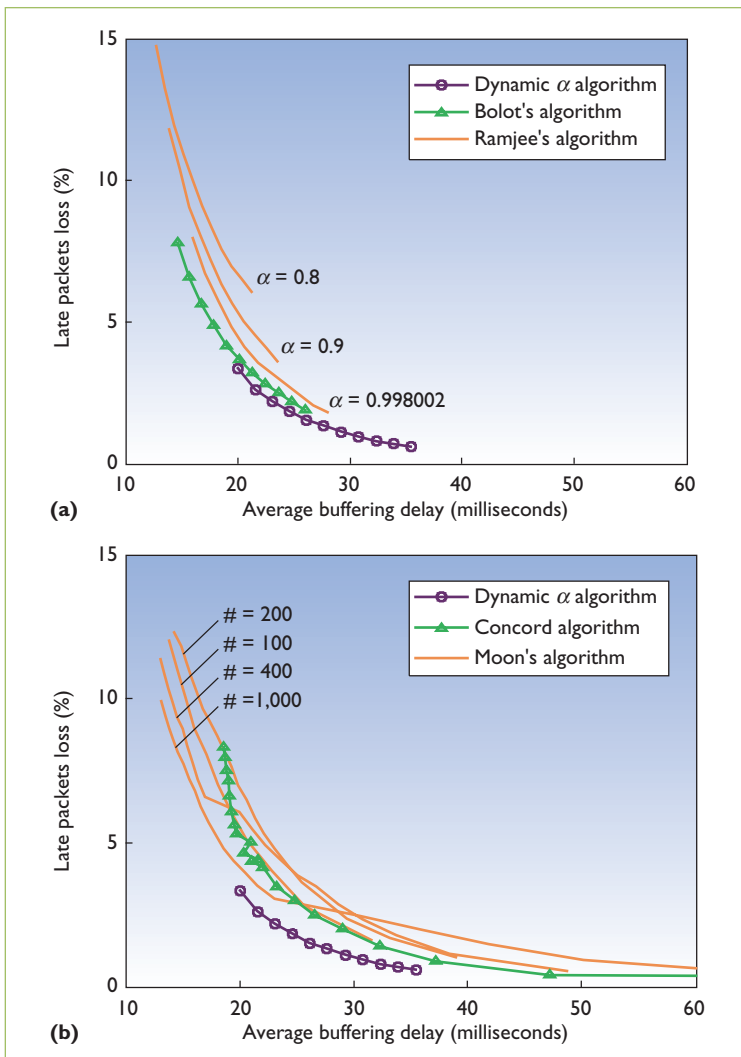

Figure 3. Loss rate and buffering delay. (a) For reactive algorithms, the dynamic α algorithm achieves a better loss–delay trade-off than Bolot or Ramjee for the full range of their control parameter β (from 2 to 4); (b) for histogram-based algorithms, again the dynamic α algorithm achieves a better loss–delay trade-off than Concord or Moon for the full range of their control parameter (desired loss from 0 to 10 percent).

time, compared to the basic Ramjee's algorithm at 45 percent (α = 0.998002), Bolot's at 30 percent, Moon's at 47 percent, and Concord at 57 percent. This corresponds to our results from the previous section in which dynamic α outperformed all the other algorithms. We believe that the correspondence between our proposed new method (which uses quality contours) and traditional metrics that employ statistical loss–delay trade-off (in the form of loss–delay curves) validates our proposed method.

The choice of buffer algorithm can strongly affect speech transmission quality, but we must remember that there is no "best" algorithm or parameter that can always achieve the best user satisfaction for all network conditions. However, with our pictorial representation of playout delays and resulting packet loss on quality contours (which gives a more detailed view of a given playout mechanism's performance), it's possible to find the best algorithm and parameter settings for a particular set of needs. Our method can work both in real time (for online monitoring of VoIP quality) and offline on prerecorded packet delays (for evaluation purposes) and doesn't require a reference speech signal. On the basis of experiments we've conducted, we believe our new method has significant potential for assessing the VoIP quality affected by playout-scheduling techniques.

References

1. *Recommendation ITU-T P.800, Methods for Subjective Determination of Transmission Quality*, Int'l Telecomm. Union-Telecomm., Aug. 1996; www.itu.int/itu-t/
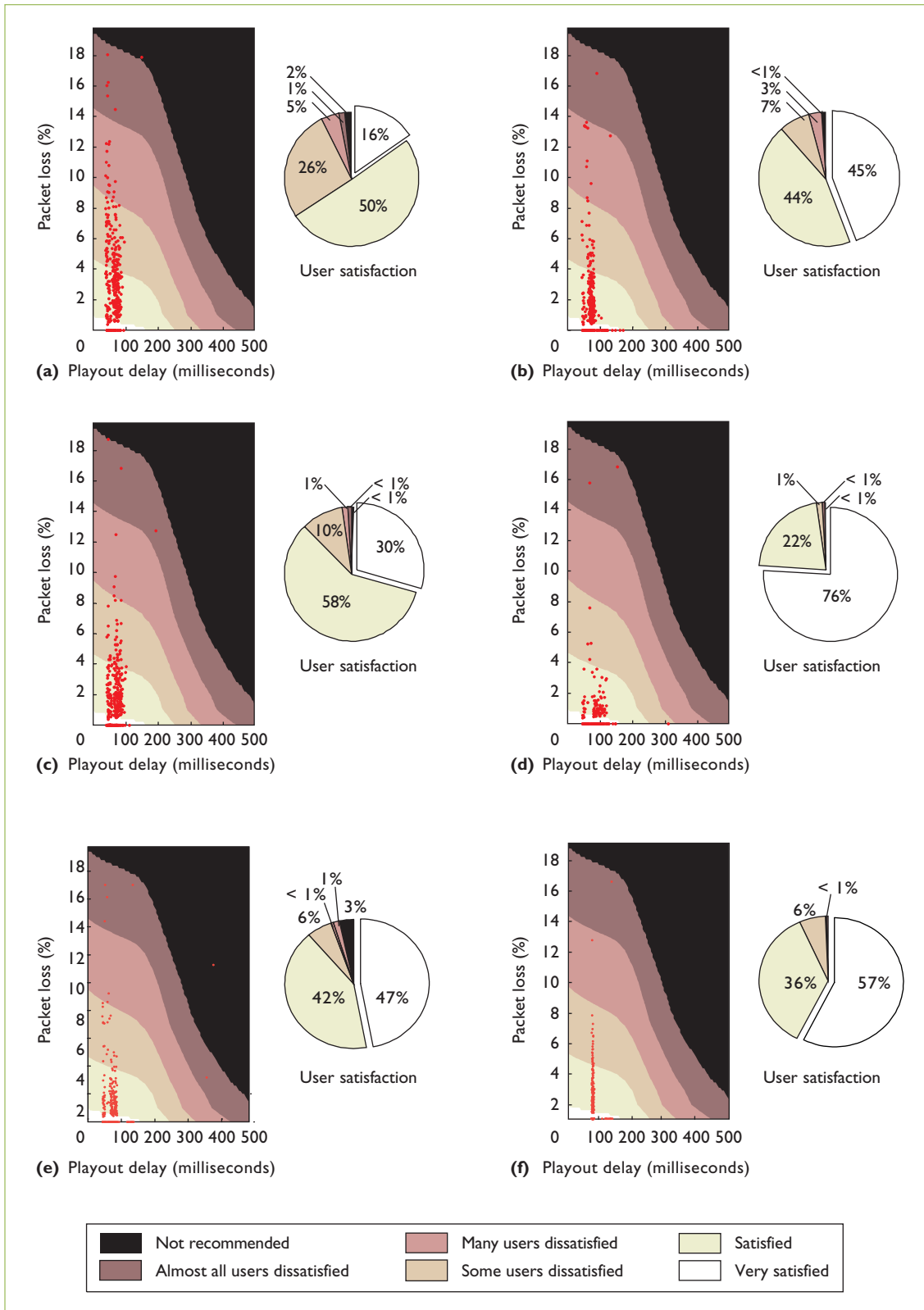
Figure 4. Playout delays and packet loss. (a) Ramjee's algorithm with $\alpha = 0.9$ gives excellent user satisfaction 16 percent of the time; (b) Ramjee's algorithm with $\alpha = 0.998002$ gives 45 percent excellent satisfaction; (c) Bolot's algorithm gives 30 percent; (d) the dynamic α algorithm gives 76 percent; (e) Moon's algorithm gives 47 percent; and (f) the Concord algorithm gives 57 percent.

publications/recs.html.

2. *Recommendation ITU-T P.862, Perceptual Evaluation of Speech Quality (PESQ), An Objective Method for End-to-End Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs*, Int'l Telecomm. Union-Telecomm., Feb. 2001; www.itu.int/itu-t/publications/recs.html.

3. A.W. Rix et al., "Perceptual Evaluation of Speech Quality (PESQ): A New Method for Speech Quality Assessment of Telephone Networks and Codecs," *Proc. IEEE Int'l Conf. Acoustics, Speech, and Signal Processing* (ICASSP 2001), IEEE CS Press, 2001, pp. 745–748.

4. *Recommendation ITU-T G.107, The E-Model, A Computational Model for Use in Transmission Planning*, Int'l Telecomm. Union-Telecomm., Mar. 2003; www.itu.int/itu-t/publications/recs.html.

5. L. Sun and E.C. Ifeachor, "Prediction of Perceived Conversational Speech Quality and Effects of Playout Buffer Algorithms," *Proc. IEEE Int'l Conf. Comm. 2003* (ICC 2003), IEEE CS Press, 2003, pp. 1–6; www.tech.plymouth.ac.uk/spmc/people/lfsun.

6. C. Hoene, S. Wiethölter, and A. Wolisz, "Predicting the Perceptual Service Quality Using a Trace of VoIP Packets," *Proc. 5th Int'l Workshop Quality of Future Internet Services 2004*, Springer, 2004, pp. 21–30.

7. M. Narbutt and M. Davis, "Assessing the Quality of VoIP Transmission Affected by Playout Buffer Scheme," *Proc. CTU/ETSI/IEE Measurement of Speech and Audio Quality in Networks Conf. 2005* (MESAQIN 2005), Czech Technical Univ., 2005; http://wireless.feld.cvut.cz/mesaqin/contributions.html.

8. Y.J. Liang, N. Färber, and B. Girod, "Adaptive Playout Scheduling and Loss Concealment for Voice Communication over IP Networks," *IEEE Trans. Multimedia*, vol. 5, no. 4, 2003, pp. 532–543; http://irms.stanford.edu/~liang/research/publications.

9. *Recommendation ITU-T G.109, Definition of Categories of Speech Transmission Quality*, Int'l Telecomm. Union-Telecomm., Sept. 1999; www.itu.int/itu-t/publications/recs.html.

10. R. Ramjee et al., "Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks," *Proc. IEEE Infocom 1994,* IEEE CS Press, 1994, pp. 680–688; http://citeseer.ist.psu.edu/ramjee94adaptive.html.

11. J.C. Bolot and A. Vega-Garcia, "Control Mechanisms for Packet Audio in the Internet," *Proc. IEEE Infocom 1996*, IEEE CS Press, 1996, pp. 232–239; http://citeserv.ist.psu.edu/bolot96control.html.

12. M. Narbutt and L. Murphy, "VoIP Playout Buffer Adjustment Using Adaptive Estimation of Network Delays," *Proc. 18th Int'l Teletraffic Congress* (ITC-18), Elsevier, 2003, pp. 1171–1180; www.eeng.dcu.ie/~narbutt.

13. N. Shivakumar et al., "The Concord Algorithm for Synchronization of Networked Multimedia Streams," *Proc. IEEE Int'l Conf. Multimedia Computing and Systems 1995* (ICMCS 95), IEEE CS Press, 1995, pp. 31–40; http://dbpubs.stanford.edu:8090/pub/1995-26.

14. S.B. Moon, J. Kurose, and D. Towsley, "Packet Audio Playout Delay Adjustment: Performance Bounds and Algorithms," *ACM/Springer Multimedia Systems*, vol. 6, no. 1, 1998, pp. 17–28.

15. *Recommendation ITU-T P.59, Artificial Conversational Speech*, Int'l Telecomm. Union-Telecomm., Mar. 1993; www.itu.int/itu-t/publications/recs.html.

Miroslaw Narbutt is a postdoctoral research fellow at the Dublin Institute of Technology. His technical interests include quality of service provisioning for real-time voice transmission in wired and wireless networks, adaptive playout scheduling for jitter compensation, and the proper tuning of communication protocol parameters to support real-time voice transmission in a wireless environment. Narbutt has a PhD in computer science from University College Dublin and an MSc in electronics and telecommunications from Poznan University of Technology, Poland. Contact him at narbutt@cnri.dit.ie.

Andrew Kelly just completed an MS in engineering with the Performance Engineering Laboratory from Dublin City University. His research interests include quality of service for voice transmission across wireless networks and transport-layer handover. Contact him at kellya@eeng.dcu.ie.

Philip Perry is an independent consultant radio engineer with expertise in radio communications system architecture, system performance evaluation, RF circuit and system design, MAC protocols, and data networking. His current research interests include VoIP delivery over heterogeneous wireless networks and the effect of hybrid ARQ used in cellular systems on real-time interactive traffic. Perry has held several positions in various universities and has worked with many companies, most notably Lucent Technologies and Ericsson Systems Expertise. Contact him at philip@perryradio.com.

Liam Murphy is a senior lecturer in computer science at University College Dublin, where he is director of the Performance Engineering Laboratory. His current research projects involve mobile and wireless systems, computer network convergence issues, and Web services performance issues. Murphy has a BE in electrical engineering from University College Dublin, and an MSc and a PhD in electrical engineering and computer sciences from the University of California, Berkeley. He is a member of the IEEE. Contact him at liam.murphy@ucd.ie.