

Development of a Sound Coding Algorithm for a Cochlear Implantation System

E. M. Gluhovsky^{a,*}, A. I. Egorov^{a,**}, and M. I. Karapetyants^{a,***}

^a Medical Engineering Laboratory, Moscow Institute of Physics and Technology (MIPT),
Dolgoprudnyi, Moscow oblast, 141700 Russia

*e-mail: glukhovskii.em@mipt.ru

**e-mail: egorov.aleksei.i@mipt.ru

***e-mail: yanturina.mi@phystech.edu

Received November 11, 2020; revised January 27, 2021; accepted January 29, 2021

Abstract—A stimulation strategy has been developed for a new design of the cochlear implantation system that is being created by the MIPT. A model has been constructed that gives some idea of how a person with a cochlear implant perceives sound processed using this strategy. The strategy is assessed by comparing the sound obtained using the inverse transformation to the original sound. Many individual physiological factors, such as cross-stimulation of adjacent channels and the state of individual nerve endings, are difficult to take into account. Nevertheless, intelligible speech reproduction is provided when information is transmitted by stimulating the cochlear zone using our strategy with the described assumptions.

DOI: 10.1134/S0020441221040059

1. INTRODUCTION

In this paper, we describe the basic principles of operation of well-known sound coding algorithms for stimulating the cochlear zone that allows patients with sensorineural hearing loss to perceive sound. Existing methods of sound processing are reviewed and an explanation is provided for determining the frequency of electrode stimulation.

The aim of our study was to simulate the sound coding algorithm for a new device called a cochlear implantation system and to assess the quality of its operation using the constructed model. We developed this device in order to replace such imported analogues as Cochlear, Med-El, and Advanced Bionics, which are all popular in our country.

We have developed and presented a new stimulation strategy for the cochlear implantation system that optimizes the computational efforts for sound processing; its main advantages have been demonstrated. The main stages of the sound coding algorithm are described. The quality of the algorithm was assessed by ear and the algorithm was simulated in the MATLAB system. This paper provides a visual comparison of the original sound and the sound perceived by the patient, as well as an analysis of the results. Amplitude spectra of the received signals are constructed, a diagram of the developed algorithm is shown, and graphs of the initial and final signals, as well as of the signal at intermediate stages of its processing, are constructed.

The cochlear implantation system technology, which helps hearing-impaired persons, requires the use of dedicated sound coding algorithms called stimulation strategies. Various methods for stimulating the cochlear receptors are known (CIS, ACE, HiRes, SPEAK, etc.). Only general information is known about the principles of their operation, since developments in this area are protected by the copyright of the device manufacturers.

As was the case with the CIS strategy, filtering was used in our strategy at the initial stage of processing. The Goertzel recursive algorithm, nonlinear compression, and a lowpass filter were then applied. The advantage of replacing the fast Fourier transform with the Goertzel algorithm to save computational resources was shown while analyzing the number of steps required for use of a particular method.

It is clear that comfortable use of the algorithm in the device requires minimum computational resources with the least loss of the quality of the transmitted sound. In this paper, we analyze various stages of the algorithm, as well as their advantages and disadvantages, and provide opportunities for improvement.

2. HUMAN PERCEPTION OF SOUND: THE DESIGN OF A COCHLEAR IMPLANTATION SYSTEM

The eardrum transmits sound pulses to the middle ear, where vibrations are transmitted to the cochlea in

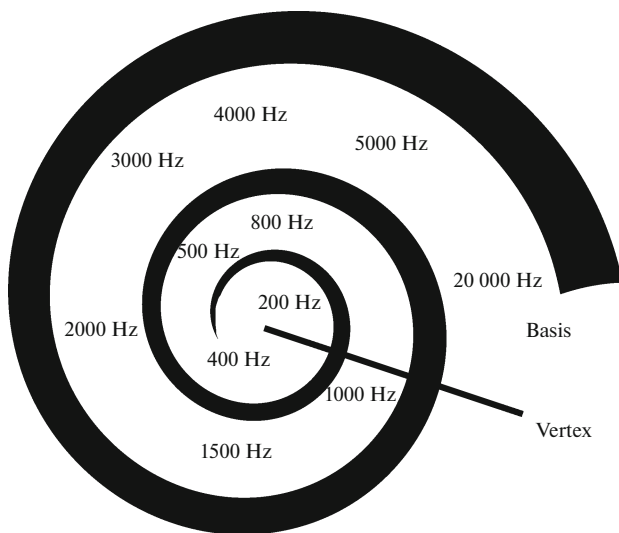


Fig. 1. The distribution of frequencies along the depth inside the cochlea.

the inner ear via the auditory bones (the malleus and incus). On the basilar membrane in the cochlea are hair cells, which are receptors of the auditory system. Vibrations are transmitted to the receptor cells through the fluid that fills the cochlea. These cells convert a mechanical pulse into an electrical pulse and it is perceived by nerves.

Damage to hair cells may cause hearing loss. In this case, a cochlear implantation system can be used. It consists of a microphone and a speech processor. Sound travels through a coil placed under the skin to an array of electrodes located on the cochlea. The signal is then transmitted to the brain by stimulating the auditory nerves.

The cochlear implantation system consists of a cochlear implant and a wearable speech processor. The cochlear implant is a device used in the case of damage to the inner ear if functioning of the spiral ganglion neurons is preserved. The speech processor is equipped with a microphone, a sound processing device, and an inductor for transmitting energy and data to the implant. The implant itself consists of a receiving coil, a stimulation unit, and an electrode matrix.

The speech processor transmits digitally encoded sound to the implant through the coil. The stimulator–receiver of the cochlear implant digitally converts sound into electrical signals and then transmits them to an electrode array using a certain stimulation strategy. The number of electrodes varies from manufacturer to manufacturer and can range from 12 to 22.

After the frequencies have been selected for the stimulation, they need to be correctly placed on the electrode so that they stimulate exactly at those points where the cochlear receptors for these frequencies are

located. For most people, they are located in the same way, as can be seen in Fig. 1.

The distribution of frequencies perceived by the ear on the cochlea is known for humans from experimental data and was approximated by the empirical formula [1]

$$FC_{GP}(x) = 0.1654 \text{ kHz} \left(\exp \left(\frac{0.13815(35 \text{ mm} - x)}{1 \text{ mm}} \right) - 1 \right).$$

3. VARIETIES OF STIMULATION STRATEGIES

The advanced combinational encoder (ACE) and continuous interleaved sampling (CIS) strategies are most similar to the stimulation strategy that we developed. Therefore, we consider them in more detail.

3.1. Description of the CIS Stimulation Strategy

The main stages of the CIS stimulation strategy are as follows (see Fig. 2). First, the signal is passed through bandpass filters that divide it into several frequencies. Each filter corresponds to its own electrode. This is followed by rectification and application of lowpass filters (usually 200 or 400 Hz). As a result, we obtain a low-frequency signal that resembles the envelope shape of the original signal. Next, this signal is compressed using a nonlinear function, e.g., a logarithm, to make sure that the electric signal, which will be further transmitted to the electrodes, falls into the signal range that is comfortable for a person. This range is set by a doctor. The amplitude ratios of the obtained electric pulses correspond to the nonlinearly compressed amplitudes of the envelopes in each stimulation channel. A total of 16 electrodes are used. The important point is that they are all stimulated in a single cycle, but not at the same time [2].

3.2. Description of the ACE Stimulation Strategy

Bandpass filters are applied to the input signal (see Fig. 3) in the ACE strategy, as well as in the CIS strategy. Envelopes are obtained in each channel after filtering. The energy is calculated for each bandwidth. Further, 6–10 frequency ranges for which the energy is maximal are selected in each cycle. The energy of the remaining ranges is not used for the stimulation. The pulses are then converted into an electric signal in a range that is comfortable for the patient to perceive and the selected electrodes are sequentially stimulated. Typically 22 electrodes are used and iteration of the cycle (selection of channels with the maximum energy) is repeated [2].

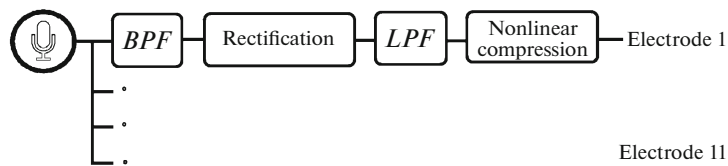


Fig. 2. The block diagram of the CIS stimulation strategy: (*BPF*) bandpass filter and (*LPF*) lowpass filter.

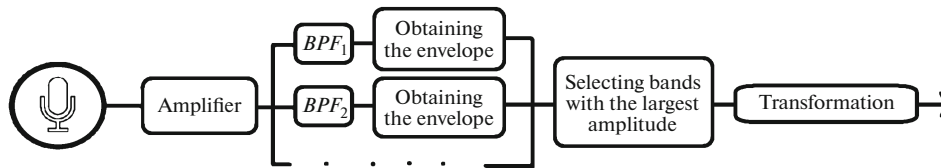


Fig. 3. The block diagram of the ACE stimulation strategy.

4. THE DEVELOPED STIMULATION STRATEGY

The diagram shown in Fig. 4 was simulated in the MATLAB software environment.

We consider the blocks of our algorithm in more detail. The initial part, i.e., filtering with bandpass filters, is similar to the CIS and ACE strategies. The bandpass filters split the signal into 16 channels. The use of these filters makes it possible to dispense with a window function that removes distortions caused by the edge effect in the Fourier transform calculations. Calculation of the discrete Fourier transform leads to discontinuities at the edges of the analyzed block, which manifest themselves by a distortion of the spectrum. This effect is less pronounced if a signal with a narrow spectrum is present in the analyzed block after applying a bandpass filter. The application of filters without the window function makes it possible to save significant computational resources and use recursive algorithms in the Fourier transform calculation. It is possible to use both serial and parallel stimulation of the electrodes at the output.

4.1. Input Bandpass Filters

The following frequencies were taken to split the signal into 16 channels [3]:

Channel no.	1	2	3	4	5	6	7	8
Frequency, Hz	333	450	540	642	762	906	1076	1278
Channel no.	9	10	11	12	13	14	15	16
Frequency, Hz	1518	1803	2142	2544	3022	3590	4264	6665

The nonlinear phase–frequency response of the filter may cause unwanted phase distortions of the signal during recovery; nevertheless, the phase does not play a role for correct human speech perception. Filters with finite-impulse response (FIR filters) that provide a linear phase–frequency response were ini-

tially selected for the most efficient processing of sound. Although the FIR filters are more computationally intensive, the sound produced by them is slightly better due to the lack of phase delays between signals, but this difference is not significant. Considering all of the above, as well as the fact that the human ear is not very susceptible to phase delays (they are more important at low frequencies), we decided to use filters with an infinite impulse response (IIR).

The reconstructed sound obtained using IIR filters was compared to the original sound (see graphs in Figs. 5a and 5b) in order to evaluate the efficiency of the developed algorithm. These graphs show that the original sound and the sound obtained after the use of our algorithm and the reconstruction differ only slightly.

4.2. The Goertzel Algorithm and the Modified Goertzel Algorithm

The Goertzel algorithm is a method for calculating the Fourier transform for a selected frequency. The standard Goertzel algorithm calculates the Fourier coefficients for a set of frequencies:

$$\{2\pi k/N\},$$

where $k = \overline{0, N-1}$, and N is the number of signal counts [4].

The computational structure of the Goertzel algorithm is similar to that of the IIR filter; therefore, it is often called the Goertzel filter. We compare it to the fast Fourier transform algorithm, which allows us to calculate the transform at specific points (e.g., we can choose the most convenient number of points). If we return to the above frequencies, we can see that the distance between the required frequencies is approximately 100 Hz. For a rough estimate to be obtained, the algorithm must be applied to approximately 160

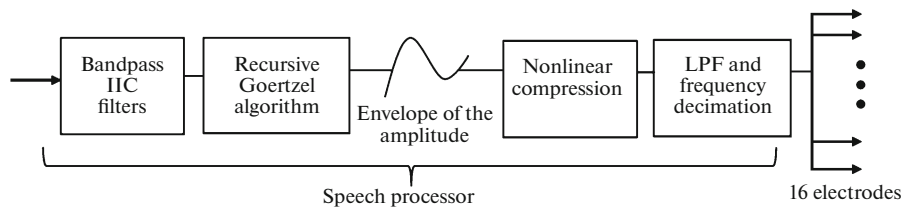


Fig. 4. The block diagram of the developed stimulation strategy.

points (the sampling frequency of 16 kHz/100 Hz) and even this fails to guarantee that we will definitely get the selected frequencies. Thus, the calculation for 160 points, obviously, will take more resources than the calculation for 16 points in the case of the Goertzel algorithm, e.g., when we calculate specific frequen-

cies. Therefore, the Goertzel algorithm is optimal, especially in view of the existence of its modification, i.e., the recursive algorithm. The diagram of the classical Goertzel filter is shown in Fig. 6. We initially consider its design and then its modifications [4]. The use of such a filter helps us to evaluate the k th bin (the

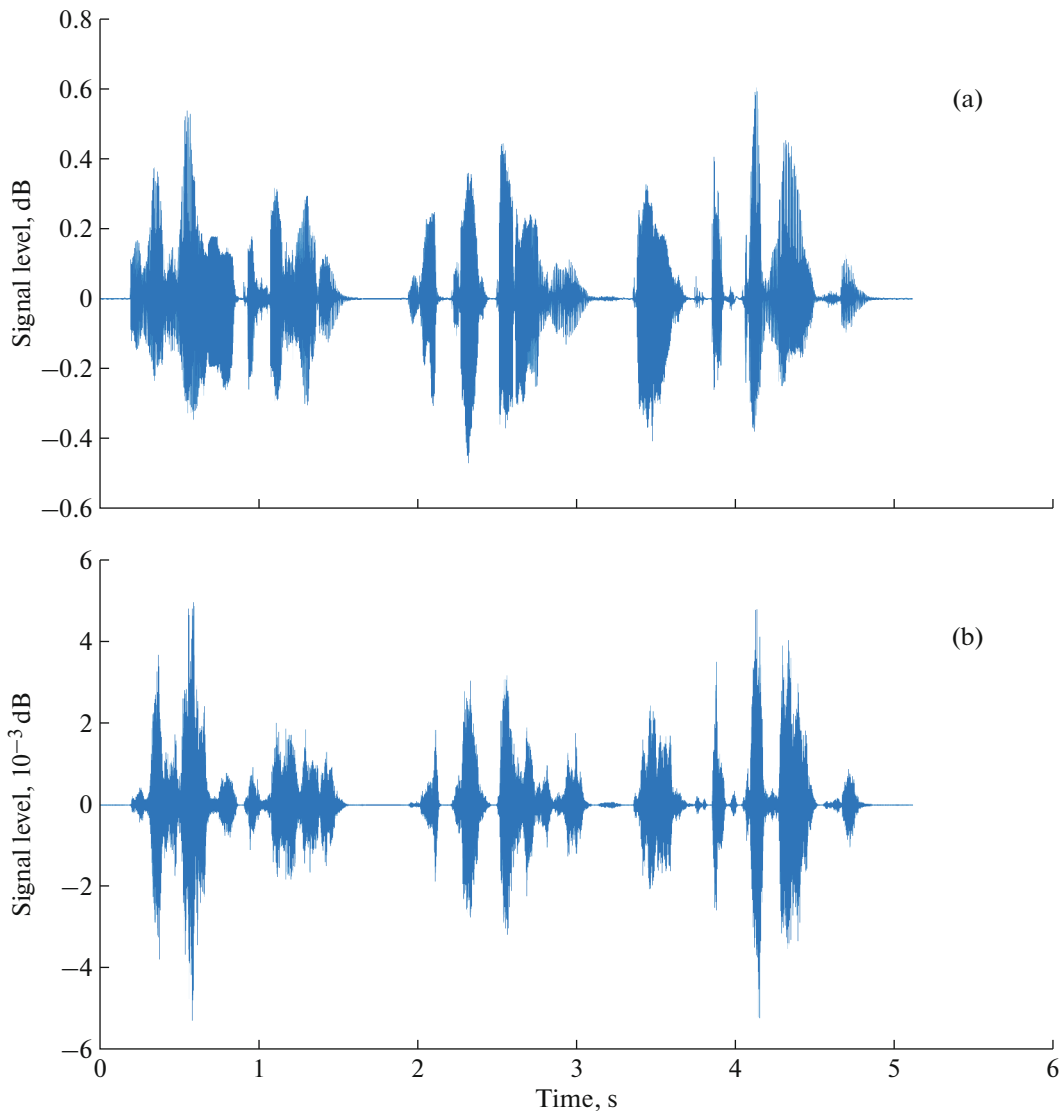


Fig. 5. (a) The original signal received by the microphone and (b) the sound signal after restoration (using the IIR filters).

fixed number of the spectral sample of the discrete Fourier transform) of the N -point discrete Fourier transform:

$$S_N(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn},$$

$$W_N^{kn} = \exp\left(-j\frac{2\pi}{N}kn\right),$$

where $x(n)$ is the signal.

We introduce the discrete exponential functions:

$$W_N^{(p+\theta)l} = \exp\left[-j\frac{2\pi}{N(p+\theta)}l\right].$$

The spectrum value $S_N^{(k+\theta)}(r)$ can be calculated from a sliding window with length N samples at the step r :

$$S_N^{(k+\theta)}(r) = \sum_{n=0}^{N-1} x(n+r)W_N^{(k+\theta)n}.$$

We use $X(z)$ to denote the z -transformation $x(n)$ and $Y(z)$ to denote $S_N^{(k+\theta)}(n)$. The transfer characteristic of the IIR filter shown in Fig. 6 is:

$$N(z) = \frac{Y(z)}{X(z)} = \frac{W_N^{-(k+\theta)}}{1 - z^{-1}W_N^{-(k+\theta)}},$$

where the integer variable k is replaced with the parametrically introduced variable $-(k + \theta)$, in which $k = 0, N - 1, 0 < \theta < 1$, and the set of analyzed frequencies is varied using the parameter θ : $\left\{\frac{2\pi(k + \theta)}{N}\right\}$.

The difference equations of the Goertzel filter are given below (the straight and reverse filter chains are to the right and to the left of the dashed line in Fig. 6, respectively):

for the reverse filter chain,

$$v(n) = 2 \cos(2\pi k/N)v(n-1) - v(n-2) + x(n),$$

for the straight filter chain,

$$y(n) = W_N^{-k}v(n) - v(n-1).$$

At the output of the Goertzel filter, we obtain the coefficients by which we can calculate the signal amplitude in each channel.

The Goertzel algorithm is convenient for reducing the computational efforts spent on processing the sound received by the device, since it replaces several procedures of other algorithms at once, i.e., the procedures of rectifying and applying a lowpass filter. The algorithm is a second-order IIR filter. Data processing occurs in blocks with a length of N points (a window), and the resulting values coincide with the corresponding coefficients of the discrete Fourier transform. A window with a given length N with an overlap calculates the coefficients at each sampling point.

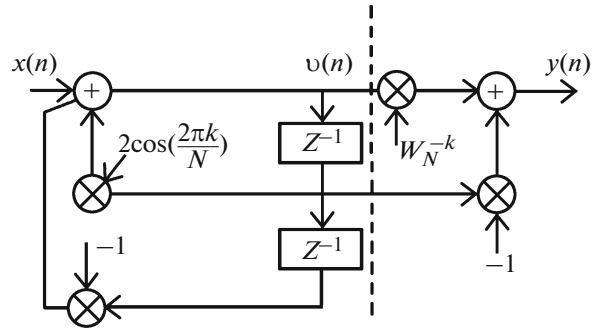


Fig. 6. The diagram of the Goertzel filter.

In our work, the simulation was carried out using the Hanning window [5]. The resulting complex values at each point of the signal determine its amplitude, i.e., allow us to obtain envelopes. Thus, it is possible to obtain 16 envelopes for each of the selected frequency components of the signal, after which the modulus of these complex values is found.

Thus, the Fourier coefficients for each individual frequency (that we selected from the series corresponding to the stimulation channels) allow us to calculate the amplitudes of the envelopes for each frequency channel. This is where the similarities to the CIS end. The CIS strategy also provides a rectified signal after the bandpass filter, which is similar in our case to the envelope. The CIS filter primarily calculates the modulus of the signal and then applies a low-pass filter. These three links are replaced with one Goertzel algorithm.

The efficiency of the algorithm can be estimated by comparing the frequency spectra of the signals (see graphs in Figs. 7a and 7b). It can be seen that noise appears at high frequencies (4.5–8.0 kHz) when the Goertzel algorithm is applied without input filters. We have shown that noise at these frequencies is suppressed when the Goertzel algorithm is applied with the input filters, as well as when the Hanning window is used without the filters.

A method exists that allows the recursive use of the Goertzel filter; its diagram is shown in Fig. 8. It turns out that every time we do not need to calculate the values at all points anew. This allows us to obtain the values of spectral samples on a real-time basis.

The payoff obtained by applying the recursive formula to the modified Goertzel filter can be seen by looking at the number of steps required for the operation of the filter (see the comparison in Table 1).

The recurrent equation for the sliding discrete Fourier transform is:

$$S_N^k(n) = W_N^{-k}[S_N^k(n-1) + x(n) - x(n-N)],$$

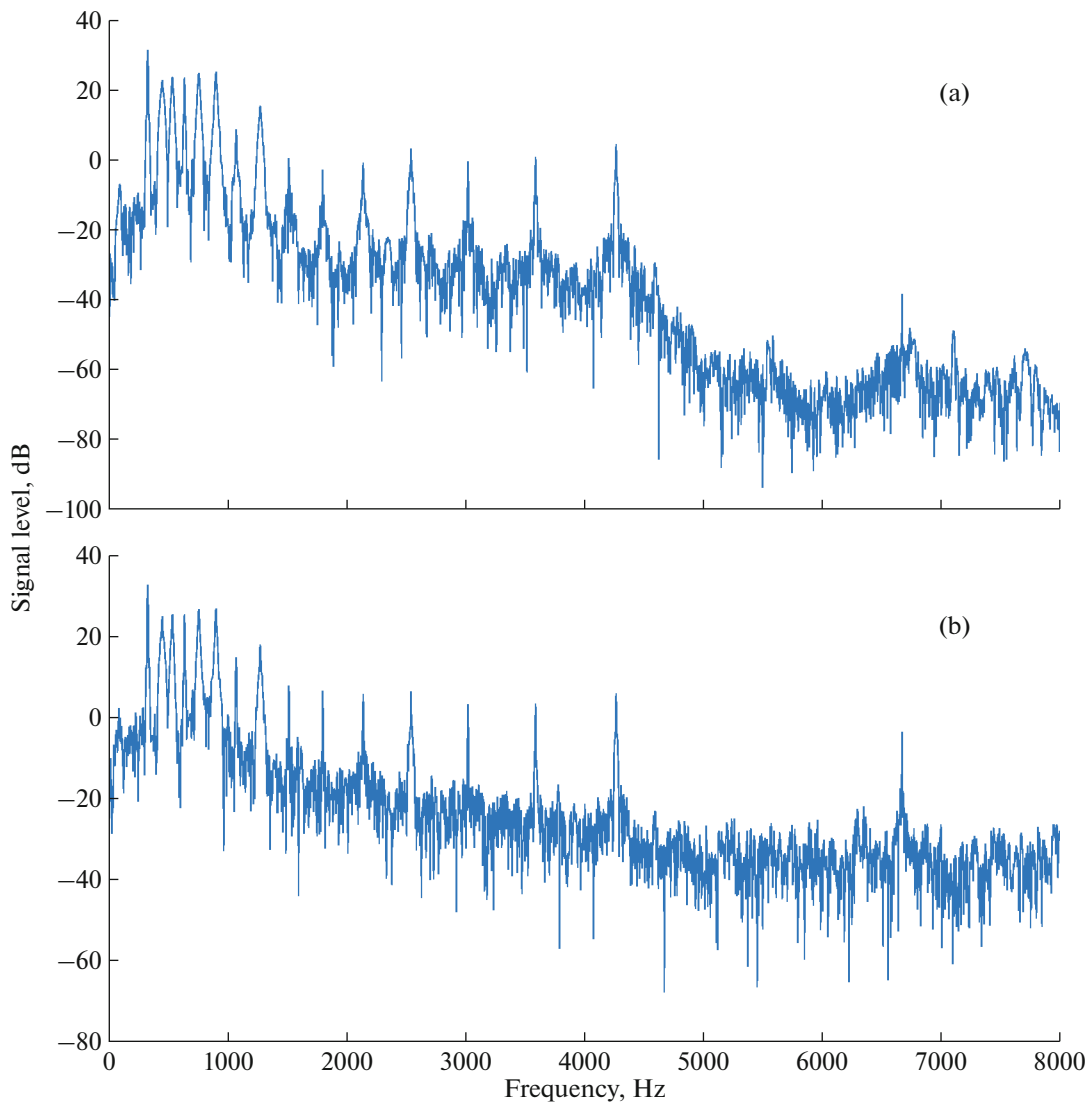


Fig. 7. The amplitude spectrum of the resulting signal after applying the Goertzel algorithm (a) with the bandpass IIR filter and (b) without it.

where $S_N^k(n)$ is the value of the k th fixed number of the spectral sample in the N -point discrete Fourier transform at the time instant n .

It should also be noted here that the computation of the Fourier transform using the Goertzel algorithm

does not imply the use of filters instead of a window function. Therefore, bandpass filters should be applied at the input so that we can use a recursive algorithm with a low consumption of computational resources. As a result, the application of filters results in less computational effort compared to the use of a window function, since the window is similar in computational efforts to FIR filters, and IIR filters are less computationally intensive. Thus, the least resource-intensive result is obtained when IIR filters are applied at the input and the modified recursive Goertzel algorithm is used.

Thus, after conducting a series of tests, we came to the conclusion that although the window function provides the same result as the filters, the filters without the window function are more convenient, since they allow the use of a modified (recursive) algorithm

Table 1. The number of operations required by the filter [4]

Method	Calculation of $S_N^{(N)}(k, \theta)$	
	real multiplications	valid additions
Goertzel algorithm	$N + 2$	$2N + 2$
Modified Goertzel algorithm	4	4
Discrete Fourier transform	$4N$	$4N$

that requires significantly less computational resources. Therefore, the best option is to use IIR filters at the input and replace the window function with the recursive modified Goertzel algorithm.

4.3. The Reverse Transformation

It is clear that the quality of a signal perceived by a person is worse compared to the signal that initially entered the device. We carry out the inverse transformation in order to understand what kind of signal is perceived by a person, i.e., we restore the original signal with coding losses. For this purpose, the resulting amplitude is multiplied by the cosine of the frequency corresponding to one of the 16 channels. The initial phases are not taken into account, since this information cannot be restored as a result of sound coding, and this is not important for speech recognition. We obtain the reconstructed sound by adding the signals reconstructed using the cosines in all channels:

$$\text{signal}(j, i) = \text{amplitude}(j, i) \cos\left(\frac{\text{freq}(i)2j\pi}{F_s}\right),$$

where i is the channel number, j is the line number in the array containing the signal or its amplitude, F_s is the sampling frequency, $\text{amplitude}(j, i)$ presents the coefficients that are equal to the signal amplitude, and $\text{freq}(i)$ is the frequency corresponding to the channel. Therefore, the recovered signal in the channel is obtained in the $\text{signal}(j, i)$ array.

Having received the reconstructed signal, we thus check the operation of the algorithm.

4.4. Nonlinear Compression

Cochlear nerve endings perceive current stimulation nonlinearly. With electrical stimulation, loudness is perceived according to the law:

$$L = k_a p^{\beta_a} = k_e i^{\beta_e},$$

where i is the current, p is the sound pressure, and the coefficients are $\beta_a = 0.6$ and $\beta_e = 2.7$ [6]. The current, and, therefore, the microphone voltage are determined by the law:

$$i = k_m p^{\beta_m},$$

where $\beta_m = \beta_a / \beta_e = 0.22$.

It can be seen that the perceived loudness is associated with stimulating impulses according to the power law. Therefore, it is necessary to perform nonlinear compression after obtaining the envelopes in order to stimulate the electrodes of the device.

This results in obtaining the amplitudes of electric pulses y . They are set at a certain level of values (comfort M and threshold B). The purpose of the nonlinear compression is to transfer the entire signal into a comfortable range that can be adjusted by the doctor. The formula for the nonlinear signal compression is [6]:

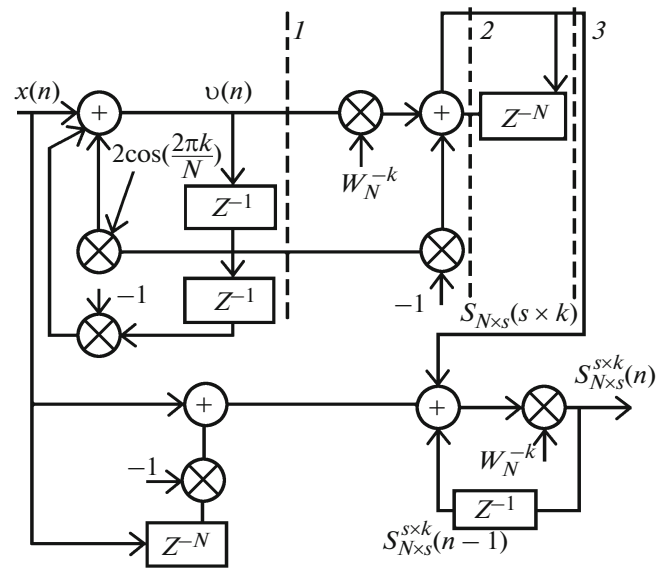


Fig. 8. The diagram of the modified Goertzel filter (the sliding discrete Fourier transform).

$$y = \begin{cases} \frac{\log\left(1 + \alpha\left(\frac{v - B}{M - B}\right)\right)}{\log(1 + \alpha)}, & B \leq v \leq M, \\ 0, & v \leq B, \\ 1, & v \geq M. \end{cases}$$

where v is the envelope, α is the compression ratio, M is the saturation level, and B is the threshold value.

4.5. The Sliding-Average Algorithm

In order to eliminate low-frequency oscillations in the signal, the sliding-average method can be applied to the envelope in each of the channels instead of the standard lowpass filter. In this case, to obtain the best result, we should take a third-order filter and select its length so that it is a multiple of the frequency period of the stimulation channel. At first, we need to determine the window (its width) over which the averaging will be performed.

The function obtained after the transformation is numerically equal at each point to the average value of the original function, which was calculated over the initially specified smoothing interval (the number of values of the original function used in the calculation). As a result, the original data are smoothed, and the above algorithm implements envelope filtering.

The values for the sliding-average method are calculated according to the formula:

$$\bar{X}(k) = \frac{1}{n} \sum_{t=k}^{n+k} X(t),$$

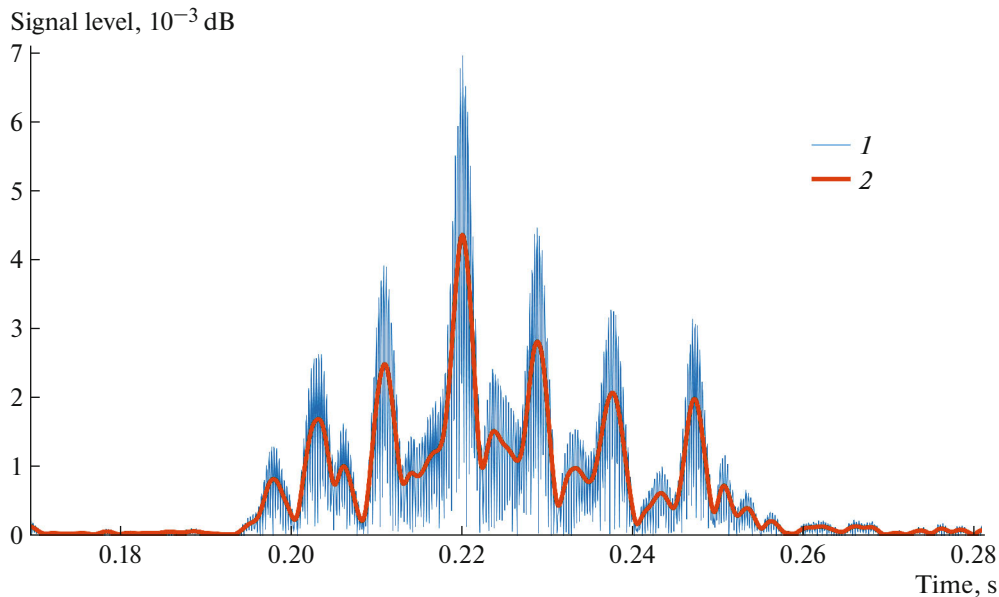


Fig. 9. (1) The envelope and (2) the smoothed envelope obtained using the sliding-average method in the 10th channel.

where n is the window size (the smoothing period) and k is the number of the term of the series whose value is replaced by the average.

Figure 9 shows the envelope 2, which was filtered using the third-order sliding-average method, as well as the original envelope 1, which was filtered. It should be noted that the envelope modulus was preliminarily calculated.

A 400-Hz lowpass filter was also used for comparison [6].

As mentioned above, this stage can be omitted and replaced with a single procedure, i.e., the Goertzel algorithm.

5. DETERMINATION OF THE ELECTRODE-STIMULATION FREQUENCY

The stimulation frequency is the number of fully completed pulses in all stimulation channels on the electrode array per second. The range of stimulation frequencies is quite wide, depending on the manufacturer.

We consider the HiRes™ Ultra Cochlear implant from Advanced Bionics as an example. From the description of the device, we find out that the maximum pulse duration is 229 μs , the maximum stimulation frequency is 83 kHz, and the number of channels is 16. We suppose that the stimulation period (it includes both the positive and negative pulses) is 192 μs ; then the frequency in an individual channel is ≈ 5.2 kHz, and the total frequency is $5.2 \times 16 \approx 83$ kHz. The assumption about the determination of the stimulation frequency is thus confirmed.

This device can implement different strategies, including virtual channels (hiRes 120) or paired stimulation in each channel (the hiRes paired strategy). Let us consider the latter in more detail. The stimulation frequency for the strategies using 8 and 16 channels is presented in Table 2 [7].

Although the stimulation frequency is very high in some devices, it fails to be a decisive factor in the quality of sound transmission, since the refractory period does not allow effective stimulation with a frequency higher than nerve endings are able to perceive. This time for a person is approximately 500 μs [8]. In addition, simultaneous stimulation of several zones causes cross excitation of the nerve endings of neighboring zones when the paired stimulation strategy is used, since the electrode array is in a conductive fluid, which does not allow stimulation of only one zone.

A study conducted on a group of 13 adults to compare paired and sequential stimulation [7] has shown that the sequential-stimulation strategy provides the best results. It also follows that the stimulation frequency of 40 kHz is quite enough for high-quality sound reproduction. The authors in [7] noted that more pronounced differences in perception between the two strategies are felt in the presence of background noise, which correlates well with real-life conditions.

Table 3 presents information on different manufacturers of cochlear implantation systems and their stimulation frequencies [9].

6. STIMULATION USING VIRTUAL CHANNELS

Along with the CIS strategy in which the stimulation is strictly sequential, there is another interesting

Table 2. Stimulation frequency vs. the number of channels and the type of stimulation

Number of channels	Stimulation frequency per channel, pulses/s	Total stimulation frequency, pulses/s	Stimulation type
16	5000	80000	Pair
16	2500	40000	Consecutive
8	5000	40000	Consecutive
8	2500	20000	Consecutive

Table 3. The characteristics of cochlear implant systems of various manufacturers [9]

Manufacturer, model	Med-El, SYNCHRONY 2	Nurotron, Venus	AB HiRes, 90 K	Nucleus, N6
Strategy	CIS	CIS	HiRes	SPEAK
Stimulation frequency over the entire electrode array, kHz	50	40	83	32
Number of electrodes	12	24	16	22

method. This is a technique that uses virtual channels and several sources to increase the number of different frequencies in transmitted sound. Electrodes to be stimulated are selected depending on the loudness. Two adjacent electrodes are stimulated by two impulses. The proportion of the current that will be delivered to two adjacent electrodes is also determined by the loudness ratio. The two outer electrodes are used as ground, directing current through the inner electrodes (two electrodes generate one virtual channel). Virtual channels result from a shift of the current density in the region between the electrodes; this happens when the current is unevenly distributed between the electrodes. The region of the cochlea between the electrodes is stimulated and the ratio of the supplied currents determines the place of stimulation.

In our case, the reference electrode is the body of an implant. If the current enters the reference electrode after leaving the stimulating zone, this is a monopolar stimulation mode. If the current leaves a certain electrode and enters an adjacent one, this is a bipolar mode.

7. CONCLUSIONS

In various countries that do not yet have a national cochlear implantation system, the scientific community is working towards developing their own stimulation strategies and devices to make hearing restoration operations more affordable. Thus, the UGR (University of Granada, Spain) team developed the Cochlear Implant Simulation version 2.0 program. It simulates the cochlear implantation system with the CIS strategy and presents the results of sound processing using known strategies [10]. This program allows the use of the Hilbert transform or the rectification and a low-pass filter for the calculation of the signal envelope. By adjusting various parameters, one can take the

interaction of adjacent channels and some physiological characteristics of a patient into account, select the number of stimulation channels, etc. We provide open source code for the MATLAB program (see Appendix) of our strategy.

The first block of the developed algorithm contains filtering with bandpass filters that divide the signal into 16 channels. This stage is also used in other stimulation strategies. Filters remove distortions and preserve the window function, due to which we save computational resources when calculating the Fourier transform. The second block we introduced contains the Goertzel recursive algorithm (it also has the advantage of saving resources), which is followed by nonlinear compression, lowpass filters, and, finally, signal transmission to the electrodes. Both the sequential and parallel stimulation can be used at the output.

In analyzing the simulation results, we drew the conclusion that, although the windowed Fourier transform provided the same result as the bandpass filters, it is more convenient to use bandpass filters without a window function. This makes it possible to use the modified (recursive) Goertzel algorithm for the Fourier transform, due to which the number of calculations can be significantly reduced. Therefore, the best option is to use IIR filters at the input and replace the windowed Fourier transform with the recursive modified Goertzel algorithm. It can also be concluded that the algorithm reproduces a high-quality signal.

Thus, a proprietary stimulation strategy that optimizes the computational efforts for sound processing has been developed for the cochlear implantation system. The inverse transformation was carried out to compare the original and reconstructed signals and evaluate the quality of the algorithm by ear. The operation of the sound coding algorithm for the cochlear implantation system was simulated in the MATLAB system. We obtained a means to hear how the sound

would change after it was processed by our stimulation strategy. We came to the conclusion that high-quality speech reproduction was achieved, but the perception of the full range of sound signals, e.g., music and sounds of nature, was difficult at that stage. Therefore,

further research will focus on simulating the strategy using virtual channels to increase the frequency resolution of auditory perception for patients with a cochlear implantation system with the same number of physical stimulation channels.

APPENDIX

Below is a possible implementation of the main stages of the developed algorithm with comments in MATLAB. The original signal was converted into sound perceived by an owner of the cochlear implantation system in order to assess its quality.

```
clear all;
close all;
% First, let us define a set of frequencies for 16 channels:
frq=[333 455 540 642 762 906 1076 1278 1518 1803 2142 2544 3022 3590 4264
6665];
% Number of channels:
sz = size(frq);
n_channel = sz(2);
% Bandpass filters:
a1= [272 394];
c1=a1./8000;
a2= [412.5 497.5];
c2=a2./8000;
a3= [489 591];
c3=a3./8000;
a4= [582 702];
c4=a4./8000;
a5= [690 834];
c5=a5./8000;
a6= [821 991];
c6=a6./8000;
a7= [975 1177];
c7=a7./8000;
a8= [1158 1398];
c8=a8./8000;
a9= [1375.5 1660.5];
c9=a9./8000;
a10= [1633.5 1972.5];
c10=a10./8000;
a11= [1941 2343];
c11=a11./8000;
a12= [2305 2783];
c12=a12./8000;
a13= [2738 3306];
c13=a13./8000;
a14= [3253 3927];
c14=a14./8000;
a15= [3863.5 5000];
c15=a15./8000;
a16= [4947 6084];
```

```

c16=a16./8000;
[B1,A1]=butter(1,c1,'bandpass');
[B2,A2]=butter(1,c2,'bandpass');
[B3,A3]=butter(1,c3,'bandpass');
[B4,A4]=butter(1,c4,'bandpass');
[B5,A5]=butter(1,c5,'bandpass');
[B6,A6]=butter(1,c6,'bandpass');
[B7,A7]=butter(1,c7,'bandpass');
[B8,A8]=butter(1,c8,'bandpass');
[B9,A9]=butter(1,c9,'bandpass');
[B10,A10]=butter(1,c10,'bandpass');
[B11,A11]=butter(1,c11,'bandpass');
[B12,A12]=butter(1,c12,'bandpass');
[B13,A13]=butter(1,c13,'bandpass');
[B14,A14]=butter(1,c14,'bandpass');
[B15,A15]=butter(1,c15,'bandpass');
[B16,A16]=butter(1,c15,'bandpass');
% Open the file in the script execution directory
directory_content = dir;
% Returns the path to the open directory
exe_path = directory_content(1).folder;
file_name = sprintf('%s\\%s', exe_path, 'your_file_name.wav');
[x,fs] = audioread(file_name);
% Sampling frequency:
FS = 16e3;
x = resample(x, FS, fs);
% Signal length:
sz = size(x);
x_len = sz(1);
% Signal data:
x=x(:,1);
pass1=fir1(100,c1,'bandpass');
pass2=fir1(100,c2,'bandpass');
pass3=fir1(100,c3,'bandpass');
pass4=fir1(100,c4,'bandpass');
pass5=fir1(100,c5,'bandpass');
pass6=fir1(100,c6,'bandpass');
pass7=fir1(100,c7,'bandpass');
pass8=fir1(100,c8,'bandpass');
pass9=fir1(100,c9,'bandpass');
pass10=fir1(100,c10,'bandpass');
pass11=fir1(100,c11,'bandpass');
pass12=fir1(100,c12,'bandpass');
pass13=fir1(100,c13,'bandpass');
pass14=fir1(100,c14,'bandpass');
pass15=fir1(100,c15,'bandpass');
pass16=fir1(100,c16,'bandpass');
% Applying bandpass filters:
outhi1=filter(pass1,1,x);
outhi2=filter(pass2,1,x);

```

```

outhi3=filter(pass3,1,x);
outhi4=filter(pass4,1,x);
outhi5=filter(pass5,1,x);
outhi6=filter(pass6,1,x);
outhi7=filter(pass7,1,x);
outhi8=filter(pass8,1,x);
outhi9=filter(pass9,1,x);
outhi10=filter(pass10,1,x);
outhi11=filter(pass11,1,x);
outhi12=filter(pass12,1,x);
outhi13=filter(pass13,1,x);
outhi14=filter(pass14,1,x);
outhi15=filter(pass15,1,x);
outhi16=filter(pass16,1,x);
% After applying bandpass filters, the signal is represented in the form:
out_vec=[outhi1, outhi2, outhi3, outhi4, outhi5, outhi6, outhi7, outhi8,
outhi9, outhi10, outhi11, outhi12, outhi13, outhi14, outhi15, outhi16];
% Without pre-filtering with bandpass filters, the signal recording looks like this:
% out_vec = [x, x, x, x, x, x, x, x, x, x, x, x, x, x, x, x];
% Now, we obtain complex values separately for each channel in order to calculate then % the amplitudes.
%Below is the implementation of the Goertzel algorithm [5]. Let us determine the width of the window N. The
%transfer characteristic of the IIR filter is described by the formula:
%  $1 - W * z^{-1}$ 
%  $H(z) = \frac{1 - W * z^{-1}}{1 - \alpha * z^{-1} + z^{-2}}$ 
%  $W = W_N^{(k)} = \exp(-2i * \pi * k / N)$ .
%  $\alpha = 2 * \cos(2 * \pi * k / N)$ .
N = 250; % width of the window
for i = 1:n_channel
    f_norm = frq(i) / FS;
    w = exp(-2i*pi*f_norm);
    alpha = 2*cos(2*pi*f_norm);
    b_opp = [1, -w, 0];
    a_opp = [1, -alpha, 1];
    for j = 1:x_len - N
        dft_window = filter(b_opp, a_opp, out_vec(j:j+N, i));
        dft_data(j+N, i) = dft_window(end) / N;
    end
% The length of the received data is shorter relative the original data by the width of the window:
    for j = x_len - N + 1:x_len
% We add the final value:
        dft_data(j, i) = dft_data(x_len - N, i);
    end
end
% We get the amplitudes from the complex quantities, and, using them, the envelopes in each channel:
dft_data_n=abs(dft_data);
% Obtaining envelopes by multiplying by cosine:
for i=1:n_channel
    for j=1:x_len
        z(j,i)=dft_data_n(j,i)*cos(frq(i)*2*pi*j/FS);
    end
end

```

```

end
% Sum of all channels with envelopes:
for j=1:x_len
% Zero one line and write the resulting sum there:
    sum_signalw(j) = 0;
% Let us summarize the obtained envelopes, save the obtained results in an audio file:
for i=1:n_channel
    sum_signalw(j) = sum_signalw(j) + z(j,i);
end
end
% Save the file in the script execution directory:
file_name = sprintf('%s\\%s', exe_path, 'recreated_sound_overall.wav');
audiowrite('recreated_sound_my.wav', sum_signalw, 16000);

```

FUNDING

This work was supported by the Ministry of Science and Higher Education of the Russian Federation as a part of the research and development work on the Development of the Manufacturing Technology and Launching into Production of an Import-substituting Multifunctional Cochlear Implantation System for the Rehabilitation of Patients with Sensorineural Hearing Loss topic (agreement on the provision of a subsidy for implementation of a comprehensive project for the creation of high-tech production dated July 10, 2018, no. 074-11-2018-028).

REFERENCES

1. Novoselova, S.M., *J. Sov. Math.*, 1991, vol. 57, no. 3, p. 3147.
<https://doi.org/10.1007/BF02355165>
2. Choi, C.T.M. and Lee, Y.-H., in *Cochlear Implant Research Updates*, Intech, 2012, vol. 2, no. 2005.
<https://doi.org/10.5772/47983>
3. Choi, C.T.M., Hsu, C.H., Tsai, W.Y., and Lee, Y.H., in *IFMBE Proceedings*, Berlin, Heidelberg: Springer, 2009, p. 310.
https://doi.org/10.1007/978-3-540-92841-6_76.
4. Ponomarev, V.A., *Digital Signal Process.*, 2014, no. 1, p. 3.
5. Oppenheim, A.V. and Schaffer, R.W., *Discrete-Time Signal Processing (Prentice-Hall Signal Processing Series)*, Pearson, 2009.
6. Swanson, B.A., *PhD Thesis*, Melbourne: Univ. of Melbourne, 2008.
7. Buechner, A., Frohne-Buechner, C., Stoeber, T., Gaertner, L., Battmer, R.-D., and Lenarz, T., *Otol. Neurotol.*, 2005, vol. 26, no. 5, p. 941.
<https://doi.org/10.1097/01.mao.0000185069.27705.f0>
8. Lai, W., *An NRT Cookbook: Guidelines for Making NRT Measurements*, Zurich: Cochlear AG, 1999.
<https://www.uzh.ch/orl/Cookbook.pdf>.
9. Zeng, F.-G., Rebscher, S., Qian-Jie Fu, Hongbin Chen, Xiaoan Sun, Li Yin, Lichuan Ping, Haihong Feng, Shiming Yang, Shusheng Gong, Beibei Yang, Hou-Yong Kang, Na Gao, and Fanglu Chi, *Hear. Res.*, 2015, vol. 322, p. 188.
<https://doi.org/10.1016/j.heares.2014.09.013>
10. de la Torre Vega, A., Bastarrica Marti, M., de la Torre Vega, R., and Sainz Quevedo, M., Cochlear Implant Simulation Version 2.0 Description and Usage of the Program. http://www.ugr.es/~atv/web_ci_SIM/en/ci_sim_en.htm.

Translated by N. Goryacheva