A NEW TECHNIQUE FOR IMPROVING QUALITY OF SPEECH IN VOICE OVER IP USING TIME-SCALE MODIFICATION

Samar Agnihotri [§], *K. Aravindhan* [†], *H. S. Jamadagni* [§], *B. I. Pawate* [†]

[§]CEDT, Indian Institute of Science, Bangalore, Karnataka-560012, India
email : {samar, hsjam}@cedt.iisc.ernet.in
[†]DSP Applications Group, Texas Instruments India Ltd., Bangalore, Karnataka-560017, India
email : {k-aravindhanl, b-pawate}@ti.com

ABSTRACT

Packet arrival-delay variations and losses seriously affect the quality of voice delivered in VoIP. In this paper, using a time-scale modification algorithm, an integrated scheme is proposed to handle these impairments without introducing additional delays. This scheme provides flexible arrival-delay cut-offs to late arriving packets, reducing the packet loss-rate at the receiver. Further, the lost packets are concealed effectively. Extensive simulations have shown that the proposed scheme delivers high-quality speech across widely varying packet arrival-delays and loss-rates. The proposed scheme is fully receiver-based and with its low computational complexity and generic nature, is applicable to any VoIP system.

1. INTRODUCTION

Voice over Internet Protocol (VoIP) is an emerging technology that enables the transport of voice over public Internet. It holds the promise for bringing together both voice and data networks and for creating the opportunities for various value-added services. The quality of delivered voice, however, is an important issue due to loss and end-to-end delay variations (commonly called *jitter*) introduced in the packet stream as it traverses the Internet. For non real-time applications, such as Telnet, FTP, email etc, these problems are much lesser issues as reliable delivery **of** data is ensured by TCP (Transmission Control Protocol). TCP is a connection-orientedprotocol that uses "best-effort" service offered by underlying IP network-layer. It incorporates various retransmission strategies based on feedback and time-out mechanisms for reliable data transfer. However, except in few cases [1], TCP cannot provide strict loss and delay guarantees demanded by real-time applications – like VoIP.

For VoIP, packet loss leads to the loss of portions **of** speech, that results in poor quality of delivered voice. The total packet loss experienced by the receiver has two components – packet loss occurring in the network due to congestion at some intermediate node(s), and at the receiver due to packets arriving later than their scheduled playout times. Since the emergence of the concept of packetized voice in late 70's, extensive work has been done to reduce packet loss and its effects [1, 2]. The proposed schemes, can be classified as sender-receiver based, network-based and receiver-based. Our work focusses on receiver-based schemes, as those are generic and do not incur additional data or computational overhead at the sender or network.

Commonly, jitter compensation is carried out at the receiver by maintaining a playout buffer to produce continuous playout, but this introduces an additional buffering delay. However, better performance can be provided by adaptive playout buffer schemes **[3, 4, 5,** *6*, **7]**. Most of these schemes schedule the playout **of** a talkspurt based on the packet delay statistics of previous talkspurts. The performance of these schemes is limited by potentially high buffering delays introduced and poor quality of speech delivered when schemes such as splicing, silence or noise substitution, and packet-repetition, are used to conceal the packet losses in the middle of a talkspurt.

So, to improve the quality of delivered voice an integrated scheme is desired that reduces receiver-side packet loss due to delay jitter. **Also,** the lost packets should be concealed as much as possible. Further, the scheme should achieve these goals without introducing additional delays.

2. GLS-TSM ALGORITHM

In this work, the proposal is to use a class of speech processing algorithms, called Time-Scale Modification **(TSM),** to improve the delivered voice quality in VoIP systems.

Time-scale modification of speech refers to processing performed on speech signals that changes the perceived rate of articulation without affecting the pitch or intelligibility of the speech. Schemes exist, both in time and frequency domains, to perform time-scale modification of speech. The most popular among these schemes, due to its low compu-

0-7803-7402-9/02/$17.00 02002 IEEE

tational loading and good speech quality delivered, is ***Synchronized Overlap and Add** (SOLA)* algorithm of Roucos and Wilgus [8]. **A** detailed analysis of *SOLA* can be found in [9]. In the proposed scheme, an improved low-complexity variant of SOLA, called ***Global Local Search Time Scale Modification (GLS-TSM)*** algorithm [10] has been used.

SOLA algorithm extracts N samples from input (analysis) signal $x[n]$ at interval S_a on per frame basis and constructs output (synthesis) signal $y[n]$ at every S_s samples. The synthesis frame length **(S,)** and analysis frame length (S_a) are related by : $S_s =$ ***a*** $*$ S„ where α is time-scale modification factor ***(a*< 1** for compression and ***a*** **> 1** for expansion). Two signals are aligned and combined at the point of highest similarity to maintain the original pitch information and to eliminate clicks, noise and reverberations.

One major drawback of **SOLA** algorithm is that it uses computationally expensive normalized cross-correlation function to search for the point of highest similarity. The normalized cross-correlation needs to be calculated for every alignment point k, during the search for highest similarity in the search window of length at least one pitch-period [9].

GLS-TSM algorithm avoids this search method by searching for the point of best alignment in two steps. Firstly, ***global similarity*** or similarity over a time interval between analysis and synthesis frames, is searched by comparing their zero-crossing rates. Then, ***local similarity*** or similarity about a sample point is searched by minimizing L_1-norm distance between *two* frames. This distance measure is defined as :

$$d_{k,i} = \frac{1}{11}\sum_{j=1}^{11} |f_x[j] - f_{y,i}[j]| \qquad (1)$$

where k is the index at which zero-crossing starts, $f_x[j]$ is the j^{th} component of the feature vector at a zero-crossing point in $x[n]$ and $f_{y,i}[j]$ is the j^{th} component of the feature vector of the i^{th} zero-cross point in $y[n]$. Further, f is an eleven dimensional feature vector to represent local information in the neighborhood of a zero-crossing point [10]. This improvement over SOLA gives ***an order of magnitude increase*** in processing speed [10].

Once the point of best alignment is located, the output signal $y[n]$, is formed by fading-in the analysis frame and fading-out synthesis frame in the overlapping interval L_m, and then duplicating input frame until all N samples are exhausted, as shown in following equation :

$$y[S_s + k + j] = (1 - g[j])y[S_s + k + j] + g[j]x[j] \qquad 0 \le j < L_m$$

$$y[S_s + k + j] = x[j],\ L_m \le j < N \qquad (2)$$

where $g(.)$ is the function for fade-in gain. In this work, linear fade-in function given by eqn **3,** is used :

$$g[j] = j/L_m\ ,\ 0 \le\ j\ <\ L_m \qquad (3)$$

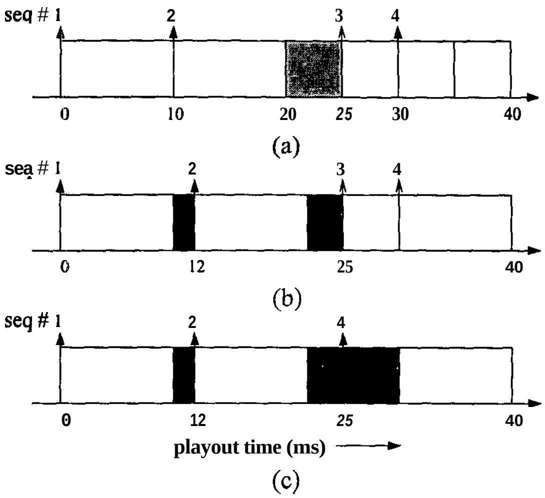

Fig. 1. Working of the proposed scheme with packet delay jitter (a) and **loss** (b)

3. ADAPTING TIME-SCALE MODIFICATION FOR VOICE OVER IP

Previously, time-scale modification has been used to improve the quality of voice in VoIP [11, 12]. But, that work was only aimed at concealing the lost packets.

The limited change in articulation rate of speech does not affect its perceived quality [9]. The proposed scheme exploits this observation to reduce the effects of packet loss and delay jitter on the quality of delivered voice. It is achieved by the time-scale modification **of** *successfully* received packets present in the playout buffer.

The GLS-TSM algorithm as described in [10], was designed to work with large speech segments (S_a of length 1600 samples or 200 ms) and ***apriori*** fixed value **of *a*.** But in VoIP, much smaller packet sizes are used, typically 10 to **40** ms. **To** modify the time-scale **of** these packets, S, must be of this size to avoid introducing extra packet holdup delays. Further, ***a*** may change on a per packet basis depending upon packet arrival pattern. ***So,*** the operational ranges of various parameters of GLS-TSM algorithm had to be redefined to match the requirements of much shorter signal segments and per packet variability of α.

In this work, redefined parameter values of GLS-TSM algorithm have been used. Further, based on the packet arrival pattern at the receiver, a new scheme has been developed to calculate ***a*** on a packet by packet basis.

The working **of** the proposed scheme is detailed in the following discussion. For illustration purpose, 10 ms voice packets, and playout buffer of **3** packets are used.

In figure 1(a), the packet **3** is delayed by **5** ms. The proposed scheme allows it to incur this delay and still accepts it **for** playout by time-expanding packet 1 with ***a*** **= 1.2** and packet 2 with $a = 1.3$, respectively. *So,* packet **3** is available when playout of packet 2 ends. But, as **5** ms of packet 3 is already compensated by packets 1 and 2, packet **3** is compressed with $a = 0.5$, Fig. 1(b). In general, if the situation arises where a packet is to be compressed with $a < 0.5$, it is compressed down to $\alpha = 0.5$ only, and rest of the compression is shared by subsequent packets. This step is necessary to maintain synchronization, and yet, not to degrade voice quality by excessively compressing a packet.

If packet **3** does not arrive at its rescheduled playout time, it is supposed to be lost, Fig. 1(c). At this stage, arrival of any packet with sequence number >**3**, is checked. For buffer size of **3** packets, this needs checking for the arrival of packets with sequence numbers 4 or ***5***. If packet 4 has arrived then it is expanded to compensate for residual playout time of packet **3.** If expansion is more than 1.5 times, then subsequent packets also take part in compensation to avoid delivering poor quality voice with annoying reverberations resulting from excessive time-expansion of a single packet. In Fig. 1(c), packet **4** is expanded with $a = 1.5$ to compensate for remaining ***5*** ms of packet **3.** In the worst case, if both packets 4 and ***5*** don't arrive, then samples from the previously successfully received packet, are repeated to compensate for residual playout time with smoothing of waveform at packet boundaries. This reduces the number of fully-repeated packets and eliminates the waveform mismatch at packet boundaries.

In this way, without introducing additional delays, the proposed scheme is able to give more acceptable delay to late arriving packets and conceal the lost packets effectively.

4. SIMULATION RESULTS

As all TSM schemes exploit the quasi-stationarity of speech, the packet length must be less than a phoneme. In this work, voice packets size was 10 ms (80 samples for $8\,kHz$ sampling rate). The input speech material was taken from *TIMIT* database. The files were downsampled to **8** kHz and two files were concatenated to create a reference speech file of duration 6–8 secs, recommended in [13].

The performance of the proposed scheme was compared with an adaptive jitter buffer **(AJB)** scheme, that maintains the running estimate ***of*** average network jitter of the session and adapts the buffer size using following set of equations :

$$\left.\begin{array}{lll} \bar{v}_i & = \alpha\bar{v}_{i-1} + (1-\alpha)|d_i - d_{i-1}| & \ldots(a) \\ nd & = \beta * \bar{v}_i & \cdots(b) \\ nd & = min\{maxDelay, max\{minDelay, nd\}\} & \\ \bar{v}_i & = nd/\beta & \cdots(c) \end{array}\right\} \quad (4)$$

where, $\bar{v}_i$ is the average network jitter estimate upto i^{th} packet, d_i is the network delay experienced by i^{th} packet, a is the weighting factor that controls the convergence of the algorithm, β is the adaptation factor, and nd is the nominal delay (the instantaneously adapted buffer size) which was bounded from above and below by *maxDelay* and *minDelay,* respectively. For simulation purposes, a was set to 0.99805, β to 8, *maxDelay* to 30ms, and *minDelay* to 10ms.

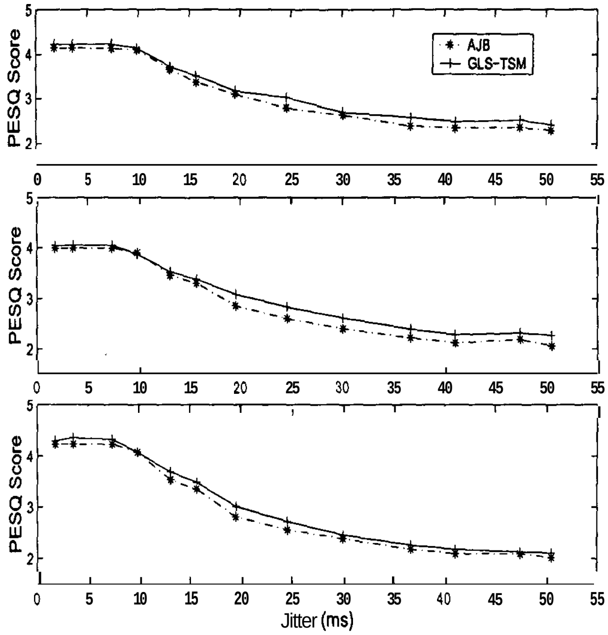

Fig. 2. MOS Predictions for 3 representative input-files with packet arrival jitter (Zero percent packet loss at network)

For proposed algorithm, playout buffer size was fixed at 30 ms. Additional delay allowed per packet was set to 4 ms and TSM frame size was set to **3** packets **(240** samples).

The performances of two schemes were compared for different delay jitters and loss rates. For delay jitter, delays were Normally distributed with mean 500 ms and varying standard deviations. The average jitter was calculated as defined in [14]. For packet-loss at network, input packet streams with varying random loss percentages were used.

To objectively test the quality of speech delivered, latest ITU-T recommendation P.862 [15] was used to predict subjective Mean Opinion Scores (MOS).

The redefined parameter ranges of **GLS-TSM** algorithm produced good quality speech for $0.5 \leq a \leq 2.0$. Simulation results for the quality of delivered voice are plotted in figures 2 and 3. These plots show that the proposed scheme, with its flexible delay cut-offs and improved loss concealment mechanism, delivers better quality speech. Further, it can be seen that the quality of delivered speech depends on network conditions as well as on input speech. Apart from objective measurements using PESQ, informal listening tests were also conducted, which showed that the per-

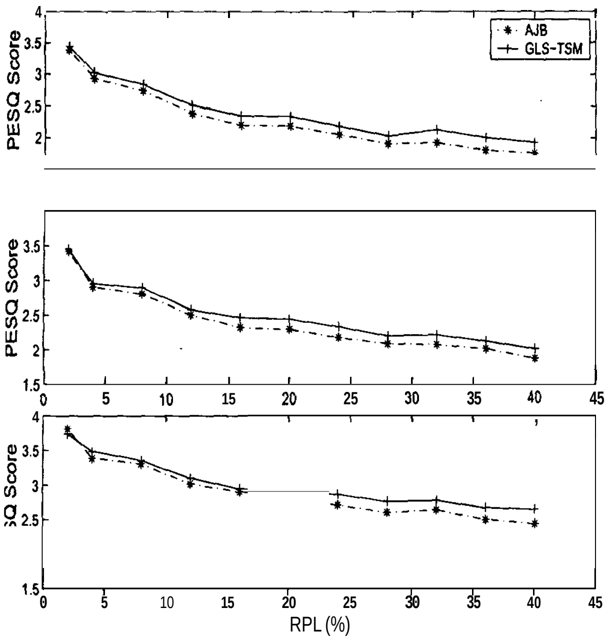

Fig. 3. MOS Predictions for **3** representative input-files with random packet-loss percentages ***(No*** packet arrival jitter)

formance differences were at least as much as shown by the results of objective tests **or** better.

5. CONCLUSIONS

Packet arrival-delay variations and losses seriously affect the quality **of** voice delivered in **VoIP** systems. In this work, **a** low-complexity scheme based on a time-scale modification algorithm has been proposed to handle these impairments in an integrated manner. By performing TSM on the packets already present in the playout buffer, the proposed scheme provides flexible acceptable delay cut-offs to late arriving packets. This reduces the packet loss rate at receiver without introducing additional buffering delays. Further, effective packet-loss concealment is also provided.

Rigorous objective and subjective speech quality tests, for large number **of** input speech samples and widely varying network conditions, were conducted. These tests have confirmed better performance of the proposed scheme.

Being generic and computationally efficient, this full receiver based scheme is suitable for any practical **VoIP** system.

6. REFERENCES

[1] *G.* Carle and E. W. Biersack, “Survey of Error Recovery Techniques for IP-based Audio-visual Multicast Applications,” ***IEEE Network,*** vol. 11, no. 6, pp. 24–36, November-December 1997.

[2] C. Perkins, O. Hodson, and V. Hardman, “A Survey of Packet Loss Recovery Techniques for Streaming Audio,” ***IEEE Network,*** vol. 12, no. 5, pp. 40-48, September-October 1998.

[3] R. Ramjee, J. Kurose, D. Towsley, and H. Schulzrinne, “Adaptive Playout Mechanisms for Packetized Audio Applications in Wide-Area Networks,” in ***Proc IEEE INFOCOM,*** June 1994, vol. 2, pp. 680–88.

[4] S. B. Moon, J. Kurose, and D. Towsley, “Packet Audio Playout Delay Adjustment : Performance Bounds and Algorithms,” Tech. Rep., Dept. of Comp. Sc., Univ. of Mass, Amherst, August 1995.

[5] P. DeLeon and C. J. Sreenan, “An Adaptive Predictor for Media Playout Buffering,” in ***IEEE Conf on Acoustics, Speech and Signal Processing (ICASSP),*** 1999, vol. 6, pp. 3097–3100.

[6] M. M. Rama Kumar, “Adaptive Playout Techniques for Packet Voice in the Internet without Resource Reservation,” M.S. thesis, Department of ECE, Indian Institute of Science, Bangalore, India, January 2000.

[7] C. J. Sreenan, J.-C. Chen, P. Agrawal, and B. Narendran, “Delay Reduction Techniques for Playout Buffering,” ***IEEE Trans. Multimedia,*** vol. 2, no. 2, pp. 88–100, June 2000.

[8] S. Roucos and A. M. Wilgus, “High Quality Time Scale Modification for Speech,” in ***IEEE Conf on Acoustics, Speech and Signal Processing (ICASSP),*** March 1985, pp. 493–496.

[9] D. J. Hejna, Jr., “Real-Time Time-Scale Modification of Speech via the Synchronized Overlap-Add Algorithm,” M.S. thesis, MIT, 1990.

[10] S. **Yim** and **B. I.** Pawate, “Computationally Efficient **Algo**rithm for Time Scale Modification (GLS-TSM),” in ***IEEE Conf on Acoustics, Speech and Signal Processing (ICASSP),*** 1996, vol. 2, pp. 1009–12.

[11] **H.** Sanneck, A. Stenger, K. B. Younes, and B. Girod, “A New Technique for Audio Packet Loss Concealment,” in ***Proc. IEEE GLOBECOM,*** 1996, pp. 48–52.

[12] A. Stenger, K. B. Younes, R. Reng, and B. Girod, “A New Error Concealment Technique for Audio Transmission with Packet Loss,” in ***Proc. Euro. Sig. Processing Conf. (EUSIPCO),*** Trieste, Italy, September 1996.

[13] ITU, ***ITU-T Rec. P.800 :Methodsfor Subjective Determination of Transmission Quality,*** August 1996.

[14] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications,” Request for Comments (RFC) 1889 (proposed standard), Internet Engineering Task Force, January 1996.

[15] ITU, ***ITU-T Rec. P.862 : Perceptual Evaluation of Speech Quality (PESQ), An Objective Methodfor End-to-end Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs,*** February 2001.