

Audio Codec Using Legendre Functions for Simplified Digital Telephony

SabapathyAnanthi¹, Ananthanarayanan Arun², Sivakannan Subramani³ and Krishnasamy Padmanabhan⁴

ananthipradeep84@gmail.com, sivakannan87@gmail.com

¹Dept. of Network Systems and IT, University of Madras, Chennai, India

²Dept. of ECE, MVJ College of Engineering, Bangalore, India

³Department of Bigdata Analytics, St.Joseph's College, Bangalore

⁴ AC College of Technology, Chennai, India

Abstract. Speech signal is based on the more complex anatomy of the mouth, throat, and nasal bone. Such speech signals are controlled by digital signal processing for encoding and by compressing the same speech. Many speech signal processing methods and standards have been developed by many researchers for audio codec's such as CELP. Advanced mathematical transformations such as Fourier, Lagrange, Hamilton, Legendre were developed for such applications. Signal processing using its equations through certain algorithms and methods to develop signals for various novel applications. Based on laryngeal pulses and oral cavity filter patterns, these techniques can be used to encode speech patterns more efficiently without losing signal clarity. In this paper a novel method of encoding speech with much simpler computations has been evolved. The famous spherical harmonic Legendre polynomials have been the key to the technique. Good compression, decent sound quality and above all, simplicity are the claimed advantages of the method. A new approach, using the spherical harmonic oscillator functions of Legendre is introduced here for this application. Thus, this paper describes this alternate scheme for speech compression and coding. Its usefulness lies in the simplicity relative to the more involved CELP techniques currently in use.

Keywords: Legendre Polynomial, CELP, Audio codec's, Speech processing, LPC

1 Introduction

Speech signal Modeling elicited from the vocal chords that have tested many mathematicians and created many levels of audio codec's. CELP is one such example. Using digital signal processing, the coding of speech was developed since last thirty years. Many CELP codec's and other standards have been developed from Linear Predictive Coding (LPC) and are used in GSM and other communication systems. [1].The accessibility of the speech waveform as an analog power signal has shifted interest from machines to electronics and now, to Digital signal processing hardware, for encoding, decoding and processing speech.

Jacob Bonesty [2], in his book, details Dudley's writings. Interpretation of speech in terms of carrier or exciting function and time-changing spectral envelopes is still used as the basic representation today. Vocoders such as Format Vocoder and Voice-Stimulated Vocoder have been found.

The GSM system supports a number of specific audio codecs [3]. Very much work has been done on mathematical modeling of the human vocal tract. The early works concentrated on the use of a filter to reproduce the sound from an excitation signal. The spectral envelope of the signal is estimated using a simplified poles only filter. This filter is termed as the LPC filter. The excitation source is supposed to vary between sounds for vowels, diphthongs and consonants. CELP codec, which uses a principle called "analysis by synthesis". In this process, encoding is done by conceptually optimizing the decode signal in a closed loop system. By comparing a variety of generated excitation pulse streams and chooses the one that produces the best sounding signal. This needs a previously stored codebook of such patterns of excitation. For a sampled sound frame of 20ms, the best codebook vector is optimally picked for least error between reconstructed sound and the original sound. This vector number alone is sent to the other end, where also, the codec there would have this dictionary or codebook values stored in its firmware. A specific algebraic structure is used in ACELP codec codebooks. All the complexity of the methods could be supported only by the later day powerful DSP chips.

While the English language has more use of voiced sounds in a sentence, the methods developed are not optimal for other, say, the languages of the Indian continent. All the complexity of the methods could be supported only by the later day powerful DSP chips. For instance, in 2005 only, the code was developed by Microchip Inc., for their DSPIC 30 family chip [4].

Despite all the involved procedures of filter coefficient determination by solving the simultaneous equations of autocorrelation values in a 20ms frame of speech samples, the generation of excitation source from a series of pulses or from random noise (voiced/unvoiced sounds) and optimal choice of the latter, the resulting reconstructed sound is only just intelligible. Recently, CELP codec's robustness with regard to inter-frame interference has been studied by Nadir Benami Rouche, 2016 [5].

2 An alternative to model based speech compression

The representation for speech;(Using the LPC filter and Excitation) V and UV represent the voiced or unvoiced speech with gain of G as shown in figure 1. This model is not complete since even animal sounds possibly will be characterized; then, they have no understandable speech output. Hence, an alternative model and a very sophisticated model, is required for the human speech production which is based compression is the objective of this paper.

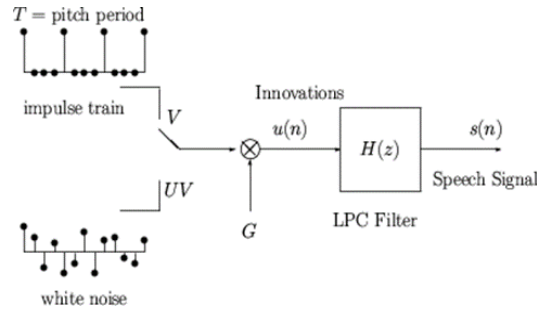


Fig.1. shows the model for speech. V is voiced, UV is unvoiced speech and G is gain.

$$H(z) = \frac{s(z)}{u(z)} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} + \dots + a_{10} z^{-10}} \quad (1)$$

The above transfer function can be extended in time series as in eqn. (2).

$$s(n) + a_1 s(n-1) + a_2 s(n-2) + a_3 s(n-3) + \dots + a_{10} s(n-10) = u(n) \quad (2)$$

Which can be write in sigma notation as in eqn. (3) ;

$$s(n) + \sum_{i=1}^{i=10} a_i s(n-i) = u(n) \quad (3)$$

The LPC representation that could be represent in vector form has the agitated temperament of throat output by the following three parameters such as gain, voice or un-draw, pitch time, as in eqn (4):

$$A = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, G, V, UV, T) \quad (4)$$

Matlab provides commands for polynomial based fitting to data as a series of powers of x. There is no method for Legendre polynomial based fitting. It should not be thought that because the Legendre polynomials use powers of x, the method is similar to the power series fitting method. Legendre polynomial compresses in one coefficient several powers of x. So, if a waveform component in the sound segment (200 samples – 25ms) comprises of a similar wave pattern, there will be only one coefficient which is large and the rest negligibly small. So, if we can reduce the number of compressed coefficients by sorting them and picking only those with good magnitude, the compression level is increased.

Matlab provides only ordinary series polynomial fitting in the curve fitting toolbox. If we compare the results of the coefficients c and a fitted for a sample data, the striking advantage of the Legendre polynomials method will be evident.

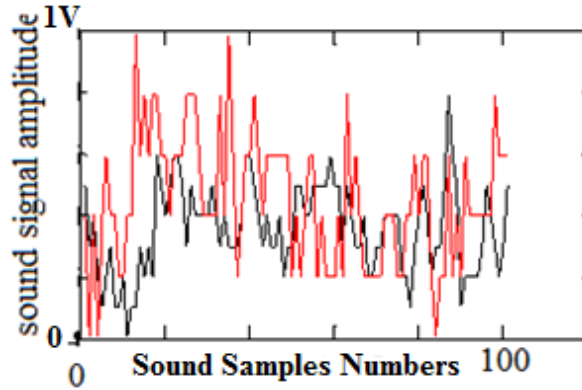


Fig.2. Original and CELP compressed file.,. Red is original, Black is reconstructed sound.

3 Spherical harmonic signals

The possibility of using signals based on Legendre polynomials is worth exploring, since the human larynx is partly conical and partly spherical and the vibrating nine chords across its circular cross section at the centre are causing movements of the spherical surface of the larynx at the junction of the vocal chords.

Consider now, the Legendre differential equation.

$$(1-x^2)y'' - 2xy' + n(n+1)y = 0 \quad (5)$$

The solution is a polynomial in x. we regard as a description with time t as independent variable.

$$(1-t^2)y'' - 2ty' + n(n+1)y = 0 \quad (6)$$

By omitting the first derived, we get, $(1-t^2)y'' + n(n+1)y = 0$ (7)

This can rewrite as, for $n=0$, as

$$(1-t^2)y'' + y = 0 \quad (8)$$

This can also rewrite as

$$y'' + \frac{y}{(1-t^2)} = 0 \quad (9)$$

$$y'' + \frac{y}{(1-t^2)} = 0$$

For the second order damped sinusoidal oscillation with a frequency ω . Its differential equation is as in eqn. (10) given below;

$$y'' + \omega^2 y = 0 \quad (10)$$

It has a variable frequency as per equation (5) with time t . The frequency is when $t=0$, $\omega=1$.

At $t=0.5$, it is

$$\sqrt{\frac{1}{1-t^2}} = \sqrt{\frac{1}{1-.5^2}} = 1.155 \quad (11)$$

The solution for the degree of difference equation (1) is given by

$$P_n(x) = \frac{1}{2^n n!} \frac{d^n}{dx^n} (x^2 - 1)^n \quad (12)$$

This can be easily done in Matlab. Figure 3 shows the waveforms for $n= 1$ to 8. It must be noted that higher values of n in P_n cause more oscillations. Therefore, the sound waves need to be fitted with the higher values of n as far as possible. The methods of curve fitting were tried by the authors for speech segments of 20ms and then reconstructed based on the coefficients of the Legendre polynomials.

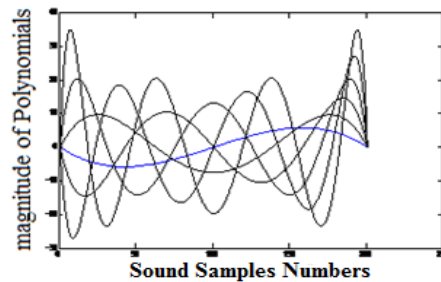


Fig.3. Shows the Associated Legendre Polynomials plot for six polynomials 3 to 8

The Associated Legendre Polynomials plot for six polynomials 3 to 8 is shown in figure 4. Thereafter, our compression routine is a simple one involving multiplications and additions. This enables even simple MCUs to be employed for digital telephony experiments. An example fitting of a decaying sinusoid of 10000 data points is given in figure.4. Thus, the fitting of speech data in an accurate manner by using these polynomials to the extent possible with the complex CELP method.

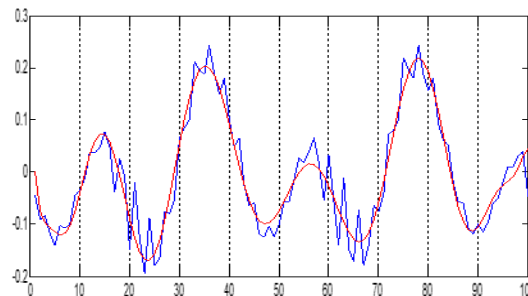


Fig.4 Shows the fits of the sample speech frame data 100 points with Associated Legendre polynomials.

The principle of fitting is based on the orthogonal property of the polynomials. Therefore, this is given by

$$\int_{-1}^1 P_n^m(x)P_l^m(x) dx = 0, \quad n \neq l \quad \int_{-1}^1 [P_n^m(x)]^2 dx = \frac{2}{2n+1} \frac{(n+m)!}{(n-m)!} \quad (13)$$

Hence, the function which is fitted with the polynomials have the coefficients A as a series.

$$f(x) = A_m P_m^n(x) + A_{m+1} P_{m+1}^m(x) + A_{m+2} P_{m+2}^m(x) + \dots \quad (14)$$

$$\text{Where } A(k) = \frac{2k+1}{2} \frac{(1-m)!}{(1+m)!} \int_{-1}^1 P(x)P_l^m(x) dx \quad (15)$$

In discrete form, the integral becomes a summation over the set of points which are mapped into the region -1 to $+1$, and we can take 100 or 200 samples of speech data in each compression frame, giving the coefficients A for each frame one after another. If we choose m polynomials, then, for every 100 or 200 points, there would be m coefficients. Thus, the

compression ratio will be ordinarily 200 or 100/m. The calculations for fitting were made using the derivations.

Further compression of the coefficients is also possible by eliminating those coefficients which have small values. Also quantization of the coefficients to just 6 bits instead of 8 (byte), will enable an increase in the ratio of compression by 33%. The figure 6 shows the speech sample for test purposes given in datacompression.com being compressed and then reconstructed. The fit is good and the reproduced sound using MATLAB's sound output command `soundsc (,)` is intelligible. We tested sounds input through the computer mike inbuilt in laptops. The `audioRecorder` object in MATLAB is used for input of speech data for some seconds. Using the same, tests were made for Indian languages also and found that the reconstructed (after compression) sounds are good, for Hindi and Tamil speech.

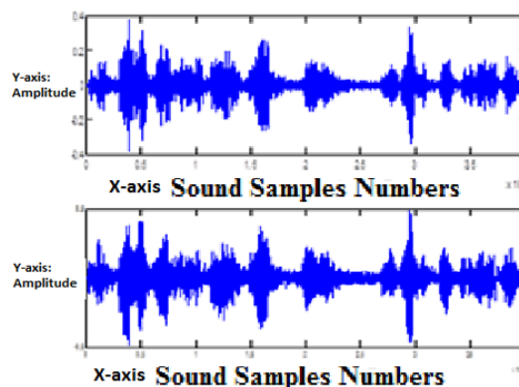


Fig.5 shows a sound sample for 3 seconds (at 8KHz sampling rate from Laptop sound card) –reconstructed and original sounds. (net compression ratio 9).

4 Conclusion

With a compression ratio of 9, the speech was intelligibly reconstructed. This gives a bit rate for telephone of $64/9 = 7.11$ Kbps. This is better than ACELP which does at a higher rate of 9.6Kbps. It is noteworthy that lower the rate of bits per seconds, the better the compression and hence possibility for real time transcoding.

References

- [1] H. Dudley, T.H. Tarnoczy, "The speaking machine of Wolfgang Von Kempelen", Journal Acoust. Soc. America. 22, pp.151– 166, online version, 2005
- [2] J.Benesty, M. M Sondhi, Y. Huang, Handbook of Speech Processing, Springer-Verlag Berlin Heidelberg, 2008.
- [3] DS70146A – Code for DSPIC 30 Family for CELP, Mouser.com, 2016.

- [4] A. McCree, et al., "A 4 kb/s hybrid MELP/CELP speech coding candidate for ITU standardization", In: Proc. IEEE International Conf. Acoust. Speech Signal Process, pp.629–632, 2002
- [5] N. B. Rouche, B. Boudra, A. Gomez, J. P. Cordoba and I. L. Estego, A dynamic FEC for improved Robust of CELP code, Springer AG pp.14-23, 2016