THE SPEECH CODING AND DECODING ALGORITHMS

VICTOR ABABII*, **VIORICA SUDACEVSCHI****

TECHNICAL UNIVERSITY OF MOLDOVA

E-mail:* ababii@mail.utm.md, ** svm@mail.utm.md

Abstract: A method of speech coding and decoding is proposed. The speech coding algorithm is based on first derivate calculation of input speech signal, identification of critical points and input signal amplitude in these points, time period measurement between critical points. The result of codification represents a sequence of amplitudes and time periods. The decoding algorithm utilizes values of COS or SIN functions for reconstruction of the input speech.

Key word: speech coding, reproduced speech, encoder algorithm, decoder algorithm.

1. INTRODUCTION

Narrowband speech codec's are used to give an efficient digital representation of telephone bandwidth speech. Often the speech is band limited to between 200 and 3400 Hz, and is sampled at 8 kHz. An ideal speech codec will represent this speech with as few bits as possible, while producing reconstructed speech which sounds identical, or *almost* identical, to the encoded speech. Of course in practice there is always a trade-off between the bit rate of the codec and the quality of its reconstructed speech [1].

Speech codec's are often broadly divided into three classes - waveform codecs, source codecs and hybrid codecs. Typically waveform codecs are used at high bit rates, and give very good quality speech. Source codecs operate at very low bit rates, but tend to produce speech which sounds synthetic. Hybrid codecs use techniques from both source and waveform coding, and give good quality speech at intermediate bit rates.

The simplest form of waveform coding is Pulse Code Modulation (PCM), which merely involves sampling and quantizing the input waveform (standard G 711). Narrow-band speech is typically band-limited to 4 kHz and sampled at 8 kHz. If linear quantization is used then to give good quality speech around twelve bits per sample are needed, giving a bit rate of 96 kbits/s. This bit rate can be reduced by using non-uniform quantization of the samples [3].

A commonly used technique in speech coding is to attempt to predict the value of the next sample from the previous samples [2]. If the predictions are effective then the error signal between the predicted samples and the actual speech samples will have a lower variance than the original speech samples. This is the basis of Differential Pulse Code Modulation (DPCM) schemes - they quantize the *difference* between the original and predicted signals.

Adaptive Differential Pulse Code Modulation (ADPCM) codecs are waveform codecs which instead of quantizing the speech signal directly, like PCM codecs, quantize the difference between the speech signal and a prediction that has been made of the speech signal. In the mid 1980s the CCITT standardized a 32 kbits/s ADPCM, known as G721, which gave reconstructed speech almost as good as the 64 kbits/s PCM codecs. Later in recommendations G726 [4] and G727 [5] codecs operating at 40, 32, 24 and 16 kbits/s were standardized.

Waveform coders are capable of providing good quality speech at bit rates down to about 16 kbits/s, but are of limited use at rates below this. Source codec's on the other hand can provide intelligible speech at 2.4 kbits/s and below, but cannot provide natural sounding speech at any bit rate. Although other forms of hybrid codec's exist, the most successful and commonly used are time domain Analysis-by-Synthesis (AbS) codec's. AbS codec's were first introduced in 1982 by Atal and Remde with what was to become known as the Multi-Pulse Excited (MPE) codec. Later the Regular-Pulse Excited (RPE) and the Code-Excited Linear Predictive (CELP) codec's were introduced.

Currently the most commonly used algorithm for producing good quality speech at rates below 10 kbits/s is Code Excited Linear Prediction (CELP). This approach was proposed by Schroeder and Atal in 1985, and differs from MPE and RPE in that the excitation signal is effectively vector quantized [7].

The complexity of the original CELP codec was much too high for it to be implemented in real-time - it took 125 seconds of Cray-1 CPU time to process 1 second of the speech signal. Since 1985 much work on reducing the complexity of CELP codec's, mainly through altering the structure of the codebook, has been done. Also large advances have been made with the speed possible from DSP chips, so that now it is relatively easy to implement a real-time CELP codec on a single, low cost, DSP chip. Several important speech coding standards have been defined based on the CELP principle, for example the American Department of Defense (DoD) standardized in 1991 a 4.8 kbits/s CELP codec as Federal Standard 1016, and the CCITT low-delay 16 kbit/s codec which was developed at AT\&T Bell Labs, and was standardized in 1992 as G728 [6].

One of the most important conditions in speech coding systems design is the low frequency and its narrow dynamic range. In this paper a method of speech coding is proposed that gives a high compression coefficient and a qualitative index of a reproduced speech signal.

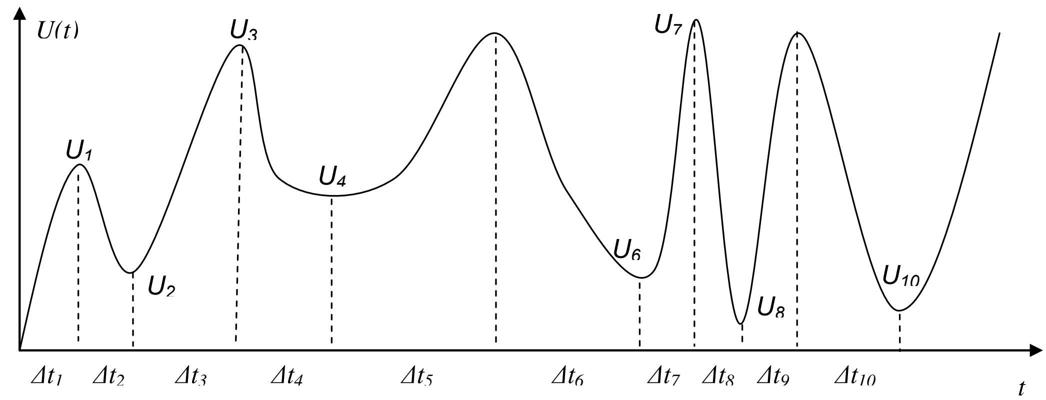

Fig. 1. Input Speech

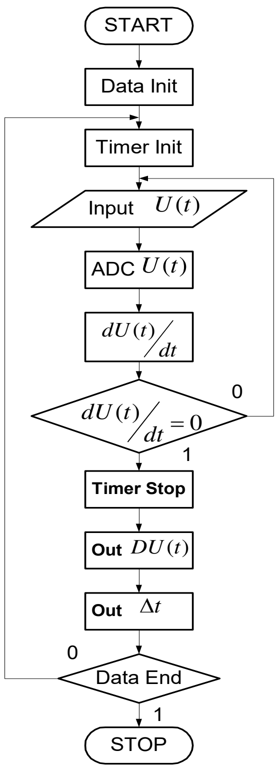

Fig. 2. Block diagram of the data coding algorithm

Let's have an input voice signal $U(t)$ that is defined at certain time period [0, T] (Fig. 1), at which the signal continuous condition is performed. This will provide the calculation of the differential value $dU(t)/dt$.

2. SPEECH CODING ALGORITHM

Speech coding algorithm is based on the first derivative property of the continuous function for the input speech signal $U(t)$. A time differentiation is done for the input speech signal, the critical points $dU(t)/dt = 0$ and time periods Δt between these critical points are calculated. The block diagram for the speech coding algorithm is shown on Figure 2, where:

START – The beginning of the algorithm;

1. **Data Init –** The initialization of the device and variables necessary for speech coding;

2. **Timer Init** – The timer initialization for Δt periods measuring;
3. **Input** $U(t)$ - Speech signal insert;
4. **ADC** $U(t)$ - The analog-to-digital transformation of the input speech signal $U(t)$. As the result the digital code $DU(t)$ is obtained;
5. $dU(t)/dt$ - The calculation of the time derivative of the input signal;
6. $dU(t)/dt = 0$ - If the input signal pass through the critical point then go to step 8, if not go to step 4;
7. **Timer Stop** – The end of the time period Δt measuring;
8. **Out** $DU(t)$ - Storage of the digital code $DU(t)$ value in the critical point for further processing;
9. **Out** Δt - Storage of the time period Δt value between critical points for further processing;
10. **Data End** – If all input data have been processed then go to step 12, if not return to step 3.
11. **STOP** – End of the algorithm.

The results of the speech coding algorithm (Table 1) will be the set of input signal's amplitudes in critical points $U(t)$,

$\left\{ U(t_i) \middle| \; \frac{dU(t)}{dt} = 0, \forall i = 1,...,n \right\}$ and the set of time periods Δt

$\left\{ \Delta t_i \middle| \left| t_{i+1} - t_i \right|, \forall \frac{dU(t_{i+1})}{dt} = 0 \;\&\; \frac{dU(t_i)}{dt} = 0, i = 1,...,n-1 \right\}$ between them.

Table 1. The results of the speech coding algorithm.

U_1	Δt_1	U_2	Δt_2	U_3	Δt_3	U_4	...	Δt_{n-1}	U_n

3. SPEECH DECODING ALGORITHM

Speech decoding is based on using of the approximation function COS or SIN and first derivate properties of the continuous function $U(t)$. In case of using function COS to approximate the voice signal the interval $[0, \pi]$ is determined for the decrease part of the waveform output signal $U^*(t)$ and interval $[\pi, 2\pi]$ is determined for the increase part of $U^*(t)$. When function SIN is used for approximation the output signal $U^*(t)$ decrease on interval $\left[\frac{\pi}{2}, \frac{3\pi}{2}\right]$ and increase on interval $\left[\frac{3\pi}{2}, \frac{5\pi}{2}\right]$. The algorithm for speech decoding is shown on Figure 3, where:

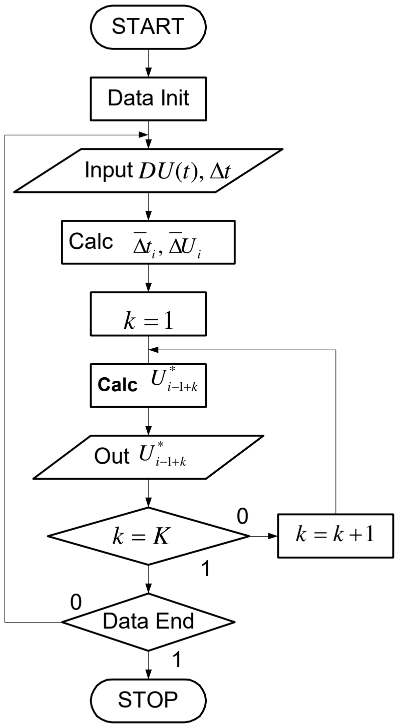

Fig. 3. Block diagram of the data decoding algorithm

1. **START** – The beginning of the algorithm;
2. **Data Init** – At this step the function for decoding is selected (COS or SIN) and the accuracy of input data processing is determined. The accuracy of data processing depends on digitization level K on the interval between two critical points;
3. **Input** $DU(t), \Delta t$ - Serial input of the speech signal amplitude $DU(t)$ in the critical point and the time period Δt between the preceding critical point and the current one;
4. $\overline{\Delta} t_i, \overline{\Delta} U_i$ **calculation** - Digitization step calculation between two critical points according to the following formula:

$$\overline{\Delta} t_i = \Delta t_i / K \qquad (1)$$

and determination of the output signal variation on interval $[\Delta t_i]$ according to formula (2)

$$\overline{\Delta} U_i = |U_{i-1} - U_i| \qquad (2);$$

5-9. The value of the restore speech signal calculation according to the following formula:

$$U^*_{i-1+k} = \begin{cases} U_{i-1} + \overline{\Delta} U_i * \left(\left(1 + COS\left(2\pi * \overline{\Delta} t_i * k\right)\right) / 2 \right), \; IF \; U_{i-1} < U_i \\ U_{i-1} - \overline{\Delta} U_i * \left(\left(1 + COS\left(\pi * \overline{\Delta} t_i * k\right)\right) / 2 \right), \; IF \; U_{i-1} > U_i \end{cases}, \forall \; k = [1, \ldots, K] \qquad (3);$$

10. **End Data** – If all input data are processed, then go to step 11, if no go to step 3;
11. **STOP** – The end of the speech decoding algorithm.

The speech decoding algorithm was tested using approximation functions COS and SIN. The results were the same.

The results of the input signal reconstruction are shown on Figure 4. It can be noted that the speech decoding algorithm allows reproducing the initial signal at a high level of precision. The output speech signal quality is practically the same as the original one.

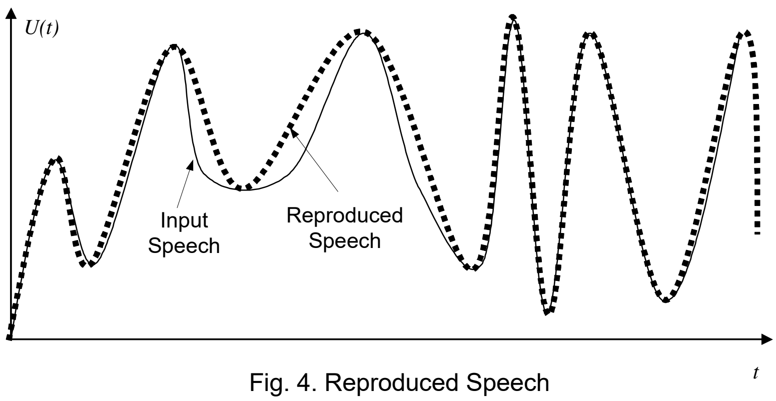

Fig. 4. Reproduced Speech

4. EXPERIMENTAL RESULTS

A code program was elaborated for proposed algorithms testing. The code program consists of two parts. First part is for speech encoding and the second one for speech decoding.

The data acquisition is made using standard port MIC IN at 92 kHz digitisation frequency. Encoding is processed in 8 bits words. Encoding program checks the condition $dU(t)/dt = 0$ using obtained digital codes and forms the table that contains the set of input signal's amplitudes in critical points $U(t)$ and the set of time periods Δt between them.

For testing proposed the program was trained with analogue signal band limited to between 100Hz and 10 kHz. The obtained results were written in a binary file which capacity depends linearly on the input signal frequency. For each period of the waveform input signal two critical points are determined that are characterized by two values: $U(t)$ and Δt. So if for each value 1 byte is needed at 100 Hz frequency 400 bytes is necessary and at 10 kHz the capacity is 40000 bytes.

Speech decoding program read the binary file and processed it according to formula (3). The results were transmitting to standard port LINE OUT that performs the conversion of digital signals in sound.

Reconstructed accuracy approaches 90% when the input frequency is low and 96% when this frequency is high.

5. CONCLUSIONS

In this paper a method for speech coding is proposed. The analysed algorithms allow obtaining a good quality for speech signal coding and decoding. The proposed encoder and decoder algorithms are easy to implement. The reproduced speech practically does not differ from the initial one. The efficiency of the implemented method is not so good if the input speech signal has a frequency larger then 15000 Hz because in this case the data compression coefficient decreases.

In our further researches we intend to implement the proposed circuits in reconfigurable structures that will allow more flexible adaptation of the coding system to local communication standards.

The coding and decoding algorithms can be utilized in both mobile phone communication and fixed phone communication.

REFERENCES

[1] Proakis J.G., Manolakis D.G., *Digital Signal Processing: Principles, Algorithms, and Applications*, 3rd ed, Prentice Hall, New Jersey, 1996.

[2] Makhoul J., ”Linear Prediction: A Tutorial Review”, *Proc. IEEE*, vol. 63(4), pp. 561-580, April 1975.

[3]. ITU Recommendation G.711. Pulse Code Modulation of Voice Frequencies, 1988.

[4]. ITU Recommendation G.726. 40, 32, 24, 16 kbit/s Adaptive Differential Pulse Code Modulation, March 1991.

[5]. ITU Recommendation G.727. 5-,4-,3- and 2 bits Sample Embedded Adaptive Differential Pulse Code Modulation, November 1994.

[6]. ITU Recommendation G.728. Coding of Speech at 16 kbit/s Using Low-Delay Code Excited Linear Prediction, November 1994.

[7]. John Makhoul, Samil Roucos, Herbert Gish, *Vector Quantization in Speech coding:* IEEE v. 73, No 11, 1985.