MATLAB Software for Speech Analysis and Coding Demonstration

Şef lucrări Cornel Balint
Politehnica University Timisoara

ABSTRACT A set of MATLAB routines was developed in order to perform a speech signal analysis and coding this signal using some standardized speech coding algorithms such as FS-1015 LPC-10e, FS-1016 CELP, ETSI GSM. The simulation software provides an interactive environment that display time- and frequency-domain representations of input signal and reconstructed speech on an IBM PC compatible equipped with a standard sound card.

1 Introduction

The new international speech coding standards for computer and telecommunication applications, such mobile communication (GSM) or Internet telephony, videoconferencing demand a efficient codec implementations on a variety of processor architectures.

Although surveys of state-of-the-art speech coders and standards appeared in the speech coding literature [Spa94], detailed algorithmic descriptions and implementation details are often omitted. A set of educational software is therefore needed to better understanding this tutorial literature. This paper describes such a software tool developed for the purpose of evaluating, simulating, and understanding standardized speech coding algorithms. In order to understanding thet standards, some speech signal analysis are implemented.

The paper focused on linear prediction (LP) codecs which have been implemented: the 2.4 kbit/s FS-1015 LPC-10e, the 4.8 kbit/s FS-1016 CELP, the 13 kbit/s ETSI GSM, and the 32 kbit/s G.721 ADPCM. A user-friendly interface was developed which enables users to experiment with a large variety of input signals, examine graphical representations of analysis/synthesis parameters, playback reconstructed output speech, and compare quality of output speech associated with the different coding standards. Graphical output is also available. MATLAB has chosen for its advantages: users are able to study a variety of signal and parameter plots, experiment the effects of channel noise and simulate the channel errors, and easy modify algorithm parameters; MATLAB code is compact and is easy to understand the algorithms; MATLAB software run on a variety of computers.

2 Speech Signal Analysis

Speech signal analysis section of the software accept input signal samples in .WAV format and permit:

- time – domain representation of signal waveform
- short time autocorrelation
- discrete Fourier transform
- spectrogram
- real cepstrum
- average magnitude difference function (AMDF)

For speech signal processing [Pro+91], [DPH93], in order to assume that the signal is short – time stationary on multiply the signal by a window function, that is zero outsides the defined range. The most common window is the rectangular window, defined as:

$$p_N(n) = \sigma(n) - \sigma(n-N) \qquad (1)$$

and thereby the speech frame signal become, by multiplying with the window function:

$$f_x(m,n) \overset{def}{=} x(n)p_N(m-n) \qquad (2)$$

To avoid discontinuities at the ends of considered interval, Hamming or cosine windows are used.

The autocorrelation function is given by [DPH93]:

$$^1r_x(\lambda,m) = \frac{1}{N} \sum_{n=m-N+1+|\lambda|}^{m} x(n)x(n-|\lambda|) \qquad (3)$$

For the short time signal power spectrum, the software use the relation (4) – (6)

$$X_n(e^{j\omega}) = \sum_{n=-\infty}^{\infty} w(n-m)x(m)e^{-j\omega n} \qquad (4)$$

$$X_n^1(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n-m)w(m)e^{j\omega n} \quad (5)$$

$$S(n,\omega) = \left|X_n(e^{j\omega})\right|^2 \quad (6)$$

The real cepstrum of the input speech signal is defined by:

$$c_s(n,m) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left\{\log\left|\sum_{l=-\infty}^{\infty} f(l,m)e^{-j\omega l}\right|\right\}e^{j\omega n}d\omega = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left\{\log\left|\sum_{l=m-N+1}^{m} f(l,m)e^{-j\omega l}\right|\right\}e^{j\omega n}d\omega \quad (7)$$

The average magnitude difference function (AMDF) is defined by:

$$D_n = \delta_n\left[\frac{1}{N}\sum_k x^2(k) + \frac{1}{N}\sum_k x^2(k-n) - \frac{2}{N}\sum_k x(k)x(k-n)\right]^{\frac{1}{2}} \quad (8)$$

For the time – domain representation of signal waveform, the default option is to display the entire input file. A "Zoom" function is available, in order to display a selected region of signal. It is possible to select the beginning point of the short frame of input signal that is used to compute short time autocorrelation function, discrete Fourier transform, spectrogram or real cepstrum.

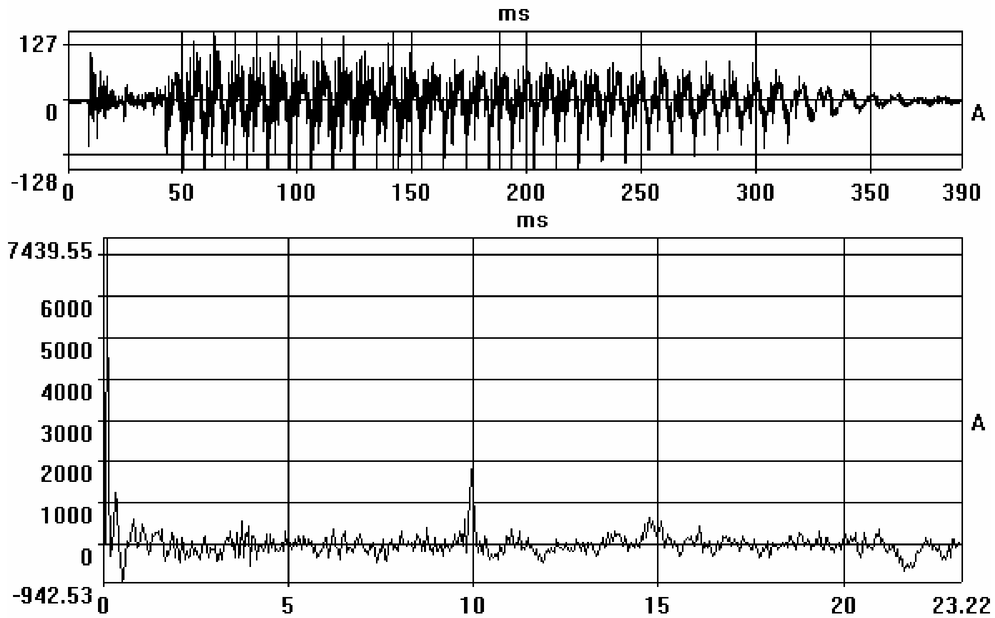

Fig. 1. Speech signal sample and power spectrum

The length of this frame used for discrete Fourier transform compute is selectable in power of 2 steps, in range 64 – 2048. The used window for the signal frame is also selectable. A "Play" function, using the sound card in Windows environment, is available for the analyzed speech signal or for a selected frame from signal.

Figures 1, 2 and 3 illustrated a speech signal sample, for which was computed and displayed the power spectrum, using a 1024 sample length frame and a Hamming window, the autocorrelation function and the average magnitude difference function.

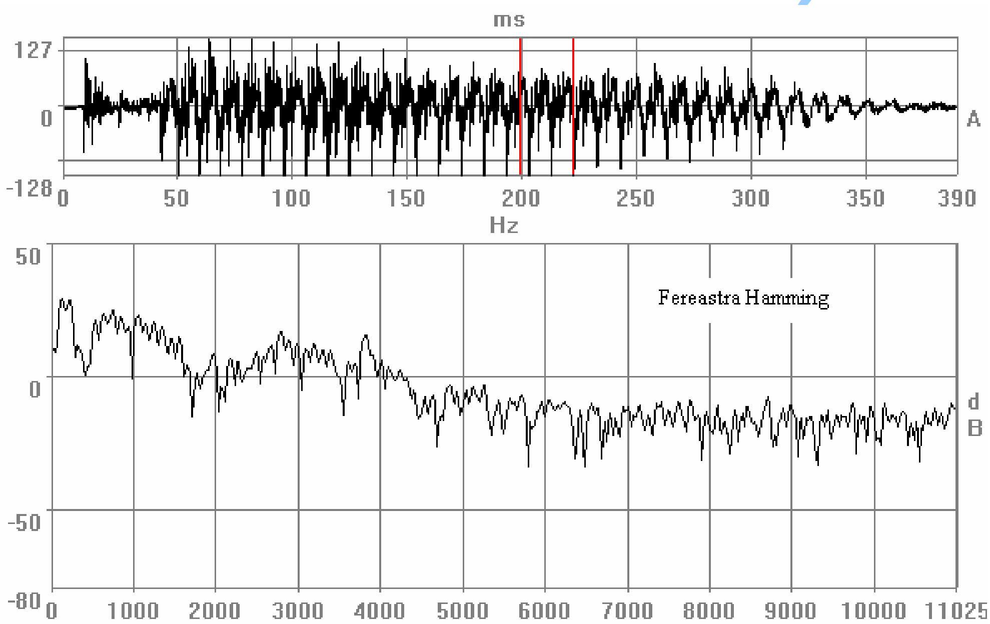

Fig. 2. Speech signal sample and autocorrelation function

3 Codec Simulations

Codec simulations section of software accept original input signal samples from a standard mono .WAV type input files, perform analysis at the transmitter, transmit parameters through a simulated channel, run synthesis at the receiver, and then generate .WAV output files. The recommended sampling frequency is 8 kHz, in order to allows comparison between coding techniques.

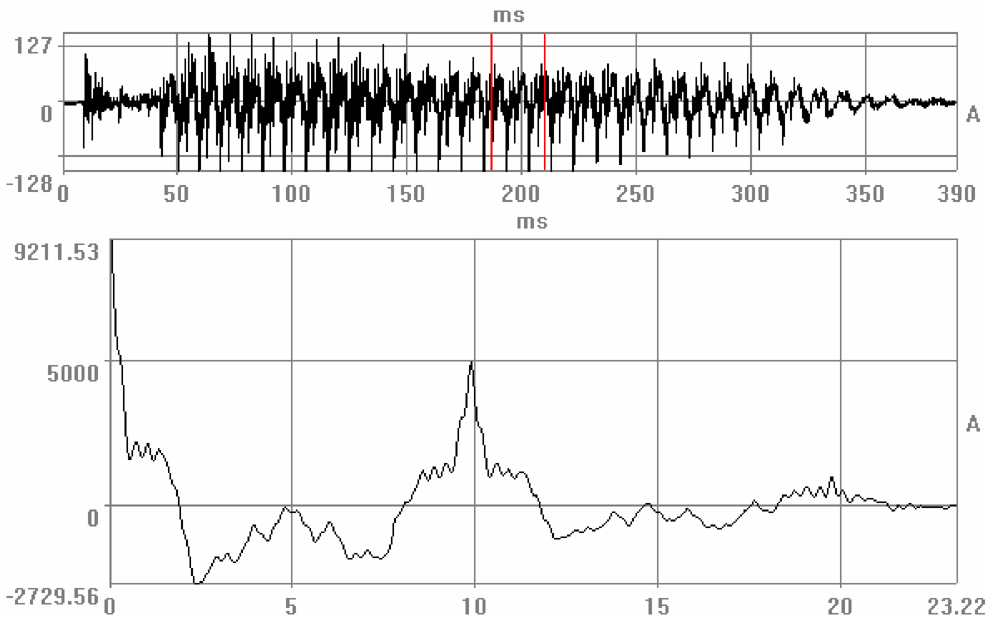

Fig. 3. Speech signal sample and average magnitude difference function

The simulated codec using the LP speech signal coding. The LP speech signal production model suppose that during a stationary frame of signal, the model can be approximated by a all pole system function of the form:

$$A(z) = \frac{1}{1 - \sum_{i=1}^{P} a_i z^{-i}} \tag{9}$$

driven by an appropriate excitation sequence, that is ideally a periodic impulse train for voiced sounds and a uncorrelated, white noise, for the unvoiced case.

In order to code speech signal using LP, the speech signal is divided into stationary frame, and for each frame the prediction coefficients a_i and the appropriate excitation sequence is determined. The LP speech signal coding methods are different by the used excitation sequence [Spa94]:

– residual excited LP use as excitation a periodic impulse train for voiced sounds and a uncorrelated, white noise, for the unvoiced case. The excitation parameters are the pitch period and the gain (example: LPC 10)
– regular pulse. The excitation parameters are the pulse position in a grid pulse and the pulse amplitude (example: RPE – GSM)

– code or vector. The excitation is a vector or a code, selected from a codebook.

A time-domain display window allows comparisons between input and reconstructed output waveforms. Is possible to see the differences in waveform matching between two algorithms. Comparisons are enhanced by a facility that allows examination of the reconstruction error.

A frequency-domain viewing window is also available allowing comparison of magnitude spectra between input and reconstructed output speech. Magnitude spectral estimates are generated using a 1024-point FFT. The LPC envelope, corresponding to quantized predictor coefficients received by the decoder, can be superimposed on both plots. One can observe spectral matching properties. In all LPC coding methods, short-term spectral characteristics are comprised in an all-pole synthesis filter. It is the excitation models which are different in these algorithms in terms of complexity, performance, and bit rate. The excitation viewing window allows observation of excitation sequences in time and frequency. Comparisons help the understanding of different excitation methodologies. In LPC-10e codec, the excitation signal for the all – pole model is a repetitive pulse (pitch) and the added noise. GSM excitation is a regular pulse excitation (RPE), in which each frame of regularly spaced pulses has distinctly different amplitude patterns than its predecessor. In CELP, random vectors have been combined with lag search vectors to obtain an optimal excitation. It is possible to display the pole locations of the decoder's LPC synthesis filter through a Z-Plane view.

The input signal for the vowel “a” is represented in figure 4. Figure 5 show the error signal for the input signal of fig. 4, calculated with a LPC analysis with 10 coefficients, and figure 6 represent the error signal power spectrum approximated using the LPC coefficients.

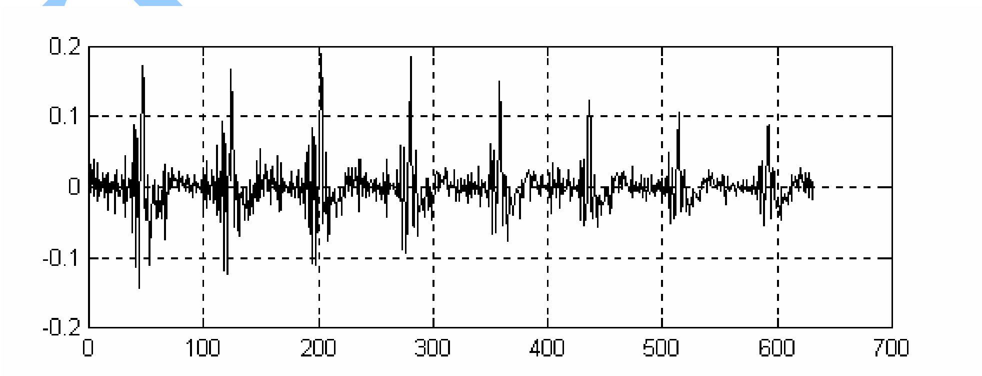

Fig. 4 Input codec signal

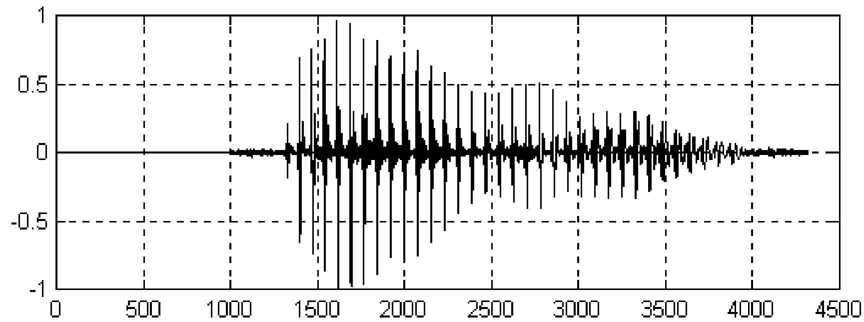

Fig. 5. Error signal

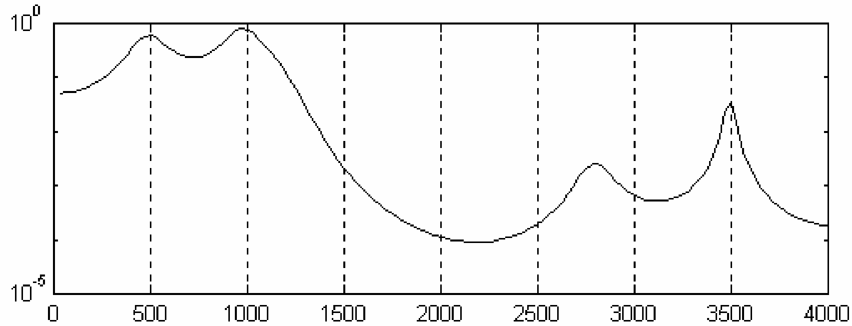

Fig. 6 Error signal power spectrum

4 Conclusion

The paper present a set of speech analysis and coding simulation software developed in order to illustrate these techniques and to create a didactic tool for speech signal processing, using a simple but powerful programming environment: Matlab [GF94].

References

[DPH93] **J. Deller, J. Proakis, J. Hansen**, *Discrete-time processing of speech signal*, Macmillian Publishing Company, 1993

[GF97] **M. Ghinea, V. Firețeanu**, *MATLAB – Calcul numeric. Grafică. Aplicații*, Editura Teora, Bucureşti, 1997

[Pro+91] **J. Proakis et al.**, *Advanced Digital Signal Processing*, Macmillian Publishing Company, 1991

[Spa94] **A. Spanias**, *Speech Coding: A Tutorial Review*, Proc. IEEE, pp. 1541-1582, Oct. 1994.