A LOW-DELAY SPEECH CODER
HONGMAO SUN, CHUNYAN WANG,
M. OMAIR AHMAD, FELLOW, IEEE, M. N. S. SWAMY, FELLOW, IEEE
Department of Electrical and Computer Engineering
Concordia University
1455 de Maisonneuve Blvd. West
Montréal, Québec
CANADA H3G 1M8

Abstract: - This paper presents an efficient low-delay CELP speech coder based on a structure given by Chen et al. [1]. The proposed coder can operate at a rate of 8 Kb/s and has an arithmetic complexity that is 20% lower than that of the CELP of [1] with an acceptable increase in the delay. The proposed coder has been tested to provide a good-quality speech.

Key words: - Speech compression, low-bit rate coders, CELP coder

1 Introduction

In modern digital communication, the transmission of voice traffic is in packets. Speech coders are the engines for creation and processing of the packets [2]. In evaluating the performance of a speech coder, several factors come into play: 1) quality, 2) bit rate, and 3) complexity. However, for the real time applications, any improvement in these performance measures should be done while keeping the coding delay within an acceptable range [2].

One of the problems in speech compression methods which utilize delayed-decision coders is that the coding gain is achieved at the expense of the coding delay. The one-way delay is basically the time elapsed from the instant a speech sample arrives at the encoder to the instant that this sample appears at the output of the decoder [1]. This definition of one-way delay does not include channel- or modem-related delays. The delay is basically attributed to data buffering, processing, and generation of coding bits (channel symbols). Roughly speaking, the one-way delay is generally between two and four frames. For example, a typical CELP algorithm with 20 ms frames is associated with a delay of about 60 ms [1]. Fast processing and encoding that transmit coding bits as they become available (on the fly) can reduce this delay.

The 16 Kb/s low-delay CELP coder, proposed by Chen et al. [1], was selected by CCITT as a G.728 standard for universal applications in 1991. It achieves a low one-way delay by: a) using a backward-adaptive predictor, and b) short excitation vectors. However, the bit rate is rather high because of the short excitation vector (5 samples) represented by a 10-bit codebook index. At present, these drawbacks limit a wide-spread application of a low-delay CELP.

In this paper, we propose an 8 Kb/s low-complexity low-delay CELP coder based on the structure given in [1] in which the one way delay is less than 5 ms. The frame-size is 2.5 ms, and the sub-frames are 1.25 ms long. Instead of the 50th-order LPC of [1], a 15th-order LPC is used. The perceptual weighting filter is 10th-order, and the log gain predictor is 5th-order. A 10-bit codebook is used to compensate for the error between the input vector and the synthesized vector.

2 Review of A Low-Delay CELP Coder

In this section, we present a brief review of the conventional low-delay CELP algorithm [1]. It is essentially a closed-loop system. In a backward-adaptive prediction, the linear predictor parameters
are determined by operating on the previously quantized speech samples which are also available at the decoder. The low-delay CELP algorithm does not utilize the pitch predictor. Instead, the order of the short-term predictor is increased to fifty to compensate for the lack of a pitch loop. The autocorrelation analysis for the LPC is based on a hybrid window which consists of recursive and non-recursive parts. The frame-size in the low-delay CELP is 2.5 ms and the sub-frames are 1.25 ms long. The parameters of the 50th-order predictor are updated every 2.5 ms. The low-delay CELP uses a gain-shape vector quantizer (VQ) for the excitation. The codebook consists of a 3-bit gain and 7-bit shape codewords. A backward-adaptive excitation gain is also used. The gain information is obtained from the previously quantized excitation using a 10th-order predictor which operates on logarithmic gains. The perceptual weighting filter is based on 10th-order LPC operating directly on the unquantized speech and is updated every 2.5 ms. Finally, the low-delay CELP utilizes adaptive short- and long-term post-filters to emphasize the pitch and formant structures of the speech.

3 proposed 8 Kb/s low-delay CELP CODER

In this section, we propose a low-delay CELP by modifying some of the functional blocks of the low-delay CELP presented in [1]. The objective here is to achieve a low-delay coder that works efficiently at 8 Kb/s and has a lower complexity than the coder given in [1]. In the proposed scheme, the frame-size used is 20 samples and the sub-frames have 10 samples. The 50th-order backward adaptation synthesis LPC of [1] is changed to the one of 15th-order. The 5-sample excitation vector is increased to a 10-sample vector. The modified 10-bit codeword consists of two parts: 3 bits for the gain and 7 bits for the shape. The two parts together produce a codebook with 1024 different excitation vectors. For the codebook search, an enhanced technique is used. A 10th-order perceptual weighting filter and a post-filter is used to improve the speech quality. The block diagrams for the proposed 8 Kb/s coder and decoder are shown in Fig.1. In the following subsections, we will further elaborate on the modifications carried out for the new coder.

3.1 Expansion of the codeword from 5 to 10 samples

While keeping the general structure of the coder given in [1] unchanged, we simply double the 5-sample size (0.625 ms) of the sub-frames in [1] to 10 samples and the corresponding excitation vector also to 10 samples (1.25 ms). With of a 10-bit codebook, we encode the speech signal at a rate of 1 bit/sample. Since the standard sampling rate is 8 kHz, 1 bit/sample corresponds to 8 Kb/s. The 10-bit codebook is generated by using the method given in [4]-[6]. Each of the 128 excitation vectors has 10 samples. Correspondingly, the 15th-order LPC predictor coefficients are updated once every two vectors by performing a backward LPC analysis, and the coefficients of the log-gain predictor is updated once every two vectors. The modification we just mentioned is valid to both the encoder as well as the decoder. The coefficients of the 10th-order perceptual weighting filter and of the post-filter are also updated once every two vectors. The perceptual weighting filter removes the buzzy noise.

3.2 Organization of the codebook

Since the codeword is increased to 10 samples, we can use a power distribution directed book search method which is a kind of a tree search, to make the book search more efficient. First the codebook is organized as it is generated: The shape book has ten samples, It has a power distribution that differs for different codewords. The codebook is divided into three parts having 3, 4 and 3 samples, respectively. The codebook itself is divided into three sections. The peak from one part of the codeword resides in one of the subsections of the codebook. Each time we make a codebook search, it needs to be done only within one subsection. This method thus saves two-third in the search calculation. To compensate for the deterioration of quality introduced by decreasing the order of the backward adaptation synthesis LPC from 50 to 15, we train the codebook with the a carefully selected residual sequences. This method adds more pitch information into the randomly generated codebook. It is well suited to restore the speech quality.
3.3 Reduction in the complexity

The complexity of the 50th-order backward adaptation synthesis LPC is a significant proportion of the overall complexity. This is mainly due to the autocorrelation calculation and the implementation of the Levinson-Durbin algorithm [2]. The complexities of both these calculations are order-dependent. The hybrid window requires $(L+N)*P+1$ multiplications, and $(L+N)*P+1$ additions. where $L$ is the frame size, $N$ is the nonrecursive window size and $P$ is the order of the LPC. According to [2], the Levinson-Durbin algorithm requires $P^2+2P$ multiplications, and $P^2$ additions. Obviously, the complexity increases rapidly as the order increases. Hence, in order to reduce the complexity, a very effective method is to reduce the order of the synthesis LPC both in the encoder and the decoder. The solution, therefore, lies in finding a way to reduce the order of the synthesis LPC.

We notice that the order of the synthesis LPC in the low-delay CELP of [1] has been increased from 10 to 50, to compensate for the loss in the female speech quality caused by eliminating the pitch predictor. In our coder, instead of increasing the order, we put the female pitch information into the shape codebook. Thus, the high order of the synthesis LPC is no longer necessary. We reduce the 50th-order to 15th-order for the LPC. As a result, we save 70% in the calculation of the autocorrelation and 90% in the implementation of the Levinson-Durbin algorithm.

Fig. 1. Proposed 8 Kb/s low-delay CELP coder. (a) Encoder. (b) Decoder.
In order to get a more efficient coder, a new codebook search method is used. It saves 30% in the complexity of codebook search over that of the low-delay CELP of [1]. The overall complexity of the final proposed low-delay CELP is 20% lower than that of the CELP presented in [1].

4 Test Results

To evaluate the performance of the proposed low-delay CELP algorithm, we ran an informal a/b (pairwise) listening test. Six listeners compared the proposed 8 Kb/s CELP coder and the 16 Kb/s CELP coder of [1]. Eight sentences spoken by 4 male and 4 female speakers were used. The test results are shown in Table 1. These subjective results show that the proposed 8 Kb/s CELP coder produces a speech quality very similar to that of the 16 Kb/s CELP of [1].

5 Conclusion

In this paper, we have proposed a low-delay CELP coder that works at 8 Kb/s and gives a good quality speech. Because of the low delay, and the lower bit rate and complexity than that of the low-delay CELP of [1], the proposed coder should suit for a wide-range of applications, such as voice over IP etc.

<table>
<thead>
<tr>
<th></th>
<th>Pref.G.728 (16 Kb/s)</th>
<th>Pref. Modified low delay CELP (6.7 Kb/s)</th>
<th>No Pref.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Female</td>
<td>48%</td>
<td>43</td>
<td>9%</td>
</tr>
<tr>
<td>Male</td>
<td>45%</td>
<td>41</td>
<td>16%</td>
</tr>
<tr>
<td>Overall</td>
<td>46.5%</td>
<td>42%</td>
<td>11.5%</td>
</tr>
</tbody>
</table>

6 Acknowledgments

This work was supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada, Fonds pour la Formation de Chercheureurs de l’Aide à la Recherche (FCAR) of Quebec, and Micronet National Network of Centers of Excellence.

7 References