Proceedings of the IASTED International Conference

Modelling and Simulation (MS '99)

May 5-8, 1999
Philadelphia, Pennsylvania - USA

Editor: M.H. Hamza

A Publication of the International Association of Science and Technology for Development - IASTED

IASTED/Acta Press
Anaheim ♦ Calglary ♦ Zürich

ISBN: 0-88986-247-8

Proceedings of the IASTED International Conference
Modeling and Simulation (MS '99)
May 5-8, 1999, Philadelphia, Pennsylvania - USA

MODELLING AND SIMULATION OF AN ADAPTIVE PLAYOUT DELAY CONTROL MECHANISM FOR PACKET AUDIO ACROSS THE INTERNET

MARCO ROCCETTI, VITTORIO GHINI, GIOVANNI PAU
Dipartimento di Scienze dell'Informazione, Università di Bologna
Via Mura A. Zamboni 7 - I-40127 Bologna ITALY
e-mail: roccetti@cs.unibo.it, ghini@cs.unibo.it, giovanni@csr.unibo.it

ABSTRACT

A novel mechanism for the dynamical adaptation of the playout delay of packetized audio in Internet voice-based connections has been presented which is based on the existence of a sufficient number of silent intervals (of sufficiently long duration) needed for effecting the dynamical setting of the playout delay [9]. In this paper, an eight-state Markov model for voice activity in conversational speech has been exploited in order to assess the adequacy of the proposed audio mechanism. Based on this model, we have carried out several simulation experiments that show that a sufficient number of silence periods (of sufficiently long duration) occur in a typical human conversation that permit an adequate application of the audio mechanism proposed in [9]. In addition, we report on several simulation experiments derived from the model that show that the proposed audio mechanism strikes a favorable balance between the average playout delay and the packet loss percentage experienced during audio conversations over the Internet.

Key words: Multimedia Systems, Communication Systems and Networks, Packet Audio, Modelling and Simulation of Packet Audio over the Internet

1. PACKET AUDIO OVER THE INTERNET

The audio tools that are used to transmit packet audio across the Internet (e.g. NeVot [1], vat [2], rat [3], the INRIA audio tool [4]) typically operate by periodically sampling audio streams at the sending host, packetizing them, and finally transmitting the obtained packets to the receiving host by using datagram based connections (e.g. IP/UDP/RTP). In order to compensate for variable network delays, a smoothing playout buffer is used at the receiver. Received audio packets, in fact, are first queued into the buffer, and then the playout of each packet is delayed for some quantity of time beyond the reception of the first packet. Typical playout control mechanisms adaptively adjust the playout delay in order to keep this (additional) delay as small as possible, while minimizing the number of packets delayed past the point at which they are scheduled to be played out. A critical trade-off exists between the length of the imposed additional delay and the amount of lost packets due to their late arrival: the longer the additional delay, the more likely it is that a packet will arrive before its scheduled playout time. Hence, if, on one side, a too large percentage of audio packet loss (over 5-10%) may impair the intelligibility of an audio transmission, on the other side, too large playout delays (e.g. more than 300-400 msec) may disrupt the interactivity of an audio conversation [5,15,16].

An audio segment may be considered as constituted of short bursts of energy (called talkspurts) separated by silence periods (during which no audio packet is generated). The above mentioned audio tools adopt a mechanism for adaptively adjusting the playout delays that keeps the same playout delay constant throughout a given talkspurt, but permits different playout delays in different talkspurts. However, the playout control mechanism that is adopted by most of those audio tools typically suffer from the main following problems [6,7]:

1) an ``external" software-based mechanism (e.g. the IP-based NTP protocol) is frequently used to synchronize the system clocks at both the sender and the receiver. Needless to say, low-frequency clock drift between the two hosts can cause the receiver buffer overflow or underflow. Unfortunately, the problem is that the adopted mechanisms are not so typically widespread all over the Internet, and, in addition, they may be too much inaccurate for coping with the real-time nature of the audio conversation;

2) the packet transmission delays experienced over the Internet are assumed to follow a Gaussian distribution. This assumption which used in all the cited tools to adjust the playout delay may not be a plausible conjecture. For example, recent experimental studies have indicated the presence of frequent and large end-to-end delay ``spikes" for periodically generated packets (as is the case with audio packets) [5,8].

In order to adequately support real-time voice over packet-switched networks (such as the Internet), we have designed and implemented an adaptive mechanism for the control of the playout delay that ameliorates all the negative effects of the audio tools mentioned above, while maintains satisfiable values of the tradeoff between the average playout delay and the packet loss due to late arrivals [9].

Such an audio mechanism dynamically adapts the playout delay of packetized audio in Internet voice-based connections but relies on the existence of a sufficient number of silent intervals (of sufficiently long duration) needed for effecting the dynamical setting of the playout delay [9,12]. In this paper, we have exploited an eight-state Markov model for voice activity in conversational speech in order to assess the efficacy of the proposed audio mechanism. Based on this model, we have developed several simulation experiments that show the existence of a sufficient number of silent intervals during typical voce-based conversational speeches. Further, those silence periods are both sufficiently long and frequent to adequately permit the dynamical adaptation of the playout delay which is required by the proposed audio mechanism. Finally, we present the results of additional simulation experiments (derived from the model) that show that the proposed audio mechanism strikes a favorable balance between the average playout delay and the packet loss percentage experienced during audio conversations. The remainder of this paper is structured as follows. In the next Section 2 we briefly recall the main design features of our mechanism, while Section 3 is devoted to present and discuss the adopted simulation model and the obtained simulation results. Finally, Section 4 concludes the paper by providing some final remarks and discussing future developments of our work.

2. A NOVEL ADAPTIVE MECHANISM FOR PACKET AUDIO

In [9] a playout delay control mechanism is presented that is suitable for adjusting the talkspurt playout delays of unicast, voice-based audio communications across the Internet. The mechanism was designed to dynamically adjust the talkspurt playout delays to the network traffic conditions without assuming neither the existence of an external mechanism for maintaining an accurate clock synchronization between the sender and the receiver, nor a specific distribution of the end-to-end transmission delays experienced by the audio packets. Succinctly, the technique for dynamically adjusting the talkspurt playout delay is based on obtaining, in periodic intervals (about 1 second), an estimation of the upper bound for the packet transmission delays experienced during an audio communication. Such an upper bound is periodically computed using round trip time values obtained from packet exchanges of a three-way handshake protocol performed between the sender and the receiver of the audio communication. At the end of such protocol, the receiver is provided with the sender's estimate of an upper bound for the transmission delay that can be used in order to dynamically compute the talkspurt playout delay and to adjust the receiver buffer dimension. The proposed mechanism guarantees that the talkspurt playout delay may be dynamically set from one talkspurt to the next without causing ``gaps" or ``time collisions" (formally defined in [9]) inside the talkspurt themselves, provided that intervening silence periods of sufficiently long duration are exploited for the adjustment.

Summarizing, the main features of the designed playout delay control mechanism are the following:

1) an internal and accurate technique that is able to maintain tight time synchronization between the system clocks of both the sender and the receiver;

2) a method for adaptively estimating the audio packet playout time (on a per-talkspurt basis) with an associated minimal computational overhead;

3) an exact and simple method for dimensioning the playout buffer (avoiding both buffer overflow and underflow) based on the measurement of the traffic conditions of the underlying network.

3. SIMULATION MODEL OF THE MECHANISM AND EXPERIMENTAL RESULTS

A simulation model has been implemented, using the C programming language, which is suitable for evaluating the performance of the designed playout control mechanism over a number of measured audio delay traces. In particular, a simulator has been developed that reads in the transmission delay of each packet from a given real audio trace, detects if it has arrived before the playout time that is computed by the playout control mechanism, and executes the algorithm. The simulator is also able to calculate the average playout delay and the packet loss for each given trace. In addition, the Unix socket interface and the datagram-based UDP protocol were used to transmit and receive the audio packets of each obtained audio delay trace. The coding scheme that was used to produce the audio packets use 8-kHz sampled speech at a bit rate of 8-kb/sec. In particular, all the audio packets used to perform the measurements were produced using a codec based on the ITU-T G.729 standard that provides coding of speech at 8-kb/sec while maintaining satisfiable audio quality [10]. Several audio delay traces (about 30) were obtained using the above mentioned software package. Each audio delay trace was obtained by transmitting (from Cesena, Italy, to Geneva, Switzerland) about 15.000 audio packets generated from prerecorded 10 minutes long audio files. After obtaining the audio traces, we run the simulator on each obtained audio delay trace in order to evaluate the performance of the designed playout delay control mechanism. The obtained results may be summarized as follows:

1) The percentage of lost packets obtained with our adaptive control mechanism was 5% - 7% in average over all the obtained audio traces. This result may be contrasted with the percentage of audio packets (20% - 23%) that would be lost if a constant playout delay of 150 msec were used for each packet.

2) The average playout delay obtained over all the audio traces was calculated as ranging between 80 and 250

msec, and only rarely playout delay spikes exceeding 600/700 msec were imposed by our mechanism. This playout delay value may be considered tolerable for audio conversations and guarantees a good degree of interactivity.

In order to better assess the performance of our mechanism we carried out an additional simulation experiment where our algorithm was simulated using the receiver buffer size as the control parameter to be varied to achieve different loss percentages. Using this simulation technique, the correspondent average playout delays were obtained as a function of the loss percentages. In Figure 1, such a playout delay is plotted as a function of the loss percentage. The plot of the playout delay has been obtained by running the simulator over all the 30 experimental traces of our experiments, and then averaging the results. In order to provide the reader with an understanding of the effect that various delay and loss rates (as well as buffer dynamics) have on the quality of the perceived audio, we have reported an approximate and intuitive representation of three different ranges for the quality of the perceived audio [16]. The three following audio quality ranges have been used: ``good" for delays of less than 200/250 msec and low loss rate, ``potentially useful" for delays of about 300-350 msec and higher loss rates, and finally ``poor" for delays larger than 350 msec and very high loss rates. As seen from the figure, and based on the consideration that audio of acceptable quality may be obtained only if lower delays are achieved while the loss percentage does not exceed the value of 10%, we can deduce that our algorithm shows very good performance.

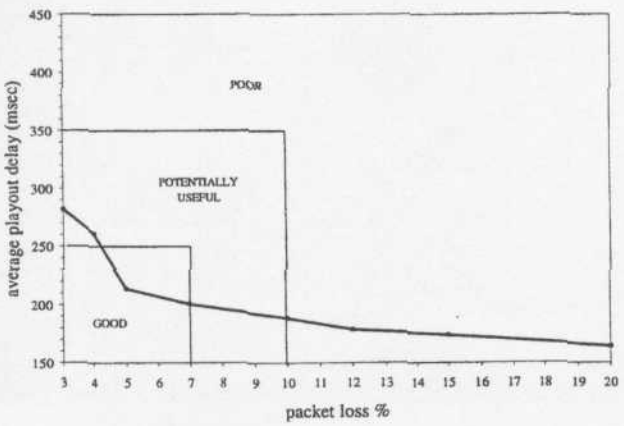

FIG. 1. PERFORMANCE OF OUR ALGORYTHM.

The need of silent intervals for allowing a playout delay control mechanism to adjust to the fluctuating network conditions is common to the most part of the existing audio tools [1,2,3,4] but renders our proposed scheme particularly relevant for voice-based applications where conversational audio with intervening silence periods between subsequent talkspurts is transmitted. Hence, in order to assess the efficacy of our mechanism, an accurate model of the talkspurt/silence characteristics of conversational speech is necessary. In particular, an accurate modelling of the voice activity characteristics of conversational speech is mandatory for understanding whether sufficient (and sufficiently long) silent intervals occur in typical human conversations that may permit the periodical activity of dynamically setting the playout delay from one talkspurt to the next as required by our mechanism.

To this aim, a modified eight-state Brady's model of conversational speech was adopted [11] that is able to describe the main on-off characteristics of human conversations. Based on this model, a set of simulation experiments have been carried out regarding, respectively, the overall quantity, the duration and the frequency of silence intervals within conversational speech.

As a first result, we obtained that the total quantity of silent intervals within a simulated two-party one-hour-long packetized conversation amounts to about 63-66% depending on the packetization interval that is typically chosen in the range of [10-30] msec. This result is summarized in Figure 2 where the total number of silent intervals (with relative duration) is shown, as obtained in a simulated one-hour-long two-party conversation. As seen from the figure, the smaller the packet size (i.e. 10 msec), the larger the number of silent intervals (i.e. 5075) we obtain.

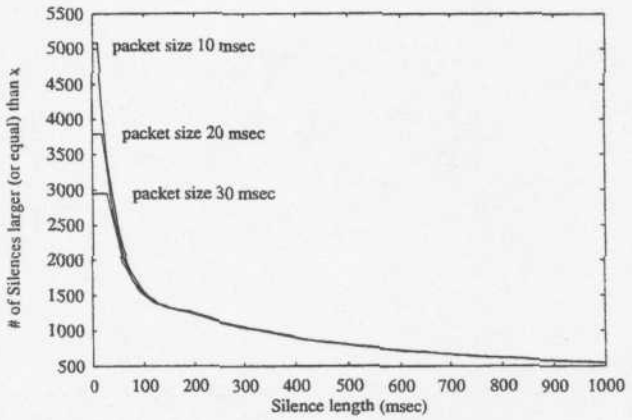

FIG. 2. # OF SILENT INTERVALS.

A second important result obtained from the simulation experiments we carried out concerns the duration (or length) of the intervening silence periods in a human conversation. The average silence length of the silence periods obtained in a simulated two-party one-hour-long conversation was measured as ranging in the interval [465-770] msec, yet again depending on the packetization interval. In particular, the smaller the packet size (i.e. 10 msec), the smaller the average silence duration (i.e. 465 msec). The main results regarding the silence interval duration are summarized in the Figures 3 and 4 where, respectively, the probability distribution of the silence interval duration, and the quantity of silent periods (of length larger than a fixed amount) out of the total quantity of all the obtained silence periods are reported.

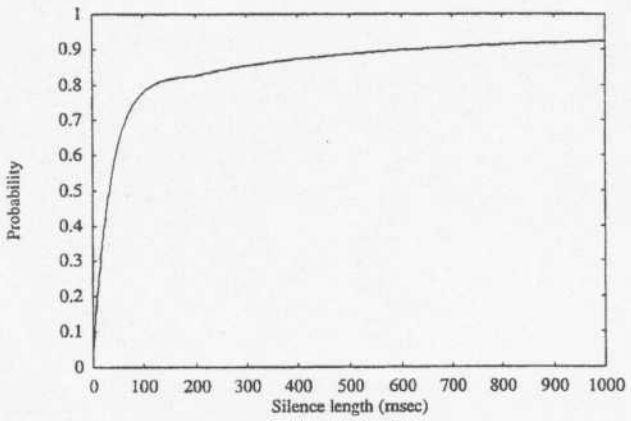

FIG. 3. SILENCE LENGTH.

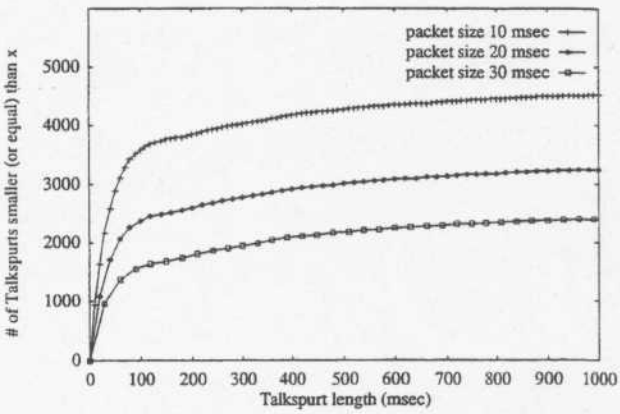

FIG. 5. # OF TALKSPURTS.

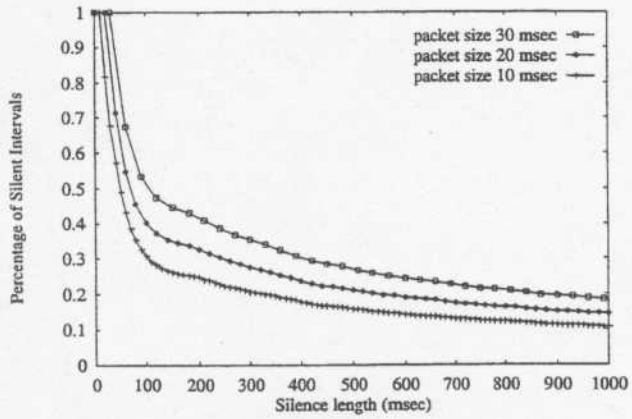

FIG. 4. PERCENTAGE OF SILENT INTERVALS (WITH DURATION LARGER THAN X) W.R.T. THE TOTAL # OF SILENCE PERIODS.

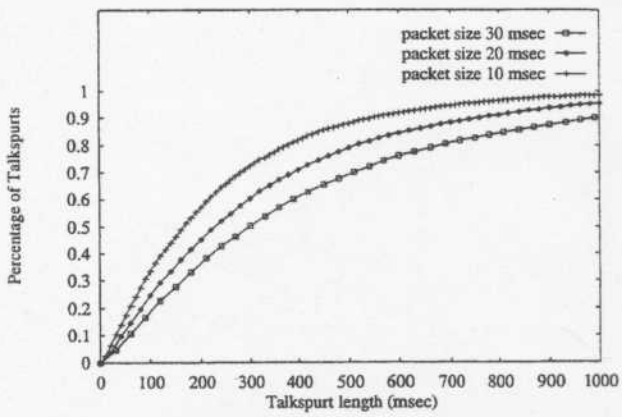

FIG. 6. PERCENTAGE OF TALKSPURTS (WITH DURATION SMALLER THAN X) W.R.T. THE TOTAL # OF TALKSPURTS.

Finally, another important parameter that influence the efficacy of the proposed playout delay control mechanism is the frequency of the intervening silence periods. Needless to say, the larger this frequency, the more likely it is that our mechanism will be successful in dynamically adjusting the playout delay. To this aim, we used the modified Brady's model to understand how many talkspurts are expected in a simulated two-party one-hour-long packetized conversation. From this simulative experiment, the following important result was obtained: the smaller the chosen packet size, the more likely it is that our mechanism will have the possibility of dynamically setting the playout delay, since the total number of silence periods increase and the average talkspurt length decreases to about 244 msec. The main results concerning the quantity and the duration of the talkspurts are depicted in the Figures 5 and 6 where, respectively, the total quantity of packetized talkspurts (with duration smaller than a fixed amount), and the quantity of talkspurts (with length smaller than a fixed amount) out of the total quantity of all the talkspurts are reported.

4. CONCLUDING REMARKS AND FUTURE WORK

In this paper, we have reported on the use of an eight-state Markov model for voice activity in conversational speech which has been exploited in order to assess the performance of an adaptive playout delay control mechanism recently proposed [9,12,13]. One of the most critical characteristic of this audio mechanism is its reliance on the existence of silent intervals (of sufficiently long duration) for effecting the dynamical setting of the playout delay.

Based on the use of the aforementioned model, we have conducted several simulation experiments that show that a sufficient number of silence periods occur during a human conversation (about 66%). In addition, those silent intervals result to be both sufficiently long and frequent so as to permit an adequate application of the proposed audio mechanism. In addition, further simulation experiments jointly conducted with experimental measurements have been reported that show that the proposed audio mechanism strikes a favorable balance between the average playout delay and the packet loss percentage experienced during audio conversations over the Internet.

A possible issue that remains open for further research includes the development of appropriate modifications of the playout delay control mechanism proposed in [9] that take into direct account the simple, accurate, and comprehensive simulative results we have obtained regarding human voice activity in conversational speech.

ACKNOWLEDGEMENTS

This research has been partially funded by Italian MURST and Italian CNR under the grant n. 98.00387.CT12.

REFERENCES

[1] H. Schulzrinne, Voice Communication across the Internet: a Network Voice Terminal, Tech. Rep., Dept. of ECE and CS Univ. of Massachusetts, Amherst (MA), 1992.

[2] V. Jacobson, S. McCanne, vat, ftp://ftp.ee.lbl.gov/conferencing/vat/ .

[3] V. Hardman, M.A. Sasse, I. Kouvelas, Successful MultiParty Audio Communication over the Internet, *Communications of the ACM* 41(5), 1998, 74-80.

[4] J. Bolot, A. Vega Garcia, Control Mechanism for Packet audio in the Internet, *Proc. of IEEE SIGCOMM '96*, San Francisco (CA), 1996.

[5] J. Bolot, H. Crepin, A. Vega Garcia, Analysis of Audio Packet Loss on the Internet, *Proc. of Network and Operating System Support for Digital Audio and Video*, Durham (NC), 1995, 163-174.

[6] A. Vega Garcia, Mecanismes de Controle pour la Transmission de l'Audio sur l'Internet, Doctoral Thesis in Computer Science, University of Nice Sophia Antipolis, Ecole Doctoral SPI, 1996.

[7] S.B. Moon, J. Kurose, D. Towsley, Packet Audio Playout Delay Adjustment: Performance Bounds and Algorithms, *ACM Multimedia Systems* 6, 1998, 17-28.

[8] W.E. Leland, M.S. Taqqu, W. Willinger, D.V. Wilson, On the Self Similar Nature of Ethernet Traffic, *IEEE/ACM Trans. on Networking* 2, 1994, 1-15.

[9] M. Roccetti, V. Ghini, G. Pau, P. Salomoni, M.E Bonfigli, Design and Experimental Evaluation of an Adaptive Playout Delay Control Mechanism for Packetized Audio for use over the Internet, UBLCS Technical Report n. 98-4 , Laboratory for Computer Science, University of Bologna, May 1998.

[10] ITU-T Recommendation G.729, Coding of Speech at 8 kb/s using Conjugate Structure Algebraic Code Excited Linear Prediction, 1996.

[11] H.P. Stern, S.A. Mahmoud, K.K Wong, A Comprehensive Model for Voice Activity in Conversational Speech - Development and Application to Performance Analysis of New-Generation Wireless Communication Systems, *Wireless Networks* 2(4), 1996, 359-367.

[12] M. Roccetti, Experimenting with Real-Time Packetized Audio for Distance Learning over Wide Area Networks, *Proc. of 1999 Western MultiConference on Computer Simulation*, San Francisco (CA), 1999, 155-160.

[13] M. Roccetti, M. Bernardo & R. Gorrieri, Packetized Audio for Industrial Applications: a Simulation Study, *Proc. of 10th European Simulation Symposium*, Nottingham (UK), 1998, 495-500.

[14] A. Eleftheriadis, S. Pejhan, D. Anastassiou, Architecture and Algorithms of the Xphone Multimedia. Communication System, *ACM Multimedia Systems Journal*, 2, 1994, 89-100.

[15] F. Panzieri, M. Roccetti, Synchronization Support and Group-Membership Services for Reliable Distributed Multimedia Applications, *ACM Multimedia Systems Journal*, 5, 1997, 1-22.

[16] T.J. Kostas, M.S. Borella, I. Sidhu, G.M. Schuster, J. Grabiec, J. Mahler, Real-Time Voice over Packet-Switched Networks, *IEEE Network*, 12, 1998, 18-27.