Low Delay Speech Coding

Vladimir Cuperman

Communications Science Laboratory, School of Engineering Science
Simon Fraser University, Burnaby, B.C., Canada V5A 1S6

Abstract

Good speech quality with low-delay coding at 8-16 kb/s has been obtained using backward adaptive analysis-by-synthesis codecs, such as Low-Delay CELP (future 16 kb/s standard), Low-Delay Vector Excitation Coding (LD-VXC), and backward adaptive tree/trellis codecs. This paper presents design and performance trade-offs for the low-delay analysis-by-synthesis codecs at rates of 8-16 kb/s. A number of approaches for improving the speech quality at 8 kb/s are discussed. Backward pitch prediction configuration is compared to a closed-loop forward configuration similar to that used in the conventional forward CELP for the adaptive code-book. Finally, the robustness to transmission errors is discussed and a number of trade-offs for reducing the sensitivity to transmission errors are presented.

1. Introduction

Traditionally, speech coders have been judged based on three major performance criteria: speech quality, rate, and complexity or cost. Recently, communications delay has become an important criterion for speech encoders used in the public switched telephone network (PSTN). In a complex network, the delays of many encoders add together, transforming the delay into a significant impairment of the system. Delay may necessitate the use of echo cancellation and in some applications remains an impairment even after echo cancellation has been performed. For these reasons, the requirements of the new 16 kb/s and 8 kb/s CCITT standards specify low delay as a major requirement.

The delay performance of a speech coder is characterized by the algorithmic delay and the total codec delay. Algorithmic delay is the one-way delay of the encoder and the decoder assuming infinite processing power for the coder implementation. Processing delay is the additional delay due to implementation with a finite processing power. The total codec delay is the sum of algorithmic delay and processing delay. The channel delay caused by transmitting over a finite bandwidth serial channel adds to the total codec delay to give the delay encountered in practical applications. The requirements of the new 8 kb/s CCITT standard specify an algorithmic delay lower than 5 ms (objective 2 ms) and a total codec delay of 10 ms (objective 5 ms). The 16 kb/s standard will have a total delay lower than 2 ms.

The existing speech codecs based on conventional Code Excited Linear Prediction (CELP) [1] or Vector Excitation Coding (VXC) [2] achieve good speech quality at the expense of a substantial delay due to forward adaptation of the short-term predictor, where input speech samples are buffered to compute synthesis filter parameters prior to actual coding of the samples. The input buffering and other processing typically result in a total codec delay of 50 to 60 ms. Hence, the low-delay requirement is not feasible with the established coders that are based on forward adaptive prediction.

Low delay speech coders are based on backward adaptive linear prediction combined with a delayed decision quantization procedure, such as vector quantization or tree/trellis coding. In a backward adaptive configuration, the parameters of the synthesis filter are not derived from the original speech signal, but instead computed by backward adaptation, extracting information only from the reconstructed signal based on the transmitted excitation information. Since both the encoder and decoder have access to the past reconstructed signal, side information is no longer needed for the synthesis filter, and the low-delay requirement can be met with a suitable choice of vector dimension.

Backward adaptation has been widely studied in conjunction with an earlier generation of speech coding systems [3]. The classical ADPCM algorithm based on scalar quantization and backward adaptive linear prediction achieves toll speech quality at 32 kb/s and has an algorithmic delay of only one sample interval, but the
speech quality degrades quickly for rates under 24 kbps. Good speech quality was obtained at 16 kbps by combining block backward adaptive linear prediction with delayed decision quantization based on tree [5, 13] or trellis [12] search. A system based on recursive backward adaptation and analysis-by-synthesis codebook excitation quantization called Vector ADPCM was introduced in [6]. Vector ADPCM, which appears to be the first application of backward adaptation to analysis-by-synthesis excitation coding, evolved into Low-Delay Vector Excitation Coding (LD-VXC) [7]. Lattice LD-VXC (LLD-VXC) achieved toll quality at 16 kbps with a delay lower than 2 ms [8]. Finally, the emerging 16 kbps speech coding standard is based on Low-Delay Code Excited Linear Prediction (LD-CELP) which uses codebook based analysis-by-synthesis excitation coding and backward block adaptation of a 50th order short-term predictor; LD-CELP achieves toll quality with a delay lower than 2 ms and good robustness in the presence of transmission errors [9].

Recently, a number of 8 kbps coders which achieve close to toll speech quality with low delay have been proposed. These include versions of LD-CELP and LD-VXC which use closed-loop quantization of the pitch predictor parameters [11, 9, 17, 18], and a tree encoder based on the fractional rate trees [14].

2. System Overview

The block diagram of an analysis-by-synthesis low-delay speech coder is shown in Fig. 1. At the encoder side, an analysis-by-synthesis technique is used for selecting the optimum excitation sequence. The excitation generator structure may be codebook based as in LD-CELP and LD-VXC, or may consist of a tree (trellis) search module and the corresponding excitation sequence generator. The term codevector will be used for an excitation codebook entry or for an excitation sequence corresponding to a given tree path.

Codebook based analysis-by-synthesis systems use gain-shape or shape-only codebooks. A gain-shape codebook generates the excitation vectors by multiplying vectors in a shape sub-codebook by gain values stored in a gain sub-codebook. In a shape-only codebook the gain sub-codebook has only one entry.

In the diagram shown in Fig. 1, each candidate excitation codevector \( c(n) \) is multiplied by a gain, and the resulting gain scaled vector, \( u(n) \), is fed into the synthesis filter. The gain is the product of the predicted gain obtained from the backward adaptive gain predictor and the gain value obtained from the gain codebook. The components of the vector \( u \) will be denoted by \( u(n) \). The output of the synthesis filter, \( y \), is then compared to the actual speech signal, \( x \), and the best candidate codevector (path) is selected using a perceptually weighted minimum square error criterion based on the weighting filter, \( W(z) \).

The index of the optimal excitation sequence is then transmitted to the decoder. At the decoder side, the received indices are used to generate the proper excitation sequence. The excitation codevector is then gain scaled using the gain computed in the same way as it is done in the encoder, and fed into the cascade of the pitch and formant synthesis filters. The output of the cascade of synthesis filters is the reconstructed speech which may be postfiltered to reduce the perceived quantization noise.

3. Gain Predictor

Low delay speech coders use backward adaptive linear prediction for excitation gain evaluation. A simple and robust predictor may be obtained by generalizing the Jayant’s robust scalar quantizer [10, 14]. Denoting by \( \sigma(n) \) the gain at the vector time index \( n \), the corresponding adaptation equation can be written at logarithmic scale

\[
\log \sigma(n) = \beta \log \sigma(n-1) + \log M(n-1)
\]

where \( \beta \) is fixed predictor coefficient, \( |\beta| < 1 \), and \( M(n-1) \) is a function of the RMS value of the vector \( u(n-1) \).

A further generalization may be obtained by using an adaptive predictor of order \( M_p \) [10, 7]

\[
M_p = \sum_{i=1}^{M_p} p_i \log |u(n-i)|
\]

where the predictor coefficients \( p_i \) may be computed by applying backward LPC analysis (autocorrelation...
method) to the sequence of RMS values of the vectors \( u(n-i) \). Alternatively, fixed coefficients \( \rho_i \) optimized on a multi-talker data base may be used [7]. The fixed predictors are more robust in the presence of transmission errors but have slightly lower performance than the adaptive predictors for clean channel conditions.

4. The Short-Term Predictor

The output of the short term predictor, \( y(n) \), is computed using the relationship:

\[
y(n) = w(n) + \sum_{i=1}^{P} h_i y(n-i) + \sum_{i=1}^{Z} g_i w(n-i)
\]  

(3)

where \( \{g_i\} \) are the coefficients of the all-zero section, and \( \{h_i\} \) are the coefficients of the all-pole section.

In block adaptive systems, an all-pole short term predictor is traditionally used \((Z = 0)\). In this case the coefficients \( h_i \) can be computed by solving the Wiener-Hopf equations

\[
R_{yy} h = r_y
\]  

(4)

where \( h = (h_1, h_2, ..., h_P)^T \), \( R_{yy} \) is the autocorrelation matrix of \( y(n) \), and \( r_y = (r_{yy}(1), r_{yy}(2), ..., r_{yy}(P))^T \).

The recursive adaptation is based on the application of the gradient algorithm for solving the prediction error minimization problem. The corresponding adaptation equations are

\[
h_i^{(\text{new})} = \lambda h_i^{(\text{old})} + \mu_y u(n)y(n-i) \quad \text{for} \quad i = 1, ..., P
\]  

(5)

\[
g_i^{(\text{new})} = \lambda g_i^{(\text{old})} + \mu_x w(n-i)w(n-i) \quad \text{for} \quad i = 1, ..., Z
\]  

(6)

where \( \lambda_y, \lambda_x \) are leakage factors for improving noisy channel performance and \( \mu_y, \mu_x \) are adaptation stepsizes. Simple analytical stability checks exist for coefficients \( h_i \) for \( P \leq 3 \). For predictors of order larger than three, the simplest solution is to use the equivalent lattice filter stability check.

![Fig. 2 Lattice Short-Term Predictor](image)

Lattice filters have significant advantages in the implementation of linear predictors. An all-pole lattice filter with recursive adaptation is shown in Fig. 2. The update of the coefficients \( k_j^{(\text{new})} \) may be done using the LMS algorithm with leakage factors for improving noisy channel performance. However, a significant reduction in the computational complexity may be obtained by using the so-called sign algorithm [8] which avoids the division operations required by the standard LMS algorithm:

\[
k_j^{(\text{new})} = (1-\mu)k_j^{(\text{old})} + \lambda \text{sign}(e_j(n))\text{sign}(r_j(n-1))
\]  

(7)

where \( 1 \leq j \leq M \). A good choice for the constants is \( \lambda = \mu = 2^{-4} \).

In the absence of a pitch predictor, speech quality in low delay coders depends significantly on the short-term predictor order. High order short-term predictors were introduced in LD-CELP where a 50th order block adaptive short-term predictor is used [10]. At 16 kb/s, in the absence of a pitch predictor, the prediction gain saturates for a 20th order predictor for male speakers, but only for a 50th order predictor for female speakers [10]. On the other hand, in the presence of a pitch predictor, the prediction gain saturates at 16 kb/s for a 20th order predictor for male and female speakers [8]. At 8 kb/s there is no performance improvement for predictors of order larger than 10 [11, 9, 18]. The poor performance of high-order predictors at 8 kb/s may be caused by the quantization noise present in the adaptation loop.

The coder robustness in the presence of transmission errors depends significantly on the choice of adaptation signals. Replacing the signal \( w(n-i) \) by \( u(n-i) \) in (6) (parallel adaptation [7, 15]) results in improved robustness at the expense of a slight performance degradation. The improvement in robustness is due to reduced error propagation. Further reduction of error propagation may be obtained by replacing the signal \( y(n-i) \) in (5) by \( u(n-i) \), i.e., introducing the residual driven adaptation [3]. However, in this case the performance for clean channels degrades significantly. A good trade-off may be obtained by shaping the residual \( u(n) \) by an FIR approximation of the short-term synthesis filter (smoothed error signal adaptation, [14]).

5. The Pitch Predictor

The pitch predictor equation for a three tap predictor is given by

\[
w(n) = u(n) + \sum_{i=1}^{1} a_i w(n-k_p-i)
\]  

(8)

where \( a_i \) are the filter coefficients and \( k_p \) is the current pitch period estimate. Block and/or recursive adaptation may be used for the predictor coefficients \( a_i \).
In block adaptation, the pitch prediction coefficients, $a_i$, may be calculated from the previous frame of predictor output, $w(n)$, using the autocorrelation [7, 15, 16] or the covariance methods [4, 5, 14].

A robust low-complexity pitch predictor using backward recursive adaptation and a pitch tracker was introduced in [15, 16]. Recursive adaptation allows better tracking of signal characteristics in backward adaptive pitch prediction [15]. The possibility of achieving robustness in the presence of transmission errors with backward recursive pitch prediction was confirmed by the results obtained at 8 kb/s in [14], despite claims that such a predictor may not be robust enough [10].

6. Excitation Sequence Optimization

The excitation sequence is obtained by minimizing the weighted mean-square error

$$d_w(x,y) = \frac{1}{M} \sum_{j=0}^{M-1} e_w^2(j)$$

where $e_w(n)$ is generated by passing $e(n) = x(n) - y(n)$ through the weighting filter $W(z)$. For codebook based systems like LD-CELP and LD-VXC, the WMSE is computed for each entry in the excitation codebook and $M$ is the vector dimension. Codebook design algorithms are discussed in [7, 8]. For tree coding, all possible path map sequences to depth $M$ are considered for computing the reconstructed speech $y(n)$ and the path giving the minimum distortion is chosen. The index $j$ refers in this case to "future" instants into the tree, and after selecting the optimum path only the index value at time $n_0$ is released. A suboptimal search algorithm is employed generally for tree encoding, while codebook encoding uses exhaustive (optimal) search.

7. Weighting Filter

The choice of the excitation sequence is based on a perceptually weighted minimum mean-square error criterion defined by the weighting filter, $W(z)$. Perceptual weighting of the error signal is used to shape the noise spectrum in order to reduce the level of perceived noise.

To derive the weighting filter parameters, the input speech signal is fed into a $M_z$-order predictor and the coefficients are adapted by the algorithms described in Section 4. The transfer function of the weighting filter is given in terms of the optimal linear predictor for clean speech, $W_1(z)$, as $W(z) = W_1(z/\gamma_1)/W_1(z/\gamma_2)$, where $0<\gamma_2<\gamma_1<1$. Suitable values for $\gamma_1$ and $\gamma_2$ are 0.9 and 0.4, respectively. The fact that the weighting filter adapts at the same time instants as the short-term predictor and uses an identical adaptation structure was found to have a significant impact on the perceived speech quality. Lower complexity at the expense of a slightly degraded performance may be obtained by using the synthesis filter parameters for designing the weighting filter.

8. Experimental results

In order to study the advantages of closed-loop pitch prediction and the effect of vector dimension upon the performance at 8 kb/s, the following LLD-VXC systems were tested [18]:

- **System 1** - open-loop pitch predictor, vector dimension 8, 8 bits shape codebook
- **System 2** - open-loop pitch predictor, vector dimension 10, 8 bits shape and 2 bits gain codebooks
- **System 3** - closed-loop pitch predictor, vector dimension 18, 8 bits shape codebook, 6 bits adaptive codebook taps, 4 bits pitch for delta pitch encoding
- **System 4** - closed-loop pitch predictor, vector dimension 20, 8 bits shape codebook, 7 bits adaptive codebook taps, 4 bits pitch

All the above systems use a 10th order lattice short-term predictor and a fixed 10th order gain predictor. The signal-to-noise ratio and informal MOS test results for the above systems are given in Table 1.
TABLE I
Performance of 8 kb/s Low Delay Speech Coders

<table>
<thead>
<tr>
<th>System</th>
<th>MOS</th>
<th>SNR</th>
<th>SEGSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>3.73</td>
<td>12.1</td>
<td>12.5</td>
</tr>
<tr>
<td>2</td>
<td>3.62</td>
<td>13.4</td>
<td>13.9</td>
</tr>
<tr>
<td>3</td>
<td>3.93</td>
<td>14.1</td>
<td>14.5</td>
</tr>
<tr>
<td>4</td>
<td>3.85</td>
<td>14.2</td>
<td>14.5</td>
</tr>
</tbody>
</table>

The results in Table 1 indicate that the systems using closed-loop pitch prediction achieve subjective speech quality comparable to the 8 kb/s VSELP standard. The systems employing open-loop pitch prediction have a MOS score lower by about 0.2. The results show that the SNR values do not always correlate well with the subjective speech quality estimates. LLD-VXC at 16 kb/s achieves a SEGSNR of about 19 dB and a MOS close to 4.1.

References


