ISCA Archive
http://www.isca-speech.org/archive

SPECOM'2004: 9th Conference
Speech and Computer
St. Petersburg, Russia
September 20-22, 2004

Parameters quantization in sinusoidal speech coder on basis of human auditory model

Denis S. Likhachov (1), Alexander A. Petrovsky (2)

(1) The Belarusian State University of Informatics and Radioelectronics, 6, P.Brovki st., Minsk, Belarus, 220027, *den2000@tut.by*
(2) The Belarusian State University of Informatics and Radioelectronics, 6, P.Brovki st., Minsk, Belarus, 220027, *palex@it.org.by*

Abstract

In the given paper a method of the parameter quantization in speech coder on basis of human auditory model is presented. Output speech parameters of the coder are three sinusoidal parameters: amplitude, frequency and phase. For amplitude and frequency quantization we propose to use a vector quantization, and for phases – scalar quantization. This parameter quantization method allows to achieve bit rate from 3 to 8 kbps depending on the reconstructed speech quality.

1. Introduction

The given below speech encoder on basis of human auditory model has a simple structure and relatively small computational complexity [1], [2]. Compared with the conventional speech coding systems, this encoder has also simple algorithmic realization and does not require making voice/unvoiced decision and pitch estimation during the speech analysis. Therefore, it is less sensitive to a background noise and to changing a speaker than conventional speech encoders.

Using of this speech encoder allows to obtain a good quality of the reconstructed speech on condition that speech parameters are not quantized.

So, for transmission through line of communication these speech parameters should be correspondingly quantized and coded.

2. A sinusoidal speech coder on basis of human auditory model

The concerned coder system is based on a sinusoidal speech representation. According to this conception both voiced and unvoiced speech components are represented by a set of the sinusoidal waves [3], [4]. The approach considered here also involves main ideas from papers [5], [6].

Speech coding process can be represented as shown in Figure 1. Sampled input speech signal s(n) is analyzed by Hamming window W(n) periodically. Peak selection is produced so that to achieve a high perceptual quality of the reconstructed speech signal. The most significant for human hearing peaks are selected using histogram G(k) and spectrum |S(k)|. Frequencies and amplitudes of the sinusoidal components are obtained by peak position and peak value correspondingly. Sinusoidal phases are calculated by real and imaginary parts of spectrum |S(k)| at respective frequencies.

So, in order to reduce necessary information for coding and improve a perceptual quality of the output speech, two human auditory system models are used in this system. The first model is represented by a bank of band-pass filters that imitate functioning of the human ear cochlear [2].

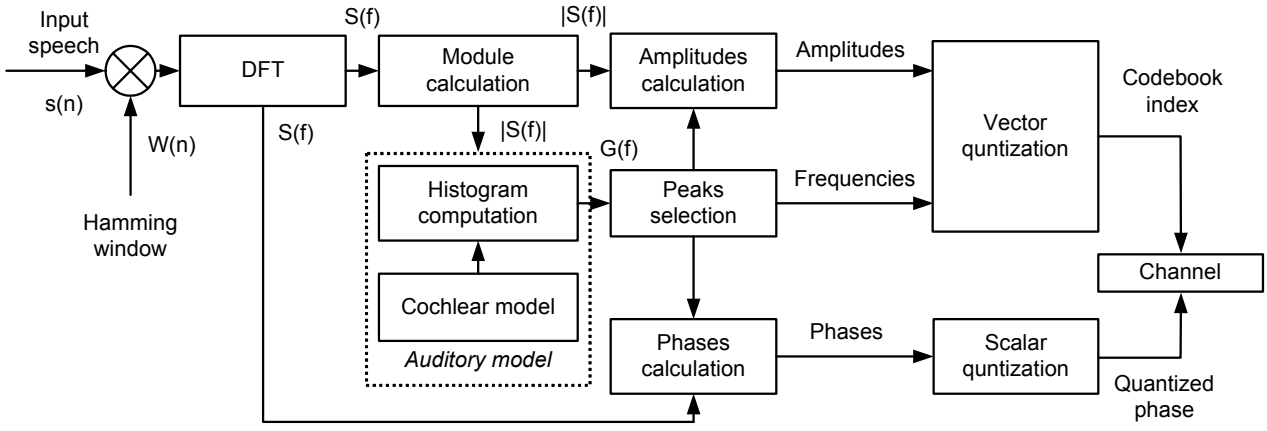

Figure 1. Speech analysis scheme

An example of amplitude-frequency responses for 32 cochlear filters are depicted by Figure 2. It is possible to find a cochlear map which relates the frequencies of cochlear filters with basilar membrane locations and with 3dB bandwidth characteristics – Figure 3. Horizontal coordinates denote frequencies of cochlear filters, and vertical coordinates show 3dB filter bandwidths.

The second model represents human auditory system at the auditory nerve level. It is based on calculation of the Ensemble Interval Histogram (EIH), which was proposed by Ghitza in papers [7]–[9]. The EIH technique allows selecting the more important, dominating sinusoidal components and enhances the perceptual quality of the synthesized speech. Experimental results demonstrate that the reconstructed signal retains most of the intelligibility and cleanness of the original speech even in case of limitation dominating sinusoidal components up to 8-10 items [1], [5]–[7].

However, because of high computational complexity the original approach [7] is hardly applicable in real time tasks, such as speech coding or speech recognition. Therefore, we propose a simplified version of the histogram computation. It is shown schematically in Figure 4. In contrast to Ghitza approach the signal processing is made in frequency domain mainly. An example of the calculation for one histogram bin is presented in Figure 5. This approach is similar to average localized synchrony detection method [10]. The proposed peak selection algorithm is described below.

Input data of the spectral peak selection algorithm are following:

N_F – length of the Fourier transform;

$|S(k)|$ – speech signal spectrum, $k = \overline{1, N_F / 2}$;

$H(k)$ – amplitude-frequency responses of the cochlear filters;

M_F – a number of all cochlear filters;

L_S – a specified number of all sinusoidal components;

L_P – a number of all spectral peaks;

$Ind_P(l)$ – array with indices of all spectral peaks from $|S(k)|$, $l = \overline{1, L_P}$;

Lv – array with level values in decibels, Lv = {10, 14, 18, 22, 26, 30, 34, 38, 42, 46, 50, 54}

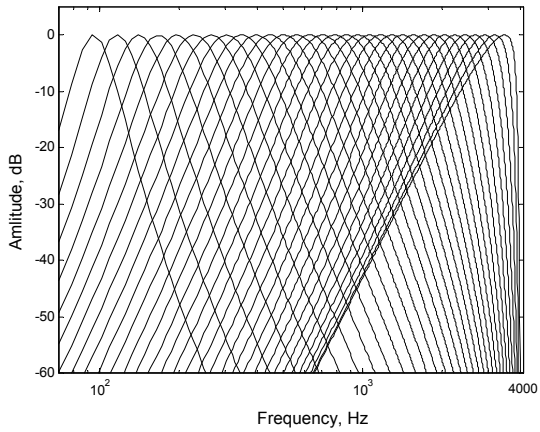

Figure 2. Amplitude-frequency responses for 32 cochlear filters

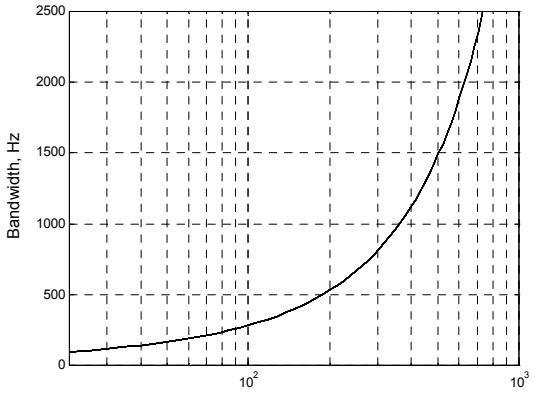

Figure 3. 3dB-bandwidth characteristics of the cochlear filter bank

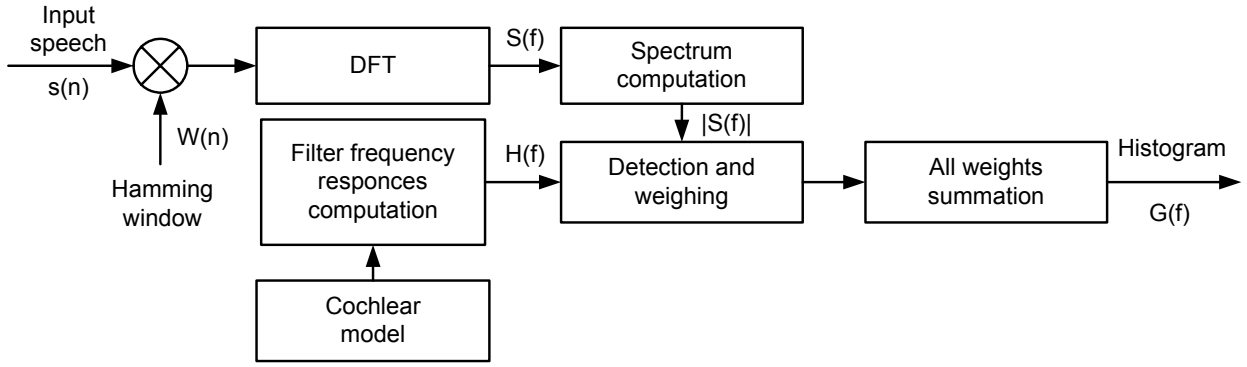

Figure 4. A simplified histogram computation

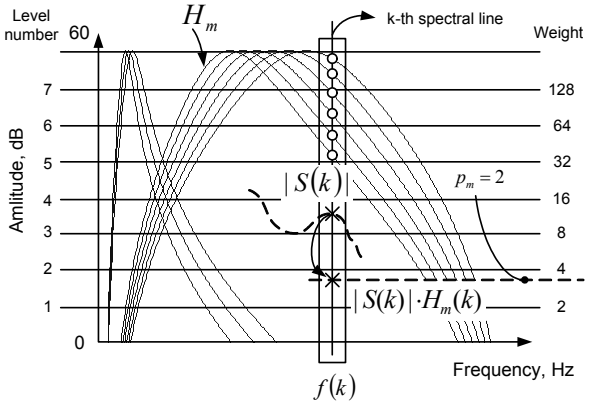

Figure 5. Calculation of a histogram bin

Peak selection algorithm can be described by the following steps:

Step 1. Array D_m is calculated by following expression:

$$D_m\left(Ind_P(l)\right) = \left|S\left(Ind_P(l)\right)\right| \cdot H_m\left(Ind_P(l)\right), \quad l = \overline{1, L_P}, \ m = \overline{1, M}_F, \tag{1}$$

where m – currently processed cochlear channel number; M_F – a number of all cochlear filters; L_P – a number of all spectral peaks; $Ind_P(l)$ – array with indexes of the obtained spectral peaks from array $|S(k)|$; $H_m(Ind_P(l))$ – amplitude-frequency response of the m-th cochlear filter in frequency position $Ind_P(l)$, $l = \overline{1, L_P}$; $\left|S\left(Ind_P(l)\right)\right|$ – value of speech signal spectrum in frequency position $Ind_P(l)$.

Step 2. Weighting coefficients P_m^l for l-th spectral peak and m-th cochlear channel are calculated by following rule:

$$\begin{aligned} &\text{if } D_m\left(Ind_P(l)\right) > Lv(i) \text{ and} \\ &D_m\left(Ind_P(l)\right) < Lv(i-1), \ i = \overline{1, N_U} \\ &\quad \text{then } P_m^l = 2^i, \ l = \overline{1, L_P}, \end{aligned} \tag{2}$$

where N_U – a number of elements in array Lv; $Lv(i)$ is i-th element of level values array.

Step 3. Array of histogram elements $G_m\left(Ind_P(l)\right)$ for l-th spectral peak and m-th cochlear channel is computed by following formula:

$$\begin{aligned} &G_m\left(Ind_P(l)\right) = \\ &= \left|S(Ind_P(l))\right| \cdot H_m(Ind_P(l)) \cdot P_m^l = \\ &\quad = D_m\left(Ind_P(l)\right) \cdot P_m^l, \end{aligned} \tag{3}$$

where P_m^l – weighting coefficient for l-th spectral peak and m-th cochlear channel.

Step 4. Histogram $G\left(Ind_P(l)\right)$ in position $Ind_P(l)$ is computed by following expression:

$$G\left(Ind_P(l)\right) = \sum_{m=1}^{M_F} G_m\left(Ind_P(l)\right), \ l = \overline{1, L_P}. \tag{4}$$

Step 5. L_S spectral peaks are selected at frequencies that correspond to L_S largest values in the histogram $G\left(Ind_P(l)\right)$.

An example of the histogram G(k) for a speech signal frame is presented in Figure 6. X-direction – frequencies from 0 to 4000 Hz (sampling frequency is 8000 Hz). Y-direction – calculated weights (the bigger is an element histogram weight – the more important is its role in human speech recognition).

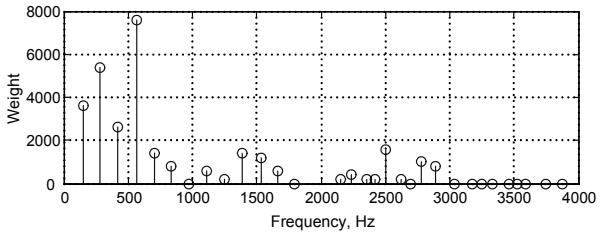

Figure 6. A histogram for one speech signal frame

Selected spectral peak locations for one speech signal frame are presented in Figure 7 (circles).

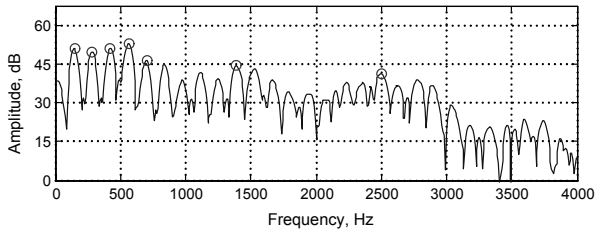

Figure 7. Selected spectral peaks

Speech synthesis procedure presumes generating sinusoids according to frequency and phase parameters, weighed by amplitude and then summed to produce a frame of synthesized speech as shown in Figure 8. The resulting speech signal is formed by adding each frame of the synthesized speech.

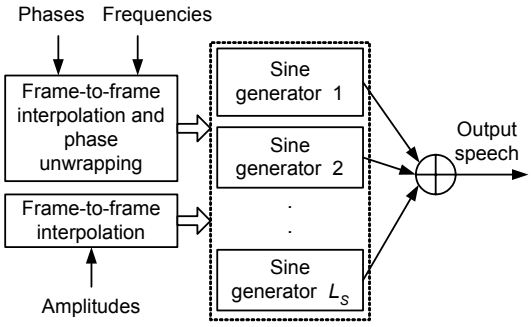

Figure 8. Speech synthesis scheme

In order to achieve good output speech quality, the speech parameters obtained as described above should then be matched and smoothed between the frames. A technique from papers [3], [4] is applied at this stage.

As an example a piece of the original speech signal in time domain is presented in Figure 9 (sampling frequency is 8000 Hz, male voice) and his spectrogram – in Figure 10. A piece of the synthetic speech signal is presented in Figure 11 and his spectrogram – in Figure 12. Following parameters were used for speech signal analysis: length of the Fourier transform N_F=1024, a number of all cochlear filters M_F=32, a number of sinusoids is 7, analyzing window length is 32 ms, frame length is 20 ms.

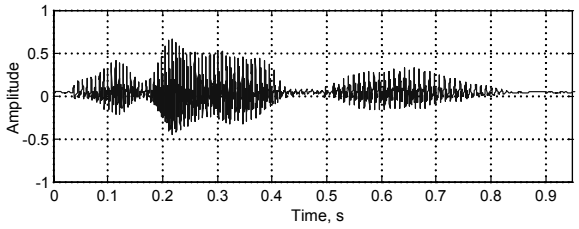

Figure 9. The original speech signal

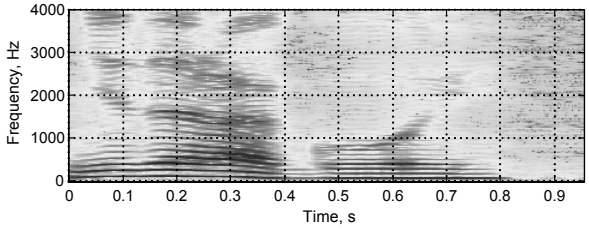

Figure 10. The original speech signal spectrogram

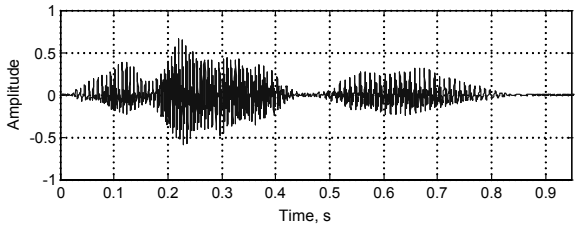

Figure 11. The synthetic speech signal

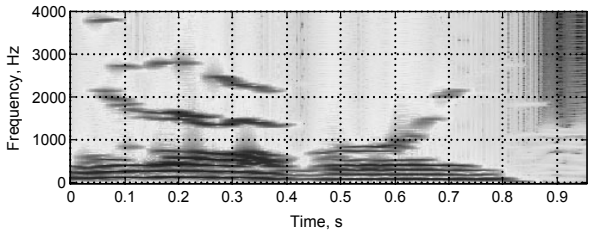

Figure 12. The synthetic speech signal spectrogram

Experiment results show that the synthesized output signal retains most of the clearness of the original speech. It has good legibility and allows to make the speaker recognition.

3. Amplitudes and frequencies quantization

We propose to use following parameter quantization approach. Since amplitudes and frequencies are obtained by speech spectrum than vector quantization in the amplitude-frequency space is proposed for their coded.

In the concerned coder system amplitudes are integer values and lie in the range from -32768 to 32768. In this case it requires 15 bit plus one sign bit for amplitude coding. So, dynamic range of the input speech signal D_{in} is

$$D_{in} = 20 \cdot \log_{10} \frac{2^{15}}{1} \approx 90 \ dB . \qquad (5)$$

Thus, in order to reduce the dynamic range, amplitudes are coded in the logarithmic range from 0 to 60 dB with equal step. It is quite enough for coding of the speech signal with good quality [11]. Logarithmic amplitudes are calculated by following formula:

$$A_{dB} = 20 \cdot \log_{10}(A_{Int}) - 30 , \qquad (6)$$

where A_{dB} – amplitudes in the logarithmic range in decibels; A_{Int} – integer amplitudes.

In our system frequencies also obtained by speech spectrum and have following possible values:

$$f = k \cdot \frac{F_S}{N_F}, \quad k = \overline{1, \frac{N_F}{2}}, \qquad (7)$$

where F_S – sample frequency; N_F – length of the discrete Fourier transform; k – frequency index.

So, it is necessary to transmit only frequency index k. For example, if the length of the discrete Fourier transform N_F=1024, then k has bit capacity of 9.

Taking into consideration the speech signal properties [12] the frequency may be coded only in the range from 40 to 3800 Hz (toll-quality speech).

However, proposed parameter quantization approach requires equal step of amplitudes and frequencies quantization. So, before quantization the amplitudes are normalized by following expression:

$$A = A_{dB} \cdot \frac{k_{\max}}{A_{\max}} = A_{dB} \cdot \frac{N_F}{2 \cdot A_{\max}}, \qquad (8)$$

where $k_{\max}$ – a highest possible value of the frequency index (if length of the discrete Fourier transform N_F =1024, than $k_{\max} = 512$); $A_{\max}$ – a highest possible value of the amplitude, in this case $A_{\max} = 60$.

A process of the parameter quantization can be described as finding from a codebook C of predetermined parameter vectors y_i , the vector that "matches" best the set of parameters of one sinusoidal component computed for a current frame of speech. When the codevector is found, its index is transmitted to the decoder which contains the replica of the quantization codebook, as illustrated in Figure 13

Input vector of the parameters for one sinusoid is defined as

$$x = \{A, k\}, \qquad (9)$$

where A – amplitude of the sinusoid; k – frequency index of the sinusoid.

Codebook is a set of L predetermined output vectors y_i:

$$C = \{y_i\}, \quad i = \overline{1, L}, \qquad (10)$$

where

$$y_i = \{A_i, k_i\}. \qquad (11)$$

Generally, quantization error (vector distortion) is defined as mean-square error [13]. In our case quantization error $d(x, y)$ is calculated by following formula:

$$d(x, y) = \frac{1}{2}\left(A^x - A^y\right)^2 + \frac{1}{2}\left(k^x - k^y\right)^2, \qquad (12)$$

where x – input vector; y – vector from the codebook; A^x, A^y – amplitudes of the sinusoid from the input vector and nearest codebook vector correspondingly; k^x, k^y – frequency indices of the sinusoid from the input vector and nearest codebook vector correspondingly.

The nearest neighbor condition is used for searching optimal element from the codebook according to the condition:

$$\begin{aligned} &q(x) = y_i \\ &only \quad if \quad d(x, y_i) \le d(x, y_j), \\ &j \ne i, \quad 1 \le j \le L, \end{aligned} \qquad (13)$$

where $q(\cdot)$ – quantization operator.

As experimental results show the described above speech parameter quantization approach gives a good results if codebook length equals 4096 and larger. However, in that case the quantization algorithm has large computational complexity that creates difficulties for using of this method in real-time systems. Also if the codebook length is very large there are some complexities in the codebook training process.

If codebook with smaller size is used, than there are noticeable distortions in the reconstructed speech signal. It is determined, that the frequency distortion extremely affects the synthetic speech quality.

That is why we propose combining vector and scalar quantization for frequencies coding at that stage.

The frequency correction process is presented in Figure 14. Similar idea was described in paper [5].

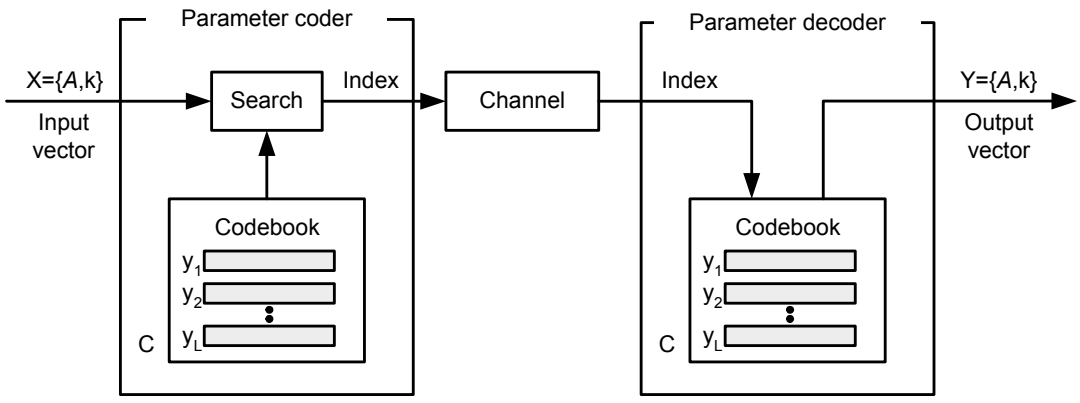

Figure 13. Vector quantization diagram

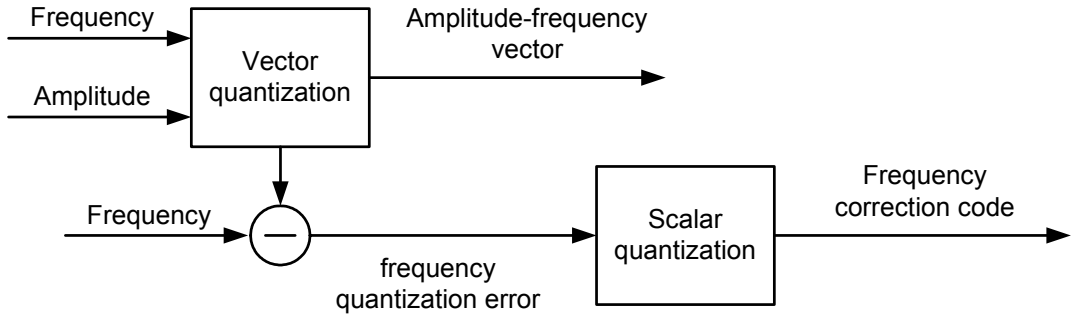

Figure 14. Frequency correction scheme

4. Codebook design

For codebook design Generalized Lloyd Algorithm is used [13], [14]. This algorithm divides a training set of vectors $\{x_n\}$, $n=\overline{1,P}$ into L codebook clusters R_i, $i=\overline{1,L}$. Below m is an iteration number, $R_i(m)$ – i-th cluster for m-th iteration with centroid $y_i(m)$.

Step 1. The entry value assignment. Start with an initial codebook $C(0)=\{y_i(0)\}$. Let m=0.

Step 2. Classification. Classify the training set into cluster sets $R_i(m)$ using the nearest neighbor condition:

$$\begin{aligned} &x \in R_i(m) \\ &only \quad if \quad d(x, y_i(m)) \le d(x, y_j(m)), \\ &j \ne i, \quad 1 \le j \le L. \end{aligned} \tag{14}$$

Step 3. Codebook vector correction. Let $m \leftarrow m+1$. Using the centroid condition, compute the centroids for the cluster sets to obtain a new codebook $C(m)=\{y_i(m)\}$ by following formula:

$$y_i(m) = cent(R_i(m)), \ i=\overline{1,L}. \tag{15}$$

Since the squared error distortion measure is used, centroids $y_i(m)$ for clusters $R_i(m)$ are centers of mass of this clusters.

If an empty cell was generated in the previous step, an alternate code vector assignment is made for that cell. New vector is created in place of the computed centroid.

Step 4. Compute the average distortion $D(m)$ for new codebook $C(m)$ by following expression:

$$D(m) = \frac{1}{L}\sum_{i=1}^{L} D_i^r(m), \tag{16}$$

where $D_i^r(m)$ – average distortion for r-th cluster and it is computed by following formula:

$$D_i^r(m) = \frac{1}{M}\sum_{x \in R_i} d(x, y_i), \tag{17}$$

where M – a number of vectors in the current cluster.

If the average distortion $D(m)$ changes are negligible in comparison with the last iteration, stop. Otherwise, go to Step 2.

So, each iteration of the algorithm monotonically decreases or keeps unchanged the average distortion of a vector quantizer. It is possible to prove, that aforecited algorithm is converged to local optimum [14]–[16].

In codebook design by proposed method, two critical factors are very important: size of the training set and a number of algorithm iterations.

A small set of training vectors will not approximate the statistical characteristics of the input vector sequence and will not give a good vector quantizer. For good result a size of the training set P should be sufficiently large. According to papers [13], [14] the ratio of training set vectors P to the number of codebook elements L should be above 50 and less than 200.

Considering a number of algorithm iterations it should be taken into account, that overly high trained codebook does not give an advantage since it will perform large distortion when it is used with other input vector.

The initial codebook is produced by random coding. The main idea of this approach is to select randomly L vectors from the training set.

The training set of vectors was created by analyzing speech fragments with various phrases and voices (male, female, child ones). For training purposes the vectors with nonzero amplitudes were used.

An example of the designed codebook with length of 256 elements is presented in Figure 15. Elements of the codebook are represented by circles. First codebook vector contains zero amplitude and frequency index k=1. The average distortion for this codebook is 16.4667.

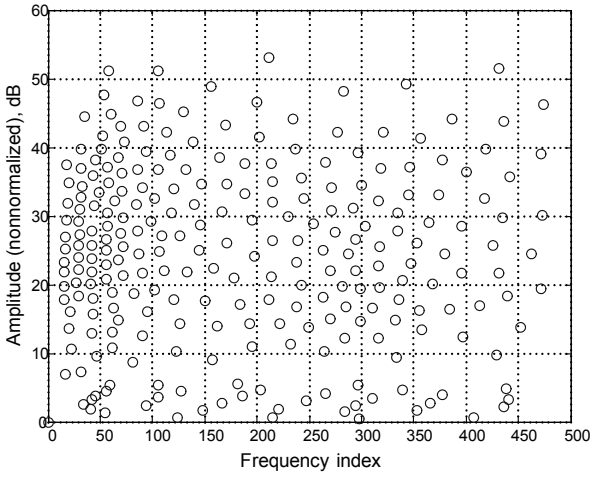

Figure 15. An example of the codebook with 256 elements

5. Phases quantization

Phases are quantized using scalar quantization. In view of the trigonometric function periodicity phases may be quantized in the range from $-\pi$ to π. 2-3 bits are enough for achieving good results under this approach.

6. Experimental results

Described above parameter coding methods were tested in Matlab. The speech signal showed in Figure 9 was used. The results of these tests for a typical speech frame are presented in Figures 16–19. As it is described above the speech signal is analyzed with speed of 50 Hz. A number of sinusoidal components vary from 5 to 10 items per frame depending on the necessary speech quality. Also the average mean-square quantization error D_q for every frame was computed.

So, the following experiments were made.

Firstly, scalar quantization of the sinusoidal parameters was made – Figure 16. In this case it is used 8 bit for amplitude coding, 9 bit for frequency coding and 3 bit for phase coding. The reconstructed speech with parameter quantization has a good quality and is very close to the reconstructed speech without parameter quantization. But under such conditions 20 bits are required for one sinusoidal coding and bit rate is from 5 to 10 kbps. The average mean-square quantization error D_q=2.0753e+006.

Secondly, two dimensional vector quantization of the amplitude and frequency was made – Figure 17. The codebook length is 4096. This approach for good result requires 12 bit for codebook index coding and 3 bit for scalar quantization of phases. For codebook design Generalized Lloyd Algorithm was used. The reconstructed speech approaches to speech quality in case of scalar quantization. In this case bit rate is from 3.75 to 7.5 kbps, but the algorithm has large computational complexity. D_q=2.2537e+006.

Thirdly, two dimensional vector quantization with frequency correction was used – Figure 18. Experiments show, that good results are obtained when codebook length equals 256 (8 bits), for frequency correction it is used 4 bits and scalar quantization of phases – 3 bits. For codebook design Generalized Lloyd Algorithm was used as well. The reconstructed speech also has a good quality, but from time to time there are some tonal artifacts. In this case bit rate is also from 3.75 to 7.5 kbps, but computational complexity is acceptable. D_q=4.4921e+006.

If frequency correction is not used, than there are large frequency distortion – Figure 19. It makes worse the reconstructed speech quality and quantization error is large, D_q=6.3887e+006.

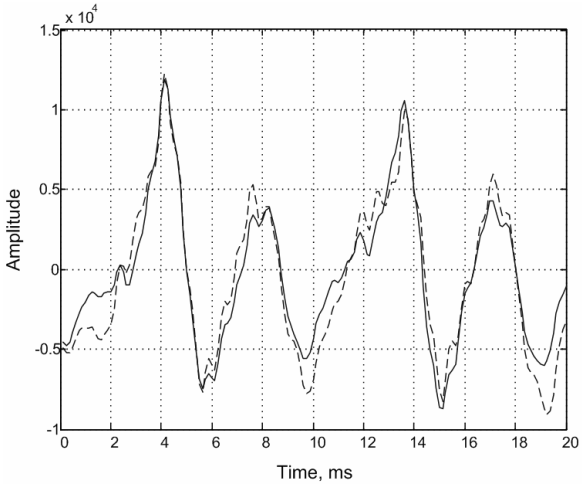

Figure 16. A frame of the reconstructed speech signal without parameter quantization (solid line) and with scalar quantization of parameters (dashed line)

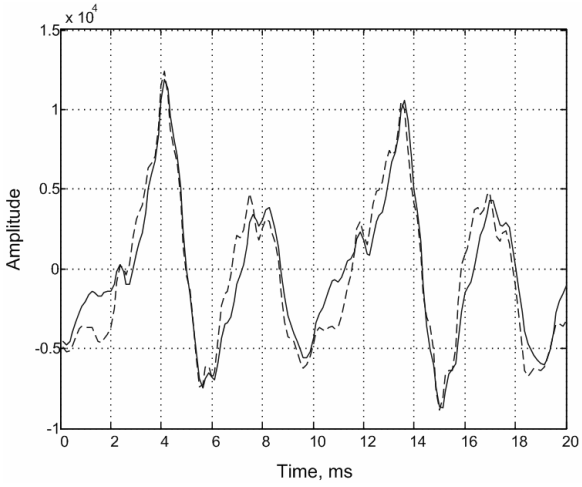

Figure 17. A frame of the reconstructed speech signal without parameter quantization (solid line) and with vector quantization (length of the codebook equals 4096) of parameters (dashed line)

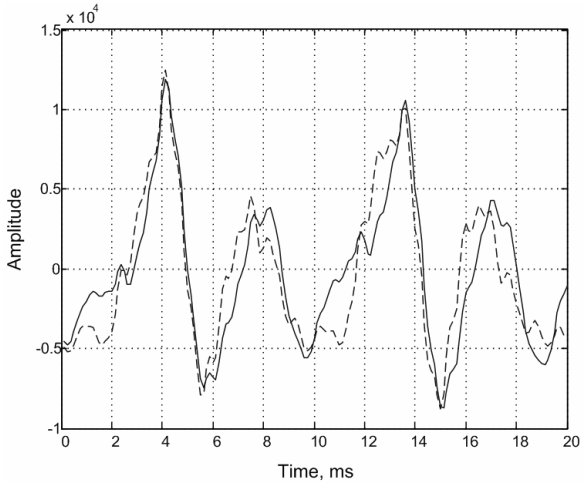

Figure 18. A frame of the reconstructed speech signal without parameter quantization (solid line) and with vector quantization (with frequency correction, length of the codebook equals 256) of parameters (dashed line)

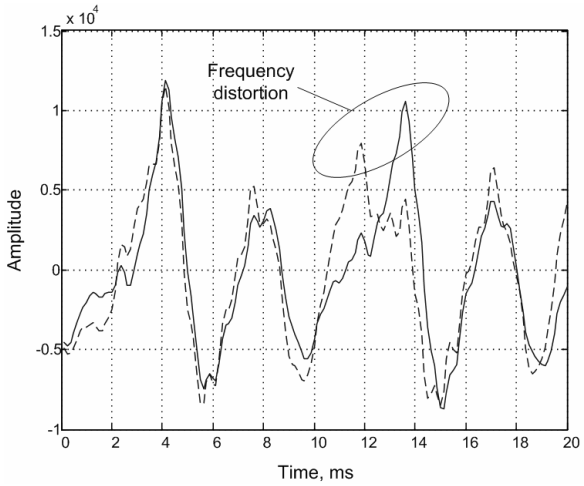

Figure 19. A frame of the reconstructed speech signal without parameter quantization (solid line) and with vector quantization (without frequency correction, length of the codebook equals 256) of parameters (dashed line)

Generally, these experimental results demonstrate, that using of the proposed parameter quantization approach allows to achieve bit rate from 3.7 to 7.5 kbps depending on the speech quality.

7. References

[1] Likhachov, D.S., Petrovsky, A.A., "Improved auditory-based speech coding using psychoacoustic model based on a cochlear filter bank and an average localized synchrony detection", CISIM'2003, Elk, Poland, 11-19, 2003.

[2] Petrovsky, A.A., Likhachov, D.S., "A digital cochlear model as a base of anthropomorphic speech processing", ICNNAI'2003, Minsk, Belarus, 126-131, 2003.

[3] McAulay, R.J., Quatieri, T.F., "Speech analysis/synthesis based on a sinusoidal representation", IEEE Trans., Acoust., Speech, Signal Process., vol. ASSP-34, 744-754 (1988).

[4] McAulay, R.J., Quatieri, T.F., "Low-rate speech coding based on the sinusoidal model", Advances in Speech Signal Processing, S. Furui, M.M. Sondhi, Eds., New York: Marcel Dekker, 1992, pp. 165-208.

[5] Wan, W., Au, O.C., Keung, C.L., Yim, C.H., "A novel approach of low bit-rate speech coding based on sinusoidal representation and auditory model", EUROSPEECH'99, 1555-1558, 1999.

[6] Au, O.C., Wan, W., Keung, C.L., Yim, C.H., "Sinusoidal representation and auditory model-based parametric matching and smoothing and its application in speech analysis/synthesis", EUROSPEECH'99, 2287 – 2290, 1999.

[7] Chitza, O., "Auditory Nerve Representation Criteria for Speech Analysis/Synthesis", IEEE Trans. Speech and Audio Process., vol. ASSP-35, no.6, 736-740 (1987 June).

[8] Chitza, O., "Auditory Nerve Representation as a Basis for Speech Processing", Advances in Speech Signal Processing, S. Furui, M.M. Sondhi, Eds., New York: Marcel Dekker, 1992, pp. 453-485.

[9] Chitza, O., "Auditory Models and Human Performance in Tasks Related to Speech Coding and Speech Recognition", IEEE Trans. Speech and Audio Process., vol.2, no.1, part II, 115-132 (1994).

[10] Ali, A.M.A., "Robust auditory-based speech processing using the average localized synchrony detection", IEEE Trans. Speech and Audio Process., vol.10, no.5, 279-292 (2002 July).

[11] Kondoz, A.M., Digital Speech Coding for Low Bit Rate Communications Systems, UK, University of Surrey: John Wiley & Sons, 1996.

[12] Rabiner, L.R., and Schafer, R., Digital Processing of Speech Signals, Englewood Cliffs, New Jersey: Prentice-Hall, 1979.

[13] Makhoul, J., Roucos, S., and Gish, H., "Vector quantization in speech coding", Proc. IEEE, vol. 73, 1551-1588 (1985 Nov.).

[14] Linde, Y., Buzo, A., and Gray, R.M., "An algorithm for vector quantizer design", IEEE Trans. Commun., vol. COM-28, no.1, 84-95 (1980 Jan.).

[15] Anderberg, M.R., Cluster Analysis for Applications, New York, NY: Academic Press, 1973.

[16] Gersho, A., and Gray, R.M., Vector Quantization and Signal Compression, Boston: Kluwer Academic Press, 1992.