Speech Coding at Low bit Rates

Shiva Sai(202SP023)
shivasaisamboju.202sp023@nitk.edu.in

Abstract

Speech coding deals with the problem of reducing the bit rate required for representing speech signals while preserving the quality of the speech reconstructed from that representation. Speech coding or compression deals with the problem of obtaining compact representation of speech signals for efficient digital storage or transmission and in reducing the bit rate required for a speech representation while preserving the quality of speech reconstructed from that representation. Hence, the main objective of speech coding techniques is to represent speech signal with minimum number of bits while maintaining its quality.

1. Introduction

Speech coding has been and still is a major issue in the area of digital speech processing. Speech coding is the act of transforming the speech signal at hand, to a more compact form, which can then be transmitted with a considerably smaller memory. The motivation behind this is the fact that access to unlimited amount of bandwidth is not possible. Therefore, there is a need to code and compress speech signals. Speech compression is required in long-distance communication, high-quality speech storage, and message encryption. For example, in digital cellular technology many users need to share the same frequency bandwidth. Utilizing speech compression makes it possible for more users to share the available system. Another example where speech compression is needed is in digital voice storage. For a fixed amount of available memory, compression makes it possible to store longer messages [1].

2. Background

The speech coding in this project will be accomplished by using a modified version of LPC-10 technique. Linear Predictive Coding is one possible technique of analyzing and synthesizing human speech.Only an overview will be included in this section, along with the previously mentioned other types of coding techniques.

LPC method has been used for a long time. Texas Instruments had developed a monolithic PMOS speech synthesizer integrated circuit as early as 1978. This marked the first time the human vocal tract had been electronically duplicated on a single chip of silicon [3]. This one of the first speech synthesizer used LPC to accomplish successful synthesis. LPC makes coding at low bit rates possible. For LPC-10, the bit rate is about 2.4 kbps. Even though this method results in an artificial sounding speech, it is intelligible. This method has found extensive use in military applications, where a high quality speech is not as important as a low bit rate to allow for heavy encryption of secret data. However, since a high quality sounding speech is required in the commercial market, engineers are faced with using other techniques that normally use higher bit rates and result in higher quality output. In LPC-10 vocal tract is represented as a time-varying filter and speech is windowed about every 20 ms. For each frame, the gain and only 10 of the coefficients of a linear prediction filter are coded for analysis and decoded for synthesis. In 1996, LPC-10 was replaced by mixed-excitation linear prediction (MELP) coder to be the United States Federal Standard for coding at 2.4 kbps. This MELP coder is an improvement to the LPC method, with some additional features that have mixed excitation, aperiodic pulses, adaptive spectral enhancement and pulse dispersion filtering as mentioned in [2].

Waveform coders on the other hand, are concerned with the production of a reconstructed signal whose waveform is as close as possible to the original signal, without any information about how the signal to be coded was generated. Therefore, in theory, this type of coders should be input signal independent and work for both speech and non-speech input signals [2]. Waveform coders produce a good quality of speech signals above bit rates of 16 kbps. However, if the bit rate is decreased below 16 kpbs, the quality deteriorates quickly. One form of waveform coding is Pulse Code Modulation (PCM). This type of waveform coding involves sampling and quantizing the input signal. PCM is a memoryless coding algorithm as mentioned in [2]. Another type of PCM is Differential Pulse Code Modulation (DPCM). This method quantizes the difference between the original and the predicted signals. This method involves prediction of the next sample from the previous samples. This is possible since there is a correlation in speech samples because of the effects of the vocal tract and the vibrations in the vocal cords [4]. It is possible to improve the predictor as well as the quantizer in DPCM if they are made adaptive, in order to match the characteristics of the speech that is to be coded. This type of coders is called Adaptive Differential Pulse Code Modulation (ADPCM).

One other type of speech coders is called the Subband coders. This type of coding involves filter bank analysis to be undertaken in order to filter the input signal into several frequency bands. Bit allocation is done to each band by a certain criterion [2]. Presently however, Subband coders are not widely used for speech coding. It is very difficult to create high quality speech by using low bit rates with this technique. As suggested in [2], Subband coding is mostly utilized in the medium to high bit rate applications of speech coding.

3. Project Description

3.1 Linear Predictive Analysis

Linear prediction (LP) is one of the most important tools in speech analysis. The philosophy behind linear prediction is that a speech sample can be approximated as a linear combination of past samples. Then, by minimizing the sum of the squared differences between the actual speech samples and the linearly predicted ones over a finite interval, a unique set of predictor coefficients can be determined [5]. LP analysis decomposes the speech into two highly independent components, the vocal tract parameters (LP coefficients) and the glottal excitation (LP residual). It is assumed that speech is produced by exciting a linear time-varying filter (the vocal tract) by random noise for unvoiced speech segments, or a train of pulses for voiced speech. Figure 1 shows a model of speech production for LP analysis [6]. It consists of a time varying filter H(z) which is excited by either a quasi periodic or a random noise source. The most general predictor form in linear prediction is the autoregressive moving average (ARMA) model where the speech sample s(n) is modelled as a linear combination of the past outputs and the present and past inputs [7–9]. It can be written mathematically as follows

$$s(n) = -\sum_{k=1}^{p} a_k s(n-k) + G\sum_{l=0}^{q} b_l u(n-l) ....(1)$$

where a_k, $1 \leq k \leq p$, b_l , $1 \leq l \leq q$ and gain G are the parameters of the filter. Equivalently, in frequency domain, the transfer function of the linear prediction speech model is [7]

$$H(z) = \frac{1+\sum_{l=1}^{q} b_l z^{-l}}{1+\sum_{k=1}^{p} a_k z^{-k}} ......(2)$$

H(z) is referred to as a pole-zero model. The zeros represent the nasals and the poles represent the resonances (formants) of the vocal-tract. When a k = 0 for $1 \leq k \leq p$. H(z) becomes an all-zero or moving average (MA) model. Conversely, when b_l =0 for $1 \leq l \leq q$, H(z) becomes an all-pole or autoregressive (AR) model [8].For non-nasal voiced speech sounds the transfer function of the vocal-tract has no zeros whereas the nasals and unvoiced sounds usually includes the poles (resonances) as well as zeros (anti resonances) [6].

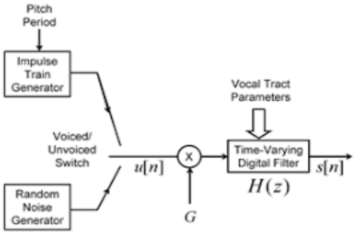

Figure 1. Model of speech production for LP analysis

Generally the all-pole model is preferred for most applications because it is computationally more efficient and its the acoustic tube model for speech production.It can model sounds such as vowels well enough. The zeros arise only in nasalsand in unvoiced sounds like fricatives. These zeros are approximately modelled by including more poles [6]. In addition, the location of a poles considerably more important perceptually than the location of a zero [10]. Moreover, it is easy to solve an all-pole model. To solve a pole-zero model, it is necessary to solve a set of nonlinear equations, but in the case of an all-pole model, only a set of linear equations need to be solved. The transfer function of the all-pole model is [8]

$$H(z)=\frac{1}{1+\sum_{k=1}^{p} a_k z^{-k}}......(3)$$

The number p implies that the past p output samples are being considered, which is also the order of the linear prediction. With this transfer function, we get a difference equation for synthesizing the speech samples s(n) as

$$s[n]=-\sum_{k=1}^{p} a_k s(n-k)+Gu[n]......(4)$$

where the coefficients a_k's are known as linear predictive coefficients (LPCs) and p is the order of the LP filter.

The error signal or the residual signal e(n) is the difference between the input speech and the estimated speech [8].

$$e(n)=s(n)+\sum_{k=1}^{p} a_k s(n-k).........(5)$$

The LP residual represents the excitations for production of speech [11]. The residual is typically a series of pulses, when derived from voiced speech or noise-like, when derived from unvoiced speech. The whole LP model can be decomposed into the two parts, the analysis part and the synthesis part as shown in Figure 2. The LP analysis filter removes the formant structure of the speech signal and leaves a lower energy output prediction error which is often called the LP residual or excitation signal. The synthesis part takes the error signal as an input [8]. The input is filtered by the synthesis filter 1/A(z), and the output is the speech signal.

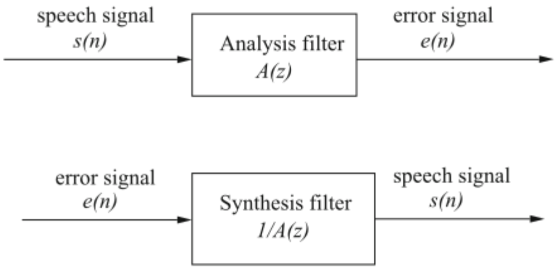

Figure 2. LP analysis and Synthesis Model

There are two widely used methods for estimating the LP coefficients (LPCs) Autocorrelation and Covariance. Both methods choose the short term filter coefficients (LPCs) a k in such a way that the energy in the error signal (residual) is minimized. For speech processing tasks, the autocorrelation method is almost exclusively used because of its computational efficiency and inherent stability whereas the covariance method does not guarantee the stability of the all-pole LP synthesis filter [6, 12].

First, speech signal s(n) is multiplied by a window w(n) to get the windowed speech segment s w (n). Normally, a Hamming or Hanning window is used. The windowed speech signal is expressed as

$$s_w(n)=s(n)w(n).....(6)$$

The next step is to minimize the energy in the residual signal. The residual energy E $_p$ is defined as [165]

$$E_p=\sum_{n=-\infty}^{\infty} e^2(n)=\sum_{n=-\infty}^{\infty}\left(s_w(n)+\sum_{k=1}^{p} a_k s_w(n-k)\right)^2....(7)$$

The values of a $_k$ that minimize E_p are found by setting the partial derivatives of the energy E_p with respect to the LPC parameters equal to 0.

$$\frac{\partial E_p}{\partial a_k}=0,1\le k\le p$$

This results in the following p linear equations for the p unknown parameters $a_1, \ldots, a_p$

$$\sum_{k=1}^{p} a_k \sum_{n=-\infty}^{n=\infty} s_w(n-i)s_w(n-k)=\sum_{n=-\infty}^{n=\infty} s_w(n-i)s_w(n)....(8)$$

$$1\le i\le p$$

This linear equations can be expressed in terms of the autocorrelation function. This is because the autocorrelation function of the windowed segment s_w (n) is defined as

$$R_s(i)=\sum_{n=-\infty}^{\infty} s_w(n)s_w(n+i),1\le i\le p....(9)$$

Exploiting the fact that the autocorrelation function is an even function i.e., Rs (i) =Rs (–i). By substituting the values from Eq. (9) in Eq. (8), we get

$$\sum_{k=1}^{p} R_s(|i-k|)a_k=-R_s(i),1\le i\le p$$

These set of p linear equations can be represented in the following matrix form as [6]

$$\begin{pmatrix} R_s(0) & R_s(1) & ..R_s(p-1) \\ R_s(1) & R_s(0) & ..R_s(p-2) \\ . & . & . \\ . & . & . \\ R_s(p-1) & R_s(p-2) & R_s(0) \end{pmatrix} \begin{vmatrix} a_1 \\ a_2 \\ a_3 \\ . \\ . \\ . \\ a_p \end{vmatrix} = \begin{vmatrix} R_s(1) \\ R_s(2) \\ R_s(3) \\ . \\ . \\ . \\ R_s(p) \end{vmatrix}$$

This can be summarized using vector-matrix notation as:

$$\boldsymbol{R}_s a = -r_s$$

where the **p×p** matrix **Rs** is known as the autocorrelation matrix.The resulting matrix is a Toeplitz matrix where all elements along a given diagonal are equal. This allows the linear equations to be solved by the Levinson-Durbin algorithm. Because of the Toeplitz structure of Rs , A(z) is minimum phase [6]. At the synthesis filter **H(z) = 1/A(z)**, the zeros of **A(z)** become the poles of H(z). Thus, the minimum phase of **A(z)** guarantees the stability of **H(z)**.

3.2 Voicing Detector

The purpose of the voicing detector is to classify a given frame as voiced or unvoiced. In many instances, voiced/unvoiced classification can easily be accomplished by observing the waveform: a frame with clear periodicity is designated as voiced, and a frame with noise-like appearance is labeled as unvoiced. In other instances, however, the boundary between voiced and unvoiced is unclear; this happens for transition frames, where the signal goes from voiced to unvoiced or viceversa. For reliable operation, the detector must take into account as many parameters as possible so as to achieve a high degree of robustness.

These parameters are input to a linear classifier having binary output. The voicing detector is one of the most critical components of the LPC coder, since misclassification of voicing states can have disastrous consequences on the quality of the syntheticspeech.Typically, voiced sounds are several order of magnitude higher in energy than unvoiced signals. For the frame (of length N) ending at instant m, the energy is given by

$$E[n] = \sum_{n=m-N+1}^{m} s^2[n]$$

For simplicity, the magnitude sum function defined by serves a similar purpose.

$$MSF[m] = \sum_{n=m-N+1}^{m} |s[n]|$$

Since voiced speech has energy concentrated in the low-frequency region, due to the relatively low value of the pitch frequency, better discrimination can be obtained by lowpass filtering the speech signal prior to energy calculation. That is, only energy of low-frequency components is taken into account. A bandwidth of 800 Hz is adequate for the purpose since the highest pitch frequency is around 500 Hz.

The zero crossing rate of the frame ending at time instant m is defined by

$$ZCR = \frac{1}{2} \sum_{n=m-N+1}^{m} |sgn(s[n]) - sgn(s[n-1])| \ldots.(10)$$

with sgn( ) the sign function returning 1 depending on the sign of the operand. Equation 10 computes the zero crossing rate by checking the samples in pairs to determine where the zero crossings occur. Note that a zero crossing is said to occur if successive samples have different signs. For voiced speech, the zero crossing rate is relatively low due to the presence of the pitch frequency component (of low-frequency nature), whereas for unvoiced speech, the zero crossing rate is high due to the noise-like appearance of the signal with a large portion of energy located in the high-frequency region.

3.3 Estimation of Pitch Period

One of the most important parameters in speech analysis, synthesis, and coding applications is the fundamental frequency, or pitch, of voiced speech. Pitch frequency is directly related to the speaker and sets the unique characteristic of a person. Voicing is generated when the airflow from the lungs is periodically interrupted by movements of the vocal cords. The time between successive vocal cord openings is called the fundamental period, or pitch period. Pitch period must be estimated at every frame. By comparing a frame with past samples, it is possible to identify the period in which the signal repeats itself, resulting in an estimate of the actual pitch period.

Many techniques have been proposed for the estimation of pitch period.In this pitch Period is calculated using AutoCorrelation Method.

Assume we want to perform the estimation on the signal s[n], with n being the time index. We consider the frame that ends at time instant m, where the length of the frame is equal to N . Then the autocorrelation value

$$R[l,m] = \sum_{n=m-N+1}^{m} s[n]s[n-l]$$

reflects the similarity between the frame s[n], n=m - N + 1 to m, with respect to the time-shifted version s[n - l], where l is a positive integer representing a time lag.By calculating the autocorrelation values for the entire range of lag, it is possible to find the value of lag associated with the highest autocorrelation representing the pitch period estimate, since, in theory, autocorrelation is maximized when the lag is equal to the pitch period.

3.4 Preemphasis

The typical spectral envelope of the speech signal has a high frequency roll-off due to radiation effects of the sound from the lips. Hence, high-frequency components have relatively low amplitude, which increases the dynamic range of the speech spectrum. As a result, LP analysis requires high computational precision to capture the features at the high end of the spectrum. More importantly, when these features are very small, the correlation matrix can become ill-conditioned and even singular, leading to computational problems. One simple solution is to process the speech signal using the filter with system function which is high pass in nature.

By pre-emphasizing, the dynamic range of the power spectrum is reduced. This process substantially reduces numerical problems during LP analysis, especially for low-precision devices. A value of a near 0.9 is usually selected.To keep a similar spectral shape for the synthetic speech, it is filtered by the de-emphasis filter with system function at the decoder side, which is the inverse filter with respect to pre-emphasis.

4. Implementation Details

The structure of a speech coding algorithm that utilizes the LPC model of speech production is presented.

Figure 3 shows the block diagram of the encoder. The input speech is first segmented into nonoverlapping frames. A pre-emphasis filter is used to adjust the spectrum of the input signal.

The voicing detector, discussed in the next section, classifies the current frame as voiced or unvoiced and outputs one bit indicating the voicing state. The pre-emphasized signal is used for LP analysis, where ten LPCs are derived.These coefficients are quantized with the indices transmitted as information of the frame. The quantized LPCs are used to build the prediction-error filter, which filters the pre-emphasized speech to obtain the prediction-error signal at its output.Pitch period is estimated from the prediction-error signal if and only if the frame is voiced. By using the prediction-error signal as input to the pitch period estimation algorithm,a more accurate estimate can be obtained since the formant structure (spectrum envelope) due to the vocal tract is removed.

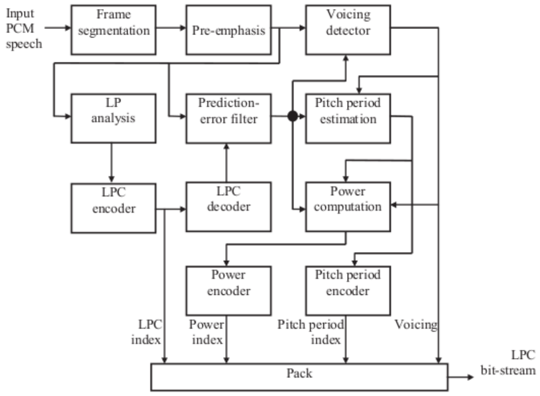

Figure 3. Encoder

Power of the prediction-error sequence is calculated next, which is different for voiced and unvoiced frames. Denoting the prediction-error sequence as e[n] , $n \in [0, N-1]$ with N being the length of the frame, we have for the unvoiced case

$$P = \frac{1}{N} \sum_{n=0}^{N-1} e^2[n]$$

For the voiced case, power is calculated using an integer number of pitch periods:

$$P = \frac{1}{\lfloor N/T \rfloor T} \sum_{n=0}^{\lfloor N/T \rfloor T-1} e^2[n]$$

with $\lfloor . \rfloor$ denotes the floor function (returns the greatest integer less than or equal to the operand). It is assumed that N > T, and hence use of the floor function ensures that the summation is always performed within the frame's boundaries.

Figure 4 shows the block diagram of the decoder and is essentially the LPC model of speech production with parameters controlled by the bit-stream. It is assumed that the output of the impulse train generator is comprised of a series of unit-amplitude impulses, while the white noise generator has unit-variance output. Gain computation is performed as follows. For the unvoiced case, the power of the synthesis filter's input must be the same as the prediction error on the encoder side. Denoting the gain by g, we have

$$g = \sqrt{P}$$

since the white noise generator has unit-variance output.

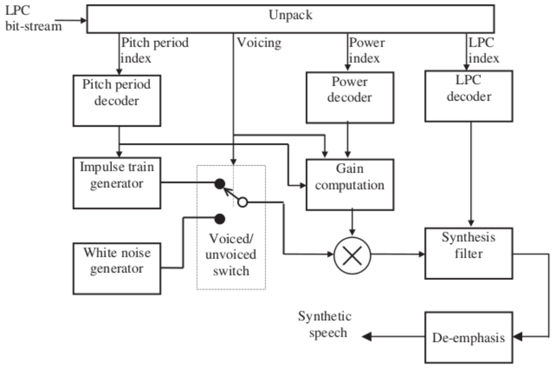

Figure 4. Decoder

For the voiced case, the power of the impulse train having an amplitude of g and a period of T, measured over an interval of length $\lfloor N/T \rfloor T$, must equal p. Carrying out the operation yields

$$g=\sqrt{Tp}$$

Finally, the output of the synthesis filter is de-emphasized to yield the synthetic speech.

5. Results

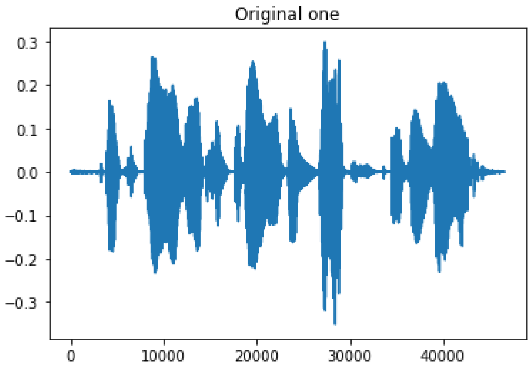

Figure 5. Original Signal

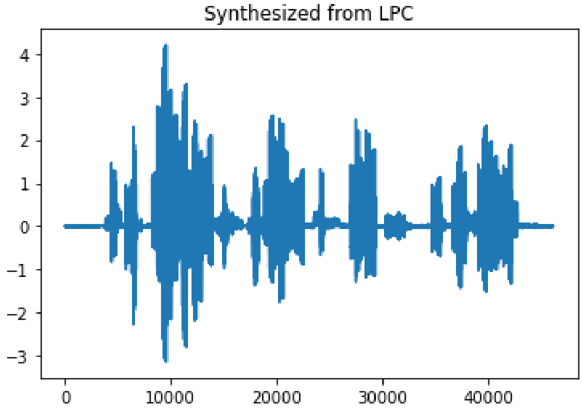

Figure 6. Reconstructed Signal from LPC

6. Conclusion

The overly simplistic model that the LPC coder relies on has relatively low computational cost and makes the low bit-rate speech coder a practical reality. The uncomplicated model, however, is also highly inaccurate in various circumstances,creating annoying artifacts in the synthetic speech.This class of coders works well for low bit-rate. Increasing the bit-rate normally does not translate into better quality, since it is restricted by the chosen model. Typical bit-rate is in the range of 2 to 5 kbps.

Some of the modern coders such as CELP ,MELP provide good rate of compression with efficient decision on classifying voiced and unvoiced signals.

7. References

[1] http://www.data-compression.com/speech.html

[2] M. H. Johnson and A. Alwan, "Speech Coding: Fundamentals and Applications", to appear as a chapter in the Encyclopedia of Telecommunications, Wiley, December 2002.

[3] http://www.ti.com/corp/docs

[4] http://www-mobile.ecs.soton.ac.uk

[5] *L. R. Rabiner and B. H. Juang, Fundamentals of Speech Recognition. Englewood Cliffs, New Jersy: Prentice-Hall, 1993*.

[6] L. R. Rabiner, Digital Signal Processing. IEEE Press, 1972.

[7] B. S. Atal and S. L. Hanauer, “Speech analysis and synthesis by linear prediction of the speech wave,”J. Acoust. Soc. Am., vol. 50, pp. 637–655, Aug. 1971.

[8] J. Makhoul, “Linear prediction: A tutorial review,” Proc. IEEE, vol. 63, pp. 561–580, Apr.1975.

[9] B. S. Atal and M. R. Schroeder, “Linear prediction analysis of speech based on a pole-zero representation,” J. Acoust. Soc. Am., vol. 64, no. 5, pp. 1310–1318, 1978.

[10] D. O’Shaughnessy, “Linear predictive coding,” IEEE Potentials, vol. 7, pp. 29–32, Feb. 1988.

[11] T. Ananthapadmanabha and B. Yegnanarayana, “Epoch extraction from linear prediction residual for identification of closed glottis interval,” IEEE Trans. Acoust., Speech, SignalProcess., vol. ASSP-27, pp. 309–319, Aug. 1979.

[12] J. Picone, “Signal modeling techniques in speech recognition,” Proc. IEEE, vol. 81,pp.1215–1247, Sep. 1993.