A Low Bandwidth Speech Coder Based on the Ratio Spectrum

Sowmiya Narasimhan
and
John G. Harris
Computational Neuro-Engineering Lab
University of Florida
Gainesville, FL 32611
{nsowmiya,harris}@cnel.ufl.edu

Abstract

We have implemented a low bandwidth speech coder using a new spectral representation called the ratio spectrum. The ratio spectrum is approximated in the digital domain by the normalized cumulative sum of the magnitude of the Fourier spectrum. This cumulative sum is guaranteed to be a monotonic nondecreasing function between zero and one. For each frame, frequency values are extracted by uniformly sampling along the ratio axis. These frequency values are coded and transmitted along with the overall power level, pitch and voicing information. The coder achieves a 1.59Kbps coding rate for intelligible speech. The speech quality of the coder can be improved by increasing the number of samples with subsequently higher bitrates.

Introduction

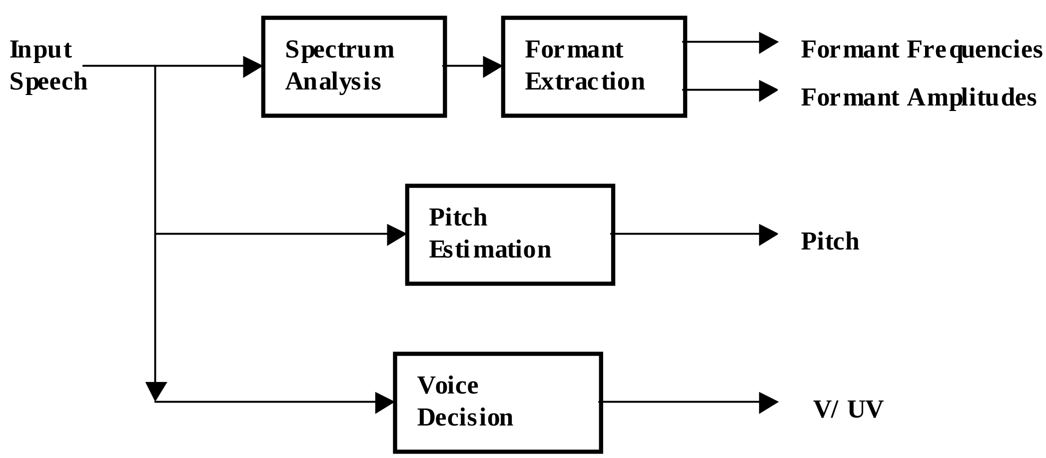

Figure 1. Formant Coder Block Diagram

We have designed a ratio spectrum speech coder by combining the advantages of formant vocoders and channel vocoders. A block diagram of the encoding part of the formant vocoder is shown in Figure 1.
Formant vocoders require that three formants be determined in each frame [2]. Their center frequency, bandwidth and amplitude are then coded and transmitted along with the pitch and voicing information.

Formant vocoders are very low bandwidth but suffer from the difficult problem of accurately determining, characterizing and tracking formants. Many times the formants are difficult for a human to determine from a power spectrum plot. Errors in formant determination severely degrade the quality of the decoded speech.

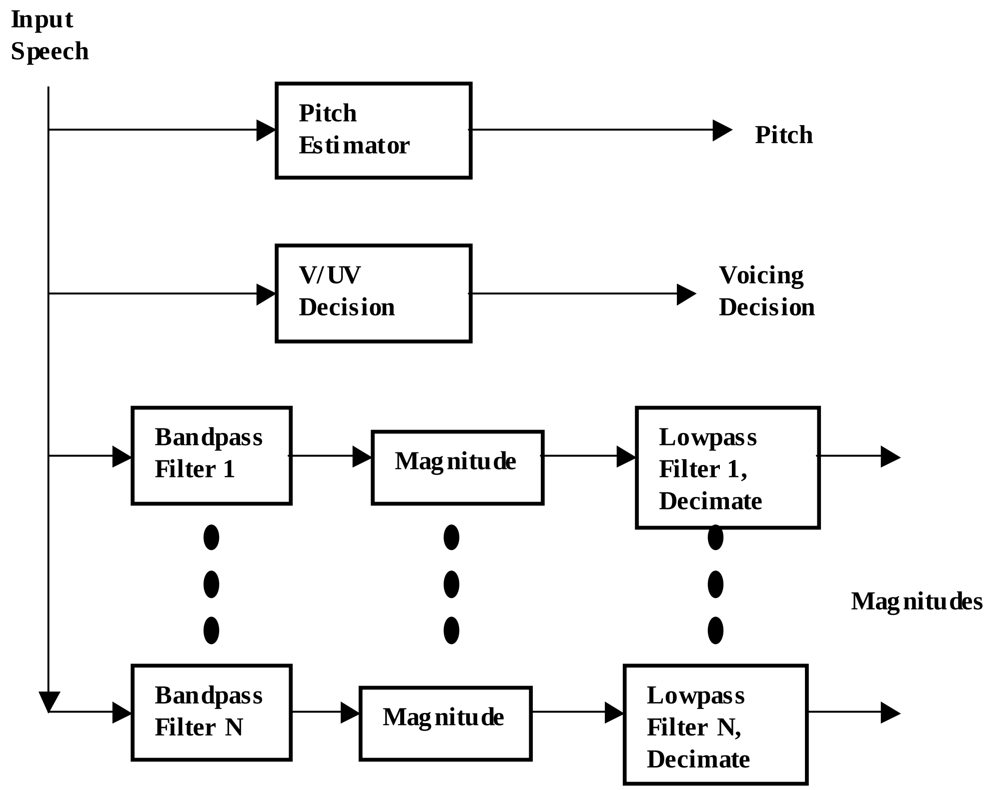

Figure 2. Channel Coder Block Diagram

Channel vocoders represent the other extreme where the frequency axis is uniformly sampled over its entire range and the proper amplitudes are coded and transmitted [2]. A block diagram of the channel vocoder is shown in Figure 2. The formant vocoder suffers from the problem of transmitting too much information. For instance, information is transmitted about frequencies where no signal energy is present. Intuitively, the ratio spectrum coder strikes a balance between formant and channel coders by carefully transmitting information only where there is energy in the signal.

Ratio Spectrum Encoder

The ratio spectrum is formally defined (in both the continuous-time and discrete-time domains) as the ratio of the power of a low-pass filtered signal to the power of the original unfiltered signal vs. the filter cutoff frequency [3]. The spectrum is necessarily a monotonic function of frequency that is bounded between zero and one. Though we can prove that this transform space is mathematically equivalent to the power spectrum of a signal, the ratio spectrum provides a number of computational advantages.

A major advantage is that analog hardware can efficiently compute the continuous-time ratio spectrum. Another advantage is the ease of sampling the ratio spectrum such that the frequency samples tend to fall in regions of high spectral energy. In this paper we use an FFT-based approximation to the ratio spectrum expressed as:

$$\Re(k) = \frac{\sum_{n=0}^{k} |X(n)|^2}{\sum_{n=0}^{N-1} |X(n)|^2}$$

Since we perform a cumulative sum of a nonnegative function (the power spectrum), this approximation still preserves the properties of a monotonically nondecreasing function ranging from zero to one. The relationship between the power spectrum and the ratio spectrum is similar to the relationship between probability density functions and cumulative distribution functions. Before we go ahead with the computation of the Ratio Spectrum, pre-emphasis is done in order to boost the higher frequencies. This is done to make sure that the higher frequencies fall in the sampling region.

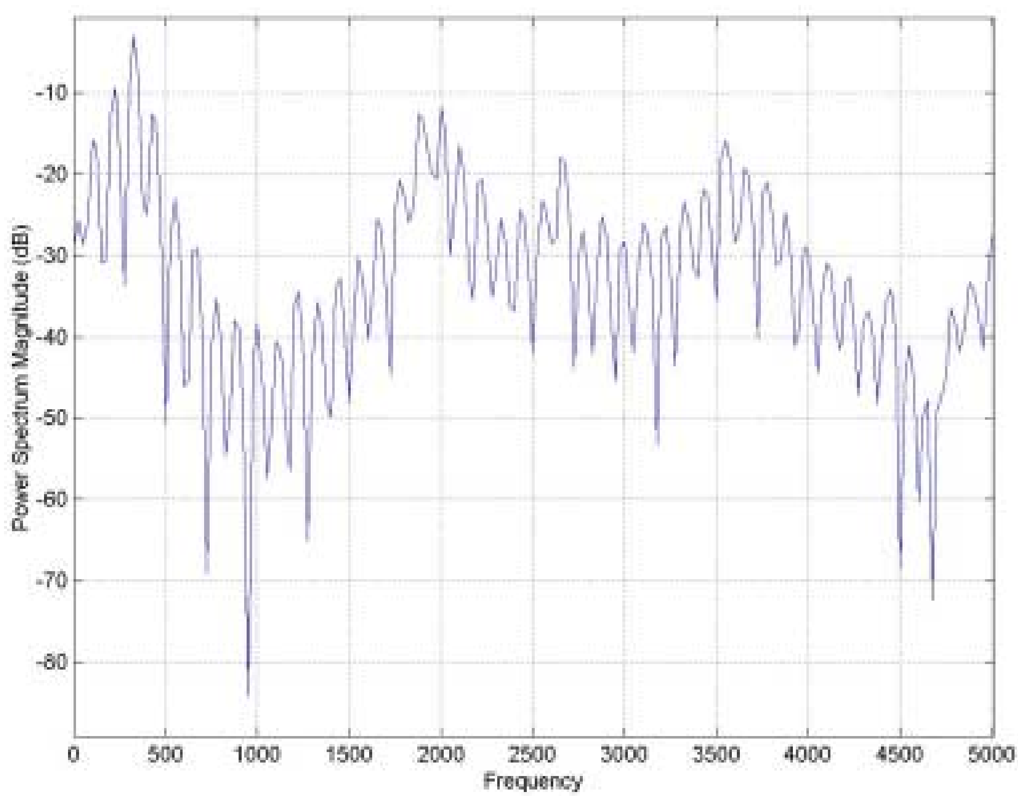

Figure 3. Power Spectrum of a single frame (Original signal)

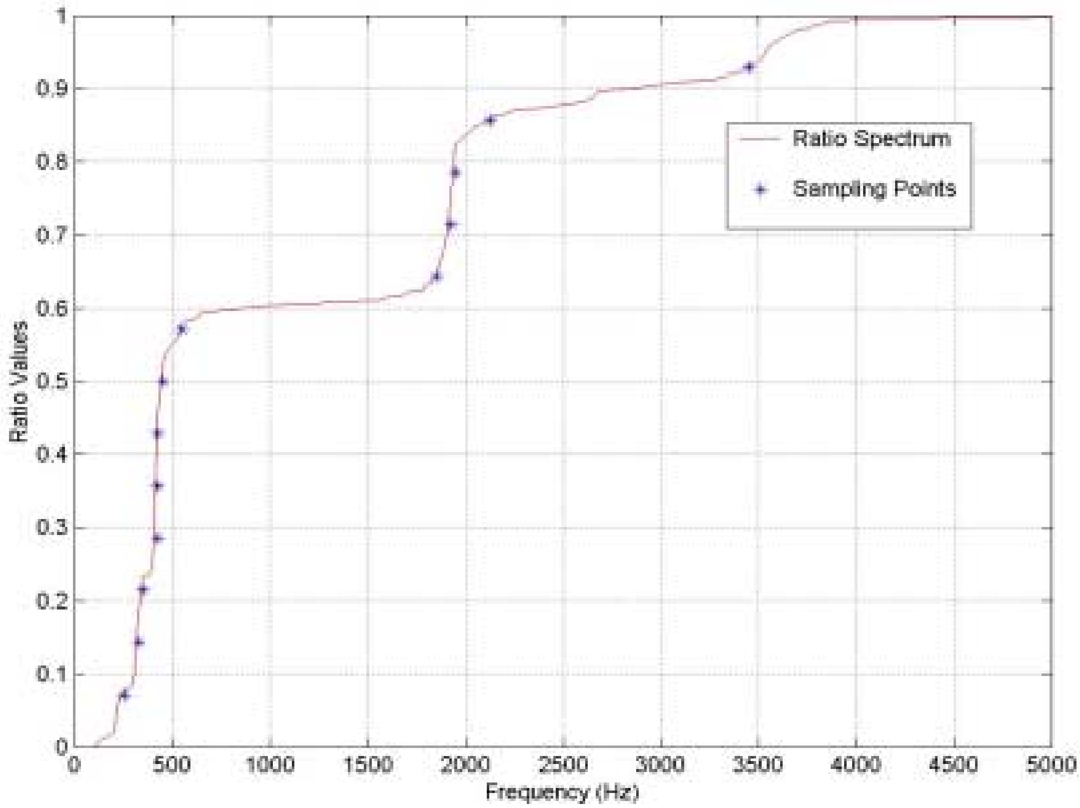

Figure 4. Ratio Spectrum of a single frame

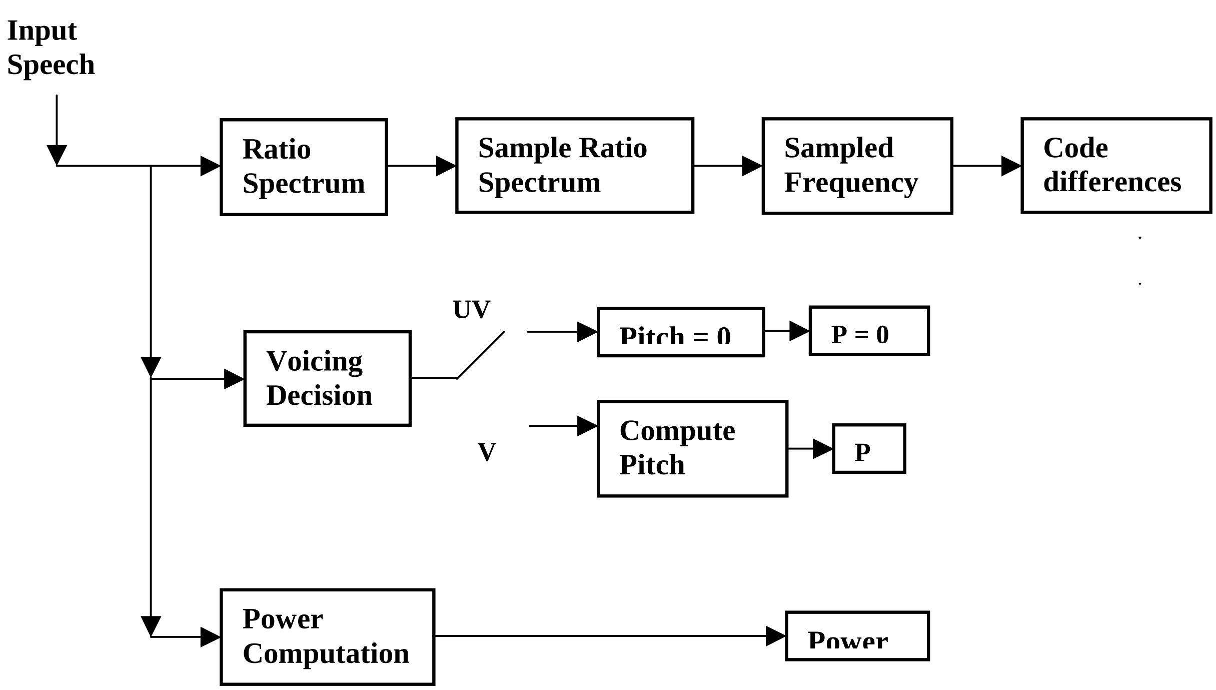

Figure 5. Ratio Spectrum Encoder Block Diagram

Decoder

In earlier work, we used spectrum inversion techniques to implement a full audio coder using only ratio spectrum samples [4]. This audio coder required 100 or more frequency samples to achieve high quality audio. In order to reduce the bandwidth, the ratio spectrum speech coder makes explicit assumptions about speech such as the existence of a pitch period and voiced/unvoiced regions. As with the formant and channel coders, the voicing decision determines whether random noise or a train of impulses should be fed into the parallel bank of bandpass filters. If the frame is unvoiced, the pitch value determines the spacing between pulses. One major difference between the ratio spectrum coder and the formant or channel coder is that no information about the amplitude is sent. If one region of the spectrum has more energy then more frequency samples will be included in that region. A block diagram of the ratio spectrum decoder is shown in Figure 6.

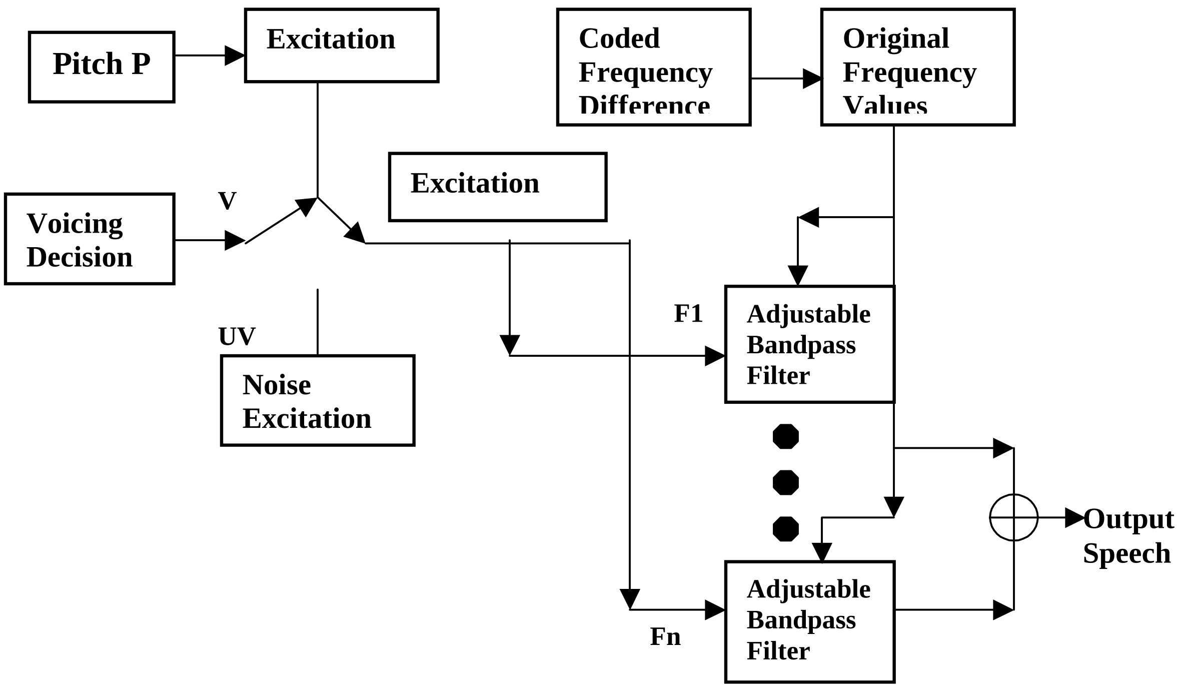

Figure 6. Ratio Spectrum Decoder Block Diagram

The major challenge is how to reconstruct a plausible speech signal from the ratio spectrum samples. A bandpass filter is centered at each frequency sample with a bandwidth related to the critical bandwidth at that frequency. Random noise or impulse trains are fed through this parallel bank of filters. Figure 7 shows an example of the power spectrum of a reconstructed time domain signal.

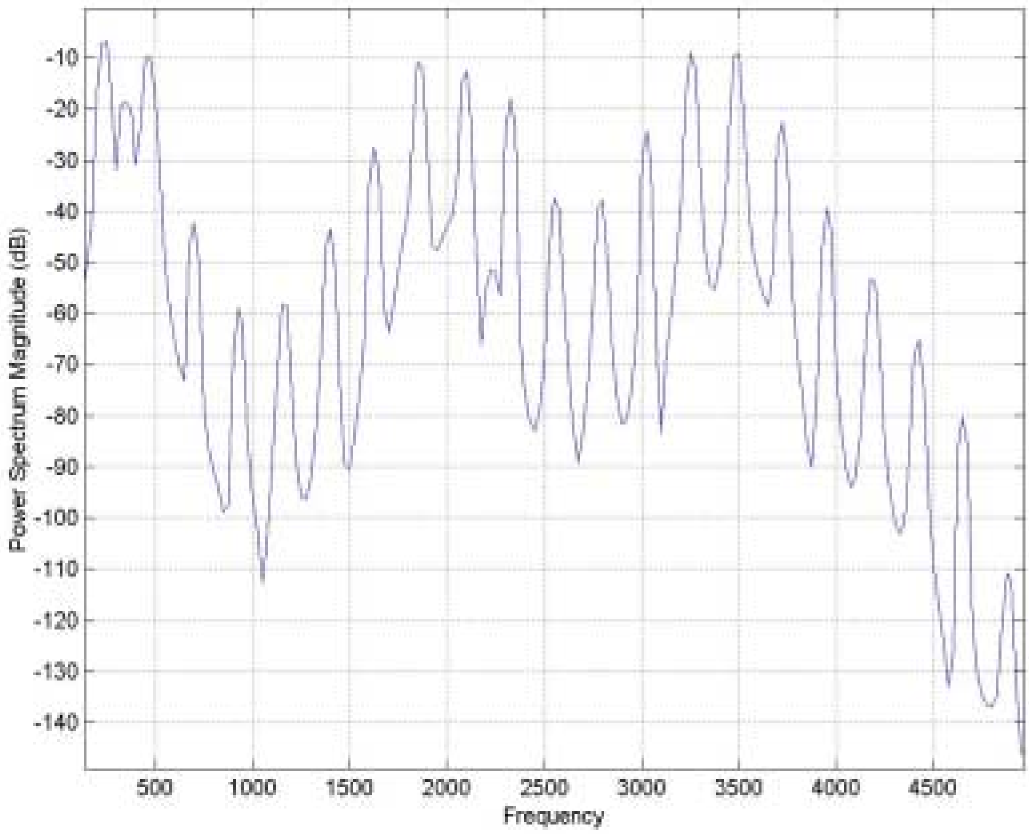

Figure 7. Power Spectrum of the reconstructed signal for a single frame.

Bandwidth of the Ratio Spectrum Coder

In order to calculate the bandwidth of the ratio spectrum coder, we estimated the probability density function of the frequency differences over many sentences in our database. Since this distribution is very much peaked, Huffman coding can achieve a tremendous bit reduction. In order to estimate the compression from Huffman coding, we use Shannon's entropy shown below [1]:

$$\mathrm{H}_i = -\sum_i p_i(j) \log p_i(j)$$

It is well known that Huffman codes are guaranteed to achieve the coding to within one bit of the optimal number of bits given by the entropy[1], we add one bit to the above expression to compute an upper bound to the bandwidth. Since each frame also uses 6 bits to code pitch and 6 bits to code power in the frame, these 12 bits are added in for each frame. However, the dominant component of the bandwidth is the coding of the 10 or more frequency samples.

# of Samples	Entropy (bits)	Coder Bitrate (Kbps)
10	3.11	1.59
13	2.90	1.88
15	2.81	2.08
17	2.68	2.24
20	2.48	2.45
25	2.31	2.84

Table 1. Change in coder bitrate vs. number of samples

Table 1 shows the resulting bitrate for various ratio spectrum coders based on the number of frequency samples that are taken. For instance, the 10 samples have an entropy of 3.11 bits rather than the 7 bits

required to naively code the samples. The 10-sample coder achieves a bitrate of 1.59Kbps when the pitch (6 bits) and power (6 bits) values are included for each frames out of the 30 frames per second. More samples improve the quality of the coder and reduce the entropy for coding a single frequency value. The decrease in entropy slows down the growth of bitrate so that a doubling in the number of samples results in an increase of only 50% in the bitrate.

Conclusion

We have implemented a low bandwidth speech coder using a new spectral representation called the ratio spectrum. The ratio spectrum is approximated in the digital domain by the normalized cumulative sum of the magnitude of the Fourier spectrum. This cumulative sum is guaranteed to be a monotonic nondecreasing function between zero and one. For each frame, frequency values are extracted by uniformly sampling along the ratio axis. These frequency values are coded and transmitted along with the overall power level, pitch and voicing information. The coder achieves a 1.59Kbps coding rate for intelligible speech.

We have already implemented the ratio spectrum audio coders in real-time on the C31 DSP board [5] but the ratio spectrum speech coder described here requires much less bandwidth. The procedure is broken into four parts namely the C code for the procedure of the coder, C code for the interfacing of the input and output devices of the board, assembly code at places where the c code takes more computation time and finally the integration. The testing and coding of each part is being done separately. We are now studying the real time implementation on the ratio spectrum coder on both the TMSC6x and the TMSC3x DSP processors in our laboratory.

References

[1] John R. Pierce. ***An Introduction to Information Theory- Symbols, Signals and Noise***. Dover Publications, Inc., New York, 1980.

[2] Randy Goldberg, Lance Riek**.** ***A Practical Handbook of Speech Coders***. CRC Press, United States, 2000.

[3] Shao-Jen Lim. ***The Ratio Spectrum***. PhD Thesis, University of Florida, 1999.

[4] Augustin De Jesus Roca. ***The Discrete Ratio Spectrum with Application to Speech and Audio Coding***. M.S Thesis, University of Florida, 1999.

[5] Jim Nickila. ***Real Time Speech Coding Using the Ratio Spectrum.*** M.S Thesis, University of Florida, 1999.