BULLETIN OF THE POLISH ACADEMY OF SCIENCES
TECHNICAL SCIENCES, Vol. 59, No. 2, 2011
DOI: 10.2478/v10175-011-0016-z

Application of evolutionary algorithm to design minimal phase digital filters with non-standard amplitude characteristics and finite bit word length

A. SŁOWIK*

Department of Electronics and Computer Science, Koszalin University of Technology, 2 Śniadeckich St., 75-453 Koszalin, Poland

Abstract. In this paper an application of evolutionary algorithm to design minimal phase digital filters with non-standard amplitude characteristics and with finite bit word length is presented. Four digital filters with infinite impulse response were designed using the proposed method. These digital filters possess: linearly falling characteristics, linearly growing characteristics, nonlinearly falling characteristics, and nonlinearly growing characteristics, and they are designed using bit words with an assumed length. This bit word length is connected with a processing register size. This register size depends on hardware possibilities where digital filter is to be implemented. In this paper, a modification of the mutation operator is introduced too. Due to this modification, better results were obtained in relation to the results obtained using the evolutionary algorithm with other mutation operators. The digital filters designed using the proposed method can be directly implemented in the hardware (DSP system) without any additional modifications.

Key words: artificial intelligence, evolutionary algorithms, digital filters, minimal phase, finite bits word length, non-standard amplitude characteristics.

1. Introduction

Digital signal filtering is a very important problem, which is very often used in practical applications. Among digital filters, we can mention filters with finite impulse response (FIR) and filters with infinite impulse response (IIR) [1]. Filters IIR are very effective and require considerably less number of multiplication, than FIR digital filters. The multiplication in digital filters is required to compute value of single sample of processed signal with assure the assumed frequency characteristics. Therefore, from the hardware point of view, the IIR filters can be very fast and permit the signal processing in real time [1, 2].

The main goals during digital filters design are assurance of filter stability and fulfilment of design assumptions connected with the shape of amplitude characteristics. In order to obtain assumed characteristics we can use one of existing approximations such as: Butterworth, Chebyshev or Cauer during the design process. But, the problem is complicated in the case, when designed filter should have a non-standard amplitude characteristics [2]. The non-standard amplitude characteristics are widely used in different kind of amplitude equalizers. Therefore, in this case, the standard approximations are useless and we must use one of the continuous optimization techniques (in the case of a design of digital filters without bit word length constraint) or one of discrete optimization techniques (in the case of design of digital filters with finite bit word length). Additionally, the function describing the problem of digital filters design is a multi-modal function [3]. Therefore, we must use one of the global optimization techniques. Among global optimization techniques we can mention: differential evolution algorithms [4, 5], particle swarm optimization algorithms [6], artificial bees colony optimization algorithms [7], continuous ant colony algorithm [8, 9], ant colony optimization algorithm [10], evolution strategy algorithms [11], cultural algorithms [12] or evolutionary algorithms [13–15]. Evolutionary algorithms are one of the most popular techniques of global optimization. Due to mutation and crossover operators, the evolutionary algorithms can escape from local extremes. Additionally, the evolutionary algorithms are very universal tools, which can be used in both kinds of optimization: in continuous optimization and in discrete optimization.

Also, during digital filter designing which will be implemented in the hardware, we must remember about physical constraints of the hardware. These constraints are based on finite length of processing registers, and the finite length of bit word representing the filter coefficients [16, 17]. Therefore, during digital filters design we must tend to create the digital filter resistive to rounding errors. Especially, it is important during design of IIR digital filters, because these filters are very sensitive to variation of values of filter coefficients [1]. Also, during IIR digital filters design, it is important to design the minimal phase digital filters. Minimal phase digital filters have two main advantages: reduced filter length and minimal group delay. Minimal phase digital filters generally require fewer computations and less memory than linear phase filters [18, 19]. Of course for design of minimal phase IIR digital filters with arbitrary amplitude characteristics we can use a Yule Walker method [20, 21]. This method designs recursive IIR digital filters using a least-squares fit to a specified

*e-mail: aslowik@ie.tu.koszalin.pl

frequency response [20, 21]. But the problem is complicated when we want to use a designed filter in programmable fixed-point DSP processors that are used for real-world applications. Then the filter coefficients obtained using Yule Walker algorithm must be scaled to the range $[-1; 1]$, and next these coefficients must be quantized to the fixed-point numbers. In real world applications the 16-bit fixed-point (Q.15) format is commonly used in most 16-bit fixed-point DSP processors, such as the TMS320C5000 [22, 23]. Q.15 format represents numbers in the range of -1 to $1 - 2^{-15}$ using a sign bit and 15 fractional bits with 2's complement format [24]. When the coefficients (for IIR digital filter) obtained using Yule Walker algorithm will be transformed to the Q.15 format, and applied in DSP processor, the shape of amplitude characteristics of designed filter probably will be changed, because IIR filters are very sensitive to variation of values of filter coefficients [1]. In the worst case, after quantization process, the digital filter designed using Yule Walker algorithm will be not fulfill design requirements after its implementation in DSP system.

Also, in literature, we can find review of genetic algorithms, which are used in design of digital filters with finite bit word length [25]. Also, in literature we can find the papers where genetics algorithms are used to design IIR digital filters with minimal phase (for example [26]). But, it is hard to find the papers, where the design problem of minimal phase digital filters with finite bit word length and non-standard amplitude characteristics is described.

In this paper, the evolutionary method of design of minimal phase IIR digital filters with non-standard amplitude characteristics and with finite bit word length is presented. Of course, the proposed method can be used in design of FIR digital filter after some small modifications. In this paper, the modification of mutation operator is introduced too. Due to this modification, we can obtain better algorithm convergence to better results. As a test of proposed method, the four 16-bit minimal phase IIR digital filters with amplitude characteristics: linearly growing, and linearly falling, and non-linearly growing, and non-linearly falling were designed using proposed approach. The method described in this paper is named EA-MP-FWL-FD (*Evolutionary Algorithm – Minimal Phase – Finite Word Length – Filter Design*). The main advantage of proposed method is that after design process, the designed filter can be implemented in the DSP processor without any additional changes. The application of scaling or quantization of digital filter coefficient are not needed, because the filter coefficients are in one of fixed-point formats (in this paper, the Q.15 format is considered). Therefore after implementation of designed filter in the hardware (DSP processor), the properties of this filter are not changed.

2. IIR digital filters

The transmittance of IIR filters in z domain can be described using following equation:

$$H(z) = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2} + \ldots + b_n \cdot z^{-n}}{1 - (a_1 \cdot z^{-1} + a_2 \cdot z^{-2} + \ldots + a_n \cdot z^{-n})}. \quad (1)$$

The main goal of the design algorithm of digital filters is to find a such a set of filter coefficients a_i, b_i ($i \in [1; n]$, where *n* is a filter order) in order to designed filter will be stable, minimal phase filter, and fulfill all design assumptions. However, if we want to obtain a digital filter, which will be resistive on rounding errors, the filter coefficients must take exactly determined values dependent on number of bits used in representation of each filter coefficient. In the case, when the number of bits which are used to represent the filter coefficient is equal to *nb*, then the *M=nb*-1 bits are allowable to realization of a value of the filter coefficient (one bit is taken as a sign bit). Therefore, the digital filter coefficients can take the values from the following domain *D* (in fixed-point format Q.*M*):

$$D = \left[\frac{(-1) \cdot 2^M}{2^M}; \frac{2^M - 1}{2^M}\right]. \quad (2)$$

In the 2's complement fractional representation, an *nb* bit binary word can represent 2^{nb} equally space numbers from $\frac{(-1) \cdot 2^M}{2^M} = -1$ to $\frac{2^M - 1}{2^M} = 1 - 2^{-M}$ (see Eq. (2)).

The binary word *BW* which consists of *nb* bits (bw_i):

$$BW = bw_M, bw_{M-1}, bw_{M-2}, \ldots, bw_2, bw_1, bw_0$$

we interpret as a fractional number *x*:

$$x = -(b_M) + \sum_{i=0}^{M-1} \left(2^{i-M} \cdot bw_i\right). \quad (3)$$

Of course if we use a fractional number in Q.*M* format, the value of coefficient a_0 will be not equal to 1 (see Eq. (1)), but a_0 will be equal to $1 - 2^{-M}$.

In order to assure, the stability of digital filters, the poles of the function (1) must be placed in unitary circle in the z plane. Of course, in order to assure, that designed filter will be minimal phase digital filter, the zeros of the function (1) must be placed also in unitary circle in the z plane.

3. Proposed method EA-MP-FWL-FD

The proposed method EA-MP-FWL-FD consists of eight steps.

In the first step, create the set *D* (see Eq. (3)) consisting of 2^{nb} values is created. For each value from the set *D* the index value is assigned. The first value from set *D* possesses index number 1, the last value from the same set is represented by index 2^{nb}. Next, the population *Pop* is randomly created. The population *Pop* consists of *PopSize* individuals. Each individual in population consists of $2 \cdot n + 1$ genes (n is represents the filter order). Each gene takes one integer value form the range $[1; 2^{nb}]$. The value written down in each gene, points to the adequate filter coefficient from the set *D*.

In the second step, an evaluation of all individuals using objective function *FC* is performed (objective function *FC* is described in the fourth section of this paper). Presented evolutionary algorithm tends to minimize the objective function *FC*.

In the third step, the best individual (having the lowest value of objective function *FC*) is selected from current population *Pop*. In the case, first algorithm iteration, selected the best individual is remembered in the variable *TheBest*. During other algorithm iterations, selected the best individual is remembered as the *TheBest*, if and only if the value of its objective function *FC* is lower than the value of the objective function for individual actually stored in the variable *TheBest*.

In the fourth step, a selection of individuals to the new population is performed. The tournament selection [13, 14] with the size of tournament group equal to 2 is chosen as a selection operator in the proposed algorithm.

In the fifth step, the best individual is selected from current population *Pop*. In the case, when the value of the objective function *FC* for selected individual is higher than the value of the objective function *FC* for individual stored in the variable *TheBest*, then the individual *TheBest* is inserted in the place of the best individual in current population *Pop*. Otherwise, a any changes are not applied.

In the sixth step, a cross-over of individuals in the population *Pop* is performed. A single one point cross-over [13, 14] is chosen as a cross-over operator. The single cross-over operator, depends on choosing (with probability *PC*) a pair of individuals form the population *Pop*. Next, the cutting point is randomly chosen for each pair of individuals. After individual cutting, the cut fragments of parental individuals are exchanged between them. Due to this exchange, the two child individuals are created. The two new individuals (child individuals) are inserted into the place of their parental individuals in the population *Pop*.

In the seventh step, a mutation of individuals is performed. The mutation operator is executed with probability *PM* for each gene in each individual in the population *Pop*. If *i*-th gene from *j*-th individuals is selected to the mutation, then its new value is determined as follows:

$$NG_{i,j} = \begin{cases} G_{i,j} + A, & \text{when } \; r < 0.5 \text{ and } \; G_{i,j} < 2^{nb} \\ G_{i,j} - A, & \text{when } \; r \geq 0.5 \text{ and } \; G_{i,j} > 1 \\ G_{i,j}, & \text{otherwise} \end{cases}, \tag{4}$$

where $NG_{i,j}$ is a new value of *i*-th gene in *j*-th individuals in population *Pop*, $G_{i,j}$ is a current value of *i*-th gene in *j*-th individual, *r* is a random value from the range $[0; 1)$, A is an integer value from the range $[1,\ 2^M]$, and it is computed as follows:

$$A = \text{round}\left(\left(1 - \frac{Iter}{G_{\max}} \cdot A_{init}\right) + 1\right), \tag{5}$$

where A_{init} is an integer value from the range $[1,\ 2^M]$, $G_{\max}$ is maximal number of algorithm iteration, $Iter$ is a number of current iteration.

The proposed mutation operator (see Eq. (4) and Eq. (5)) considerably improved the quality of obtained results in relation to the results obtained using simple mutation operator [13, 14], and mutation presented in paper [27]. The simple mutation operator depends on choosing of new value of genes from the whole accessible range of variability $[1;\ 2^{nb}]$. The operator presented in paper [27] depends on choosing of new value of genes by adding 1 or subtracting 1 from the value of genes chosen to mutation. The mutation operator presented in this paper is more adaptive. At the start of the algorithm it searches solution space more globally, and at the finish of the algorithm it searches solution space more locally.

If after mutation a new value $NG_{i,j}$ of *i*-th gene in *j*-th individual in population *Pop* is equal to negative number or zero number then value 1 is assigned to given gene. In the case, if after mutation the value of $NG_{i,j}$ is higher than 2^{nb} then value 2^{nb} is assigned to this gene.

In the eight step, algorithm termination criteria are checked. Reaching of maximal number of algorithm iteration $G_{\max}$ or reaching of solution having the value of the objective function *FC* equal to 0 are assumed as a termination conditions in proposed algorithm. If algorithm termination criteria are fulfilled, then the algorithm is stopped and the result stored in *TheBest* individual is returned as a solution of a given problem. But, if the algorithm termination criteria are not fulfilled, then jumping to the second step of the proposed algorithm is performed.

4. Objective function FC

In order to obtain the value of objective function *FC* for *i*-th individual in the population, firstly the amplitude characteristics $H(f)_i$ which coefficients are stored in the *i*-th individual is computed. The amplitude characteristics is computed using *R* values of normalized frequency $f \in [0; 1]$ (where 1 represents the Nyquist frequency; in proposed method normalized frequency is divided into R points). Also, the poles and zeros of transmittance function (see Eq. (1)) are computed for each individual in the population *Pop*. If we have amplitude characteristics and the values of poles and values of zeros of transmittance function for *i*-th individuals, we can compute the objective function *FC*. The objective function *FC* is computed as follows (in Eqs. (6)–(12)) the index *i* represents *i*-th individuals in population):

$$\begin{gathered} FC_i = AmplitudeError_i + \\ w \cdot \left(StabError_i + MinPhaseError_i\right), \end{gathered} \tag{6}$$

$$AmplitudeError_i = \sum_{k=1}^{R} AmpErr_{i,k}, \tag{7}$$

$$AmpErr_{i,k} = \begin{cases} H\left(f_k\right)_i - Upper_{i,k}, \\ \qquad\qquad \text{when } H\left(f_k\right)_i > Upper_{i,k} \\ Lower_{i,k} - H\left(f_k\right)_i, \\ \qquad\qquad \text{when } H\left(f_k\right)_i < Lower_{i,k} \\ 0, \qquad \text{otherwise} \end{cases}, \tag{8}$$

$$StabError_i = \sum_{j=1}^{J} StabErr_{i,j}, \tag{9}$$

$$StabErr_{i,j} = \begin{cases} |p_{i,j}| - 1, & \text{when } |p_{i,j}| \geq 1 \\ 0, & \text{otherwise} \end{cases}, \tag{10}$$

$$MinPhaseError_i = \sum_{q=1}^{Q} PhaseErr_{i,q} \tag{11}$$

$$PhaseErr_{i,q} = \begin{cases} |z_{i,q}| - 1, & \text{when } |z_{i,q}| \geq 1 \\ 0, & \text{otherwise} \end{cases}, \tag{12}$$

where w is a value of penalty factor (during experiments $w = 10^5$ is assumed), $AmpErr_{i,k}$ is a partial value of amplitude characteristics error for k-th value of normalized frequency, $H(f_k)_i$ is a value of amplitude characteristics for k-th value of normalized frequency f, $Lower_{i,k}$ is a value of lower constraint for amplitude characteristics value for k-th value of normalized frequency, $Upper_{i,k}$ is a value of upper constraint for amplitude characteristics value for k-th value of normalized frequency, $StabErr_{i,j}$ is a partial filter stability error for j-th pole of transmittance function, J is a number of poles of transmittance function, $|p_{i,j}|$ is a value of module for j-th pole of transmittance function, $PhaseErr_{i,q}$ is a partial filter minimal phase error for q-th zero of transmittance function, Q is a number of zeros of transmittance function, $|z_{i,q}|$ is a value of module for q-th zero of transmittance function.

5. Description of experiments

The four sixteen bit digital filters with non-standard amplitude characteristics were designed in order to test of quality of the proposed method. We have assumed following amplitude characteristics: linearly falling (a), linearly growing (b), non-linearly falling (c), and non-linearly growing (d). The parameters of there characteristics are as follows:

a) the attenuation should be equal to 0 [dB] for normalized frequency equal to 0, and the attenuation should be equal to 40 [dB] for normalized frequency equal to 1. The attenuation should linearly fall for remaining values of normalized frequency from in range (0; 1).

b) the attenuation should be equal to 40 [dB] for normalized frequency equal to 0, and the attenuation should be equal to 0 [dB] for normalized frequency equal to 1. The attenuation should linearly grow for remaining values of normalized frequency in the range (0; 1).

c) the attenuation should be equal to 0 [dB] for normalized frequency equal to 0, and the attenuation should be equal to 40 [dB] for normalized frequency equal to 1. The attenuation should non-linearly fall for remaining values of normalized frequency in the range (0; 1). The quadratic function is assumed as a non-linear function.

d) the attenuation should be equal to 40 [dB] for normalized frequency equal to 0, and the attenuation should be equal to 0 [dB] for normalized frequency equal to 1. The attenuation should linearly grow for remaining values of normalized frequency in the range (0; 1). The quadratic function is assumed as a non-linear function.

In the four designed filters, we have assumed, that the values of maximal admissible deviations of amplitude characteristics for any value of normalized frequency do not exceed values ± 0.5 [dB] ($Upper_{i,k} = 0.5$ [dB]; $Lower_{i,k} = -0.5$ [dB]).

In the designed digital filters, the assumed constraints are presented graphically (see Fig. 1 and Fig. 2) for better understanding.

Additionally, we have assumed, that normalized frequency was divided into 128 points ($R = 128$), and that 10-th order IIR digital filters are designed ($n = 10$). Also, we have assumed, that digital filters will be realized using 16 bits word ($nb = 16$) in Q.15 fractional format. The remaining parameters of evolutionary algorithm are as follows: number of individuals in population *PopSize* $= 100$, the initial value for proposed mutation operator $A_{init} = 1000$, maximal value of evolutionary algorithm generations $G_{\max} = 3000$, the probability of cross-over *PC* $= 0.7$, the probability of mutation $PM = 1/(2 \cdot n + 1)$.

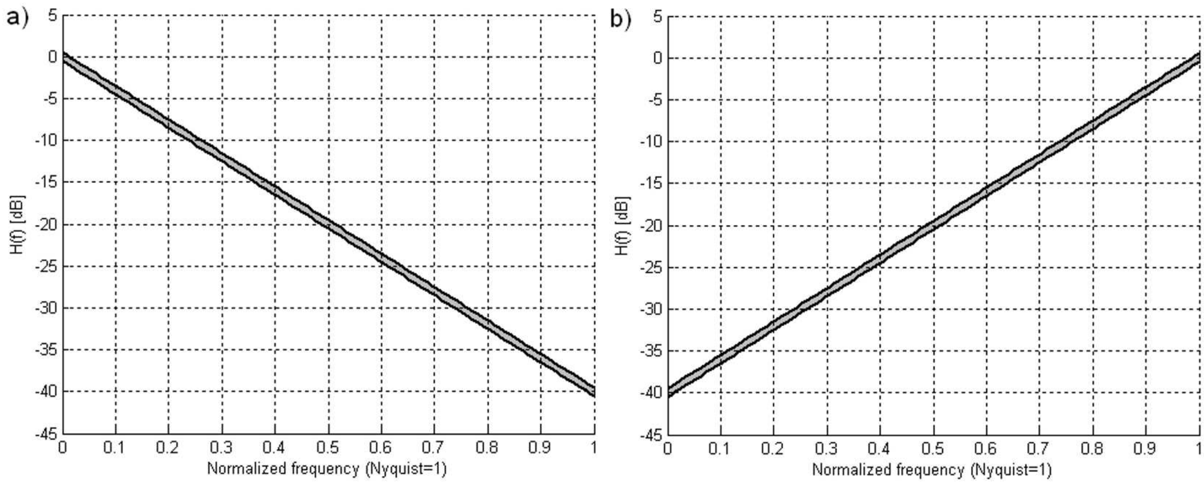

Fig. 1. Assumed constraints of amplitude characteristics for designed filters: linearly falling (a), linearly growing (b)

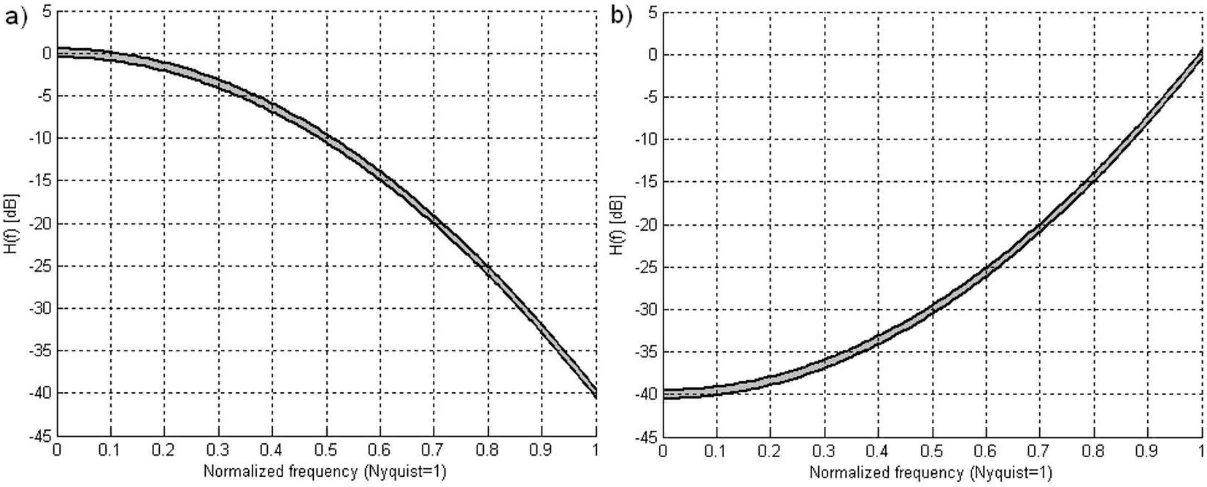

Fig. 2. Assumed constraints of amplitude characteristics for designed filters: nonlinearly falling (a), nonlinearly growing (b)

The computations were 20-fold repeated. The best solutions obtained using proposed method for four designed filters are as follows:

a) linearly falling amplitude characteristics
$TheBest$ ={3254, 4687, 4625, 4272, 2958, 2819, 2273, 1804, 848, 30, 65494, 51895, 14892, 61358, 63789, 9314, 58236, 5850, 56575, 2082, 79} (this solution was obtained after 2625 generations of the proposed algorithm)
b) linearly growing amplitude characteristics
$TheBest$ ={3255, 61213, 3091, 63701, 762, 65129, 482, 192, 65383, 65393, 1, 17869, 7160, 6648, 945, 65536, 1375, 7627, 8070, 4582, 1013} (this solution was obtained after 2280 generations of the proposed algorithm)
c) nonlinearly falling amplitude characteristics
$TheBest$ ={7057, 10299, 3417, 61474, 60691, 64330, 269, 102, 659, 633, 65536, 51962, 64600, 52399, 7668, 63355, 65391, 65536, 3212, 62547, 1717} (this solution was obtained after 1951 generations of the proposed algorithm)
d) nonlinearly growing amplitude characteristics
$TheBest$ ={1543, 64143, 637, 65400, 1, 65536, 1, 65535, 4, 2, 1, 31366, 423, 65153, 2921, 578, 65463, 658, 232, 9, 65323} (this solution was obtained after 2354 generations of the proposed algorithm).

The values in these solutions represent the index of the filter coefficient from the set D. The set D contains all allowable coefficient values for given bit word length (in this case it is assumed that the coefficients are in Q.15 fractional format). Based on transmittance function and property of Q.M format (in this case M is equal to 15), we assumed that the value of a_0 coefficient is equal to $1 - 2^{-M}$, therefore the value of this coefficient is not coded in the solutions. The values of coefficients are coded (in the solutions) as follows:

$$\mathit{TheBest} = \{b_0, b_1, \ldots, b_{n-1}, b_n, a_1, a_2, \ldots, a_{n-1}, a_n\}$$

Based on obtained solutions the values of coefficient (in Q.15 format) for four designed digital filters are as follows (the coefficient values are rounded to the four decimal places from editorial point of view). In the parenthesis the hexadecimal values of coefficients are presented. These hexadecimal values can be used in any DSP system with Q.15 fractional format.

a) linearly falling amplitude characteristics
b = {0.0993 (0CB5), 0.1430 (124E), 0.1411 (1210), 0.1303 (10AF), 0.0902 (0B8D), 0.0860 (0B02), 0.0693 (08E0), 0.0550 (070B), 0.0258 (034F), 0.0009 (001D), −0.0013 (FFD5)}
a = {-0.4163 (CAB6), 0.4544 (3A2B), −0.1275 (EFAD), −0.0533 (F92C), 0.2842 (2461), −0.2228 (E37B), 0.1785 (16D9), −0.2735 (DCFE), 0.0635 (0821), 0.0024 (004E)}
b) linearly growing amplitude characteristics
b = {0.0993 (0CB6), −0.1320 (EF1C), 0.0943 (0C12), −0.0560 (F8D4), 0.0232 (02F9), −0.0125 (FE68), 0.0147 (01E1), 0.0058 (00BF), −0.0047 (FF66), −0.0044 (FF70), 0 (0000)}
a = {0.5453 (45CC), 0.2185 (1BF7), 0.2029 (19F7), 0.0288 (03B0), −0 (FFFF), 0.0419 (055E), 0.2327 (1DCA), 0.2462 (1F85), 0.1398 (11E5), 0.0309 (03F4)}
c) nonlinearly falling amplitude characteristics
b = {0.2153 (1B90), 0.3143 (283A), 0.1042 (0D58), −0.1240 (F021), −0.1479 (ED12), −0.0368 (FB49), 0.0082 (010C), 0.0031 (0065), 0.0201 (0292), 0.0193 (0278), −0 (FFFF)}
a = {−0.4143 (CAF9), −0.0286 (FC57), −0.4009 (CCAE), 0.2340 (1DF3), −0.0666 (F77A), −0.0045 (FF6E), −0 (FFFF), 0.0980 (0C8B), −0.0912 (F452), 0.0524 (06B4)}
d) nonlinearly growing amplitude characteristics
b = {0.0471 (0606), −0.0425 (FA8E), 0.0194 (027C), −0.0042 (FF77), 0 (0000), −0 (FFFF), 0 (0000), −0.0001 (FFFE), 0.0001 (0003), 0 (0001), 0 (0000)}
a = {0.9572 (7A85), 0.0129 (01A6), −0.0117 (FE80), 0.0891 (0B68), 0.0176 (0241), −0.0023 (FFB6), 0.0201 (0291), 0.0070 (00E7), 0.0002 (0008), −0.0065 (FF2A)}

The value "−0" represents the case when very small negative value (near to zero) has been rounded to zero.

In Figs. 3–6, the amplitude characteristics and the phase characteristics are presented for the best results (the best designed digital filters obtained using proposed method).

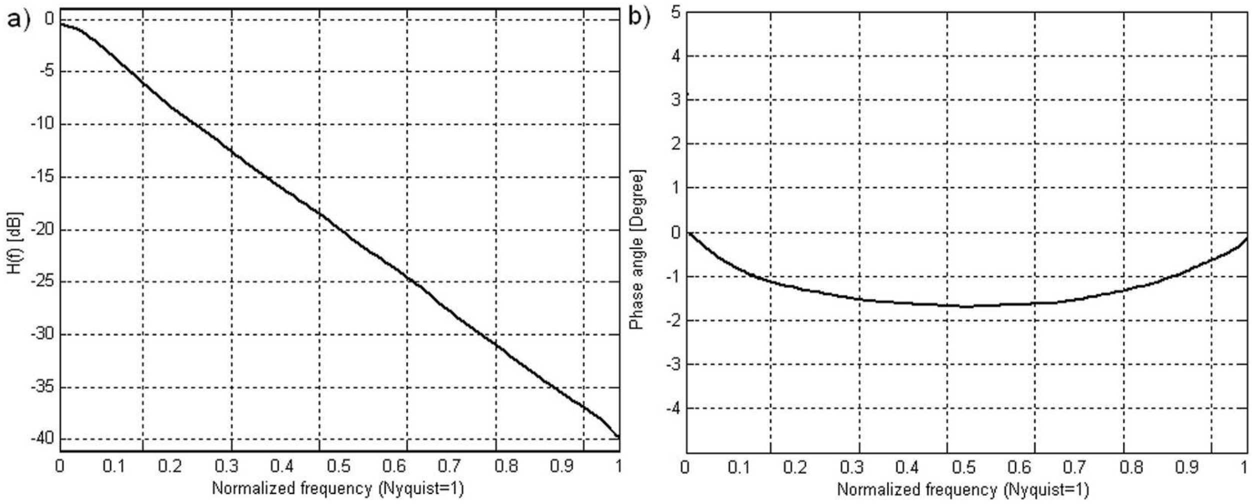

Fig. 3. Results obtained for linearly falling amplitude characteristics: amplitude (a), phase (b)

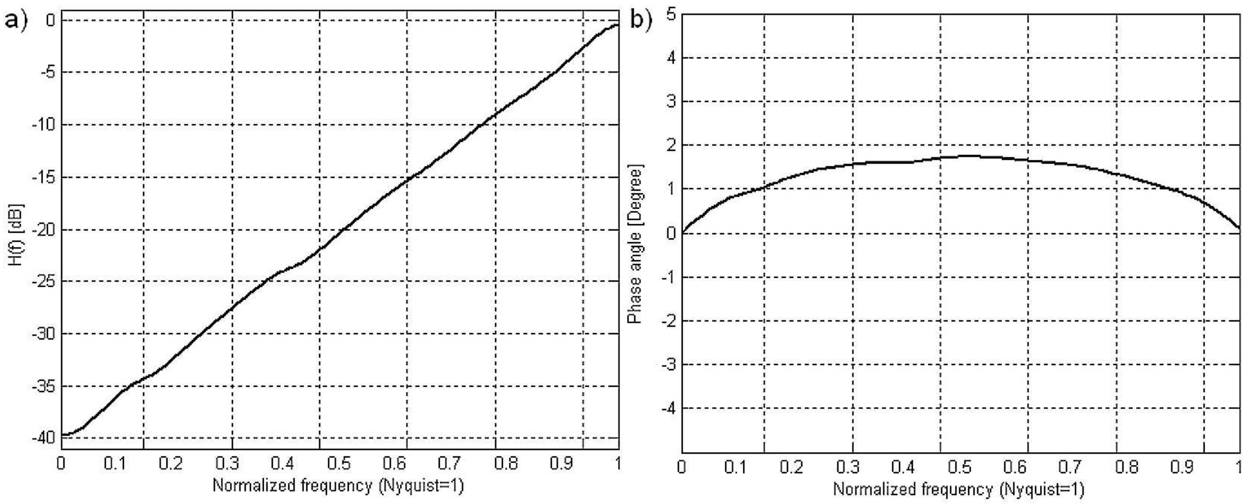

Fig. 4. Results obtained for linearly growing amplitude characteristics: amplitude (a), phase (b)

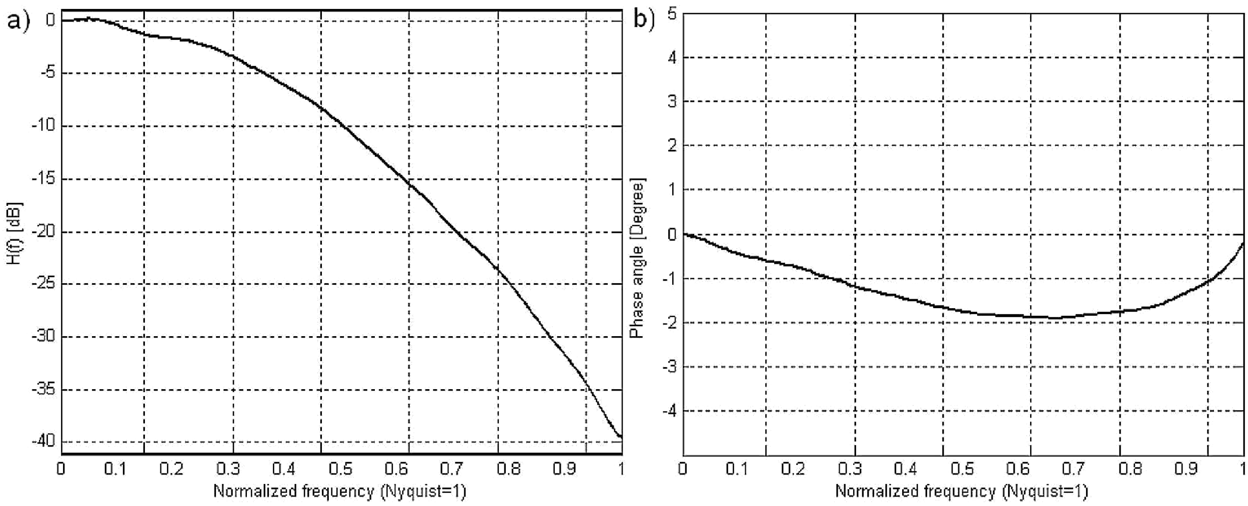

Fig. 5. Results obtained for nonlinearly falling amplitude characteristics: amplitude (a), phase (b)

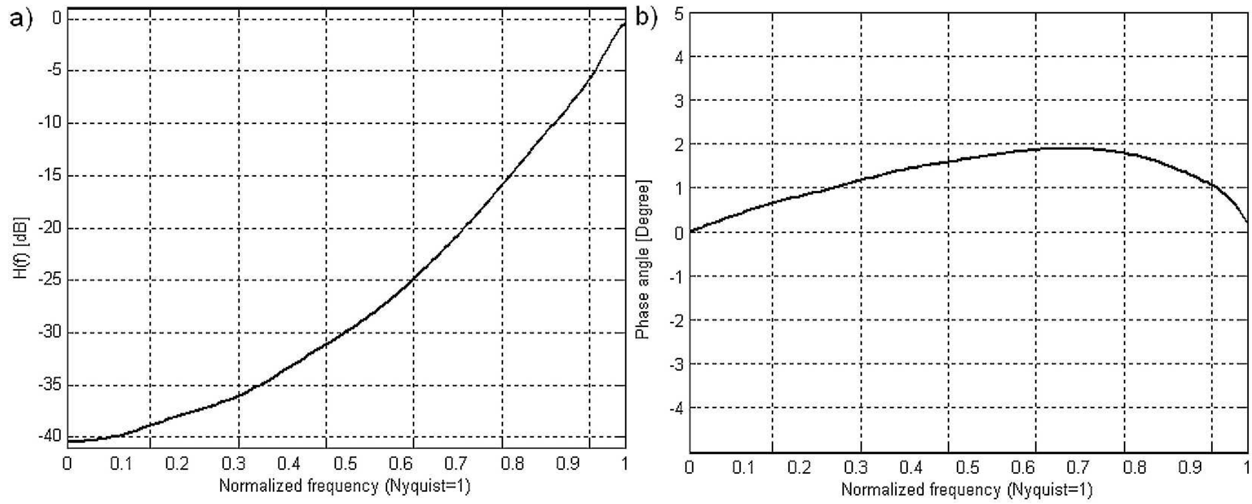

Fig. 6. Results obtained for nonlinearly growing amplitude characteristics: amplitude (a), phase (b)

From Figs. 3–6, it can be seen, that designed filters fulfills design assumption related to the shape of amplitude characteristics. The results presented in Figs. 3–6 where obtained after:

- 2625 generations, for linearly falling amplitude characteristics,
- 2280 generations, for linearly growing amplitude characteristics,
- 1951 generations, for nonlinearly falling amplitude characteristics,
- 2354 generations, for nonlinearly growing amplitude characteristics.

The shape of the phase characteristics was not optimized using proposed method. We have only guarantee that designed digital filters will be minimal phase filters (all zeros of transmittance function (1) must be located inside unitary circle in the z plane). From Figs. 3–6, we can see that in all designed filters, the phase difference is not higher than one degree.

In Figs. 7–10, the amplitude characteristics deviation for particular values of normalized frequency and for particular digital filters are presented for better illustration of obtained results. Also in Figs. 7–10, the locations of the poles and the zeros of designed filters are shown. In these figures, the poles are marked as a crossbar, and the zeros are marked as circle.

It can be seen from Figs. 7–10, that for each value of normalized frequency, the deviations of amplitude characteristics value are in the range ± 0.5 [dB]. Therefore, designed filters fulfill design assumptions. Also, we can see, that all designed filters are stable (the all poles of transmittance function are located inside unitary circle in z plane) and filters are minimal phase (the all zeros of transmittance function are located inside unitary circle in the z plane). Also, it is worth to say, that after implementation of these filters in DSP system, the properties of designed filter will not changed.

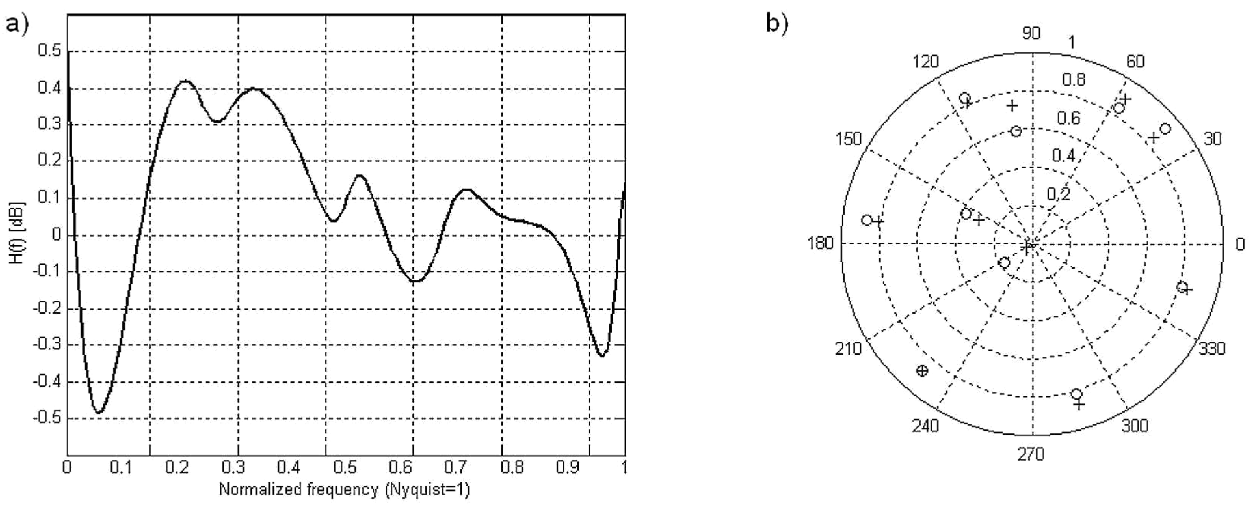

Fig. 7. Results obtained for linearly falling amplitude characteristics: deviations of amplitude characteristics (a), poles and zeros of transmittance function (b)

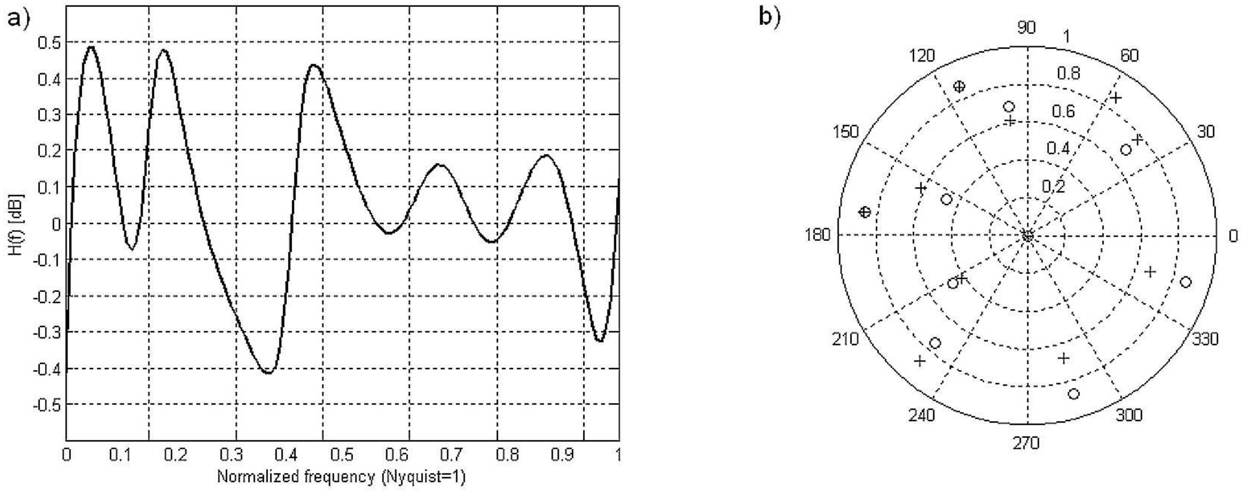

Fig. 8. Results obtained for linearly growing amplitude characteristics: deviations of amplitude characteristics (a), poles and zeros of transmittance function (b)

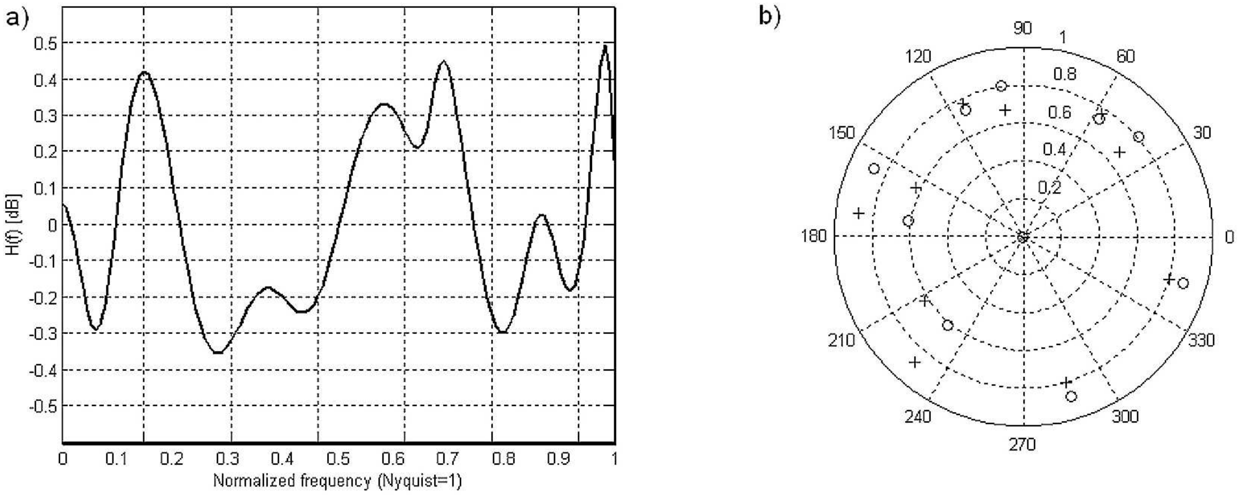

Fig. 9. Results obtained for nonlinearly falling amplitude characteristics: deviations of amplitude characteristics (a), poles and zeros of transmittance function (b)

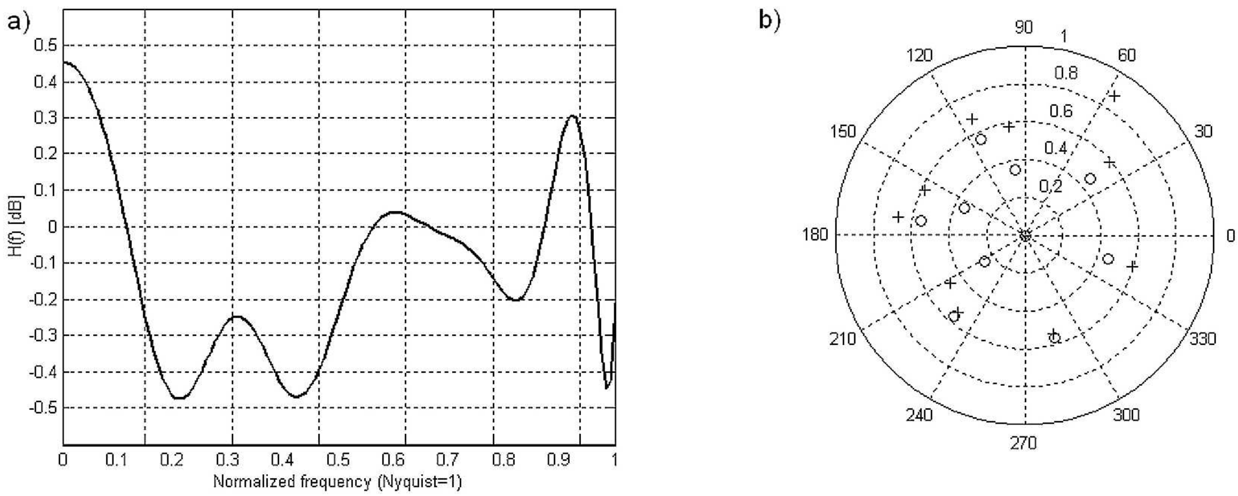

Fig. 10. Results obtained for nonlinearly growing amplitude characteristics: deviations of amplitude characteristics (a), poles and zeros of transmittance function (b)

Table 1
Statistics information about EA-MF-FWL-FD method based on four digital filters design problem

Amplitude characteristics	EA-MF-FWL-FD method			EA with simple mutation			EA with mutation from [27]		
	Success	Average	StdDev	Success	Average	StdDev	Success	Average	StdDev
LF	7	0.8590	1.1442	0	353.169	376.736	0	2432	343.346
LG	8	2.1180	5.5991	1	341.875	373.898	0	2532	415.005
NF	12	231.098	549.893	3	181.660	219.469	0	1902	478.796
NG	10	11.112	28.165	0	691.124	577.365	0	3245	276.831

Table 2
The value of objective function *FC* for different design methods

Amplitude characteristics	Amplitude deviation [dB]	EA-MF-FWL-FD method	Yule Walker method	Yule Walker method with quantization
LF	0.5	0	0.081076969560474	24.459167141254902
	0.3	0	0.407398087453072	26.216812464065253
	0.1	0.819665266512595	1.174695286145923	28.285095310635853
	0.0	3.548739842754208	4.361639969566997	32.163822161079366
LG	0.5	0	0.184043760619360	19.801865577763561
	0.3	0	0.504782475997226	21.734684701196549
	0.1	0.754945957089525	1.402429293845539	24.951029804909194
	0.0	5.472359567248908	13.323436033367559	36.853250942392926
NF	0.5	0	0.136833634987937	0.342125440492566
	0.3	0	0.382235527273240	0.710184568896176
	0.1	0	0.999571340937126	1.247135997953799
	0.0	2.518861974787978	5.453824290040547	5.345209503375336
NG	0.5	0	0	4.320034465860859
	0.3	0	0.162854189848984	5.544277816506174
	0.1	1.113221317833311	6.427077762231655	8.019947670209254
	0.0	7.148130356623089	16.031238233396323	17.447883460850040

In Table 1, the some statistics information about proposed method EA-MP-FWL-FD and evolutionary algorithm (EA) with simple mutation operator are presented. The symbols are as follows: $Success$ is the number of designed digital filters which fulfill all design assumptions (objective function FC equal to 0) after 20-fold repetition of each algorithm, $Average$ is an average value of the best values of objective function from 20-repetition of the each algorithm, $StdDev$ is a standard deviation value of obtained results, LF is a linearly falling amplitude characteristics, LG is a linearly growing characteristics, NF is a non-linearly falling characteristics, and NG is non-linearly growing characteristics.

It can be seen from Table 1, that EA-MF-FWL-FD algorithm gives better results than EA algorithm with simple mutation, and EA algorithm with mutation presented in paper [27]. From Table 1, we can see that 37 digital filters were designed using proposed method in 80-fold repetitions of the algorithm (20-fold repetition for each amplitude characteristics).

In the second experiment we compare results obtained using proposed method with results obtained using Yule Walker [20, 21] method. The minimal phase IIR digital filters with arbitrary amplitude characteristics can be designed using Yule Walker (YW) algorithm. But this algorithm not generate the coefficient which are ready to use in DSP system (coefficient in for example Q.15 format). Therefore, the coefficient obtained using YW method firstly must be scaled to the range $[-1;\ 1]$, and then must be quantized to Q.15 format. This intervention is not necessary if we use method proposed in this paper. In the table 2, the value of objective function FC for digital filters designed using: Yule Walker algorithm (YW), Yule Walker algorithm with scaled and quantized filter coefficients (QYW), and proposed EA-MF-FWL-FD method are presented. The designed digital filters fulfill all design assumptions, when the value of objective function FC is equal to 0. The results presented for EA-MF-FWL-FD method are the best results obtained using 10-fold repetition of EA-MF-FWL-FD algorithm. In Table 2, LF is a linearly falling amplitude characteristics, LG is a linearly growing characteristics, NF is a non-linearly falling characteristics, and NG is non-linearly growing characteristics.

From Table 2, it can be seen, that the result obtained using proposed method are better than results obtained Yule Walker method and much better than results obtained using Yule Walker method with scaling and quantization. If we want to implement a designed filter into hardware, then the results (digital filters) obtained using EA-MF-FWL-FD method will be not changed, but digital filters designed using Yule Walker method will be changed (compare the results obtained using Yule Walker algorithm with the results obtained using Yule Walker algorithm with quantization).

In the third experiments we have compare the computational time of EA-MF-FWL-FD method and Yule Walker method. The computational time for design of any digital filter (from first experiment) using Yule Walker method is approximately equal to 0.02 [s]. In the case of EA-MF-FWL-FD method, the computational time for one generation of the algorithm is approximately equal to 0.071 [s] (when population size is equal to 100). Therefore to design the filters from first experiments we need:

- 186.375 seconds, for linearly falling amplitude characteristics,
- 161.88 seconds, for linearly growing amplitude characteristics,
- 138.521 seconds, for nonlinearly falling amplitude characteristics,
- 167.134 seconds, for nonlinearly growing amplitude characteristics.

It can be seen, that the Yule Walker method is much faster than EA-MF-FWL-FD method. But only using proposed method, we can design digital filter which properties will not be changed after hardware implementation.

6. Conclusions

In this paper, the evolutionary method used to design minimal phase digital filters with non-standard amplitude characteristics and with finite bit word length is presented. In the paper, four digital filters were designed using proposed method. The designed filters are stable, minimal phase, and fulfill all design assumptions. In this paper, also adaptive mutation operator adapted to the considered problem is introduced. The results obtained using evolutionary algorithm with proposed mutation are better than results obtained using evolutionary algorithm with simple mutation and mutation presented in paper [27]. Also, the proposed method has been compared with Yule Walker method. The EA-MF-FWL-FD method is much slower than Yule Walker method. The main advantage of proposed method is that designed digital filters can be directly implemented in DSP systems without change of its properties. Also, we would like to noticed, that digital filters with coefficients in different arithmetic format (depend on DSP system) can be easily designed using proposed method, and always these filters are ready to be directly implemented into the hardware (without change of its properties). In conclusion, it is worth to say, that after small modifications, the proposed method can be used to design FIR digital filters.

Acknowledgements. The author thanks to the anonymous Reviewers for their valuable comments, which greatly helped to improve the contents of this paper.

REFERENCES

[1] R. Lyons, *Introduction to Digital Signal Processing*, WKL, Warsaw, 2000.

[2] A. Slowik and M. Bialko, "Evolutionary design of IIR digital filters with non-standard amplitude characteristics", *Proc. 3-rd Nat. Conf. Electronics* 1, 345–350 (2004).

[3] M. Erba, R. Rossi, V. Liberali, and A.G. Tettamanzi, "Digital filter design through simulated evolution", *Proc. ECCTD '01* 2, 137–140 (2001).

[4] S. Rainer and K.V. Price, "Differential evolution — a simple and efficient heuristic for global optimization over continuous spaces", *J. Global Optimization* 11, 341–359 (1997).

[5] K.V. Price, "An introduction to differential evolution", D. Corne, M. Dorigo, and F. Glover, editors, *New Ideas in Optimization*, pp. 79–108, McGraw-Hill, London, 1999.

[6] J. Kennedy, R.C. Eberhart, and Y. Shi, *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco, 2001.

[7] D.T. Pham, A. Ghanbarzadeh, E. Ko, S. Otri, S. Rahim, and M. Zaidi, "The bees algorithm – a novel tool for complex optimisation problems", *IPROMS 2006, Intelligent Production, Machines and Systems* 1, CD-ROM (2006).

[8] A. Slowik and M. Bialko, "Design and optimization of IIR digital filters with non-standard characteristics using continous ant colony optimization algorithm", *5th Hellenic Conf. on Artificial Intelligence, Lecture Notes in Computer Science* 5138, 395–400 (2008).

[9] K. Socha and M. Doringo, "Ant colony optimization for continous domains", *Eur. J. Operational Research* 185 (3), 1155–1173 (2008).

[10] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents", *IEEE Transactions on SMC*-B 26 (1), 29–41 (1996).

[11] H.G. Beyer and H.P. Schwefel, "Evolution strategies: a comprehensive introduction", *J. Natural Computing* 1 (1), 3–52 (2002).

[12] R.L. Becerra and C.A. Coello, "Cultured differential evolution for constrained optimization", *Computer Methods in Applied Mechanics and Engineering* 195 (33–36), 4303–4322 (2006).

[13] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company Inc., Boston, 1989.

[14] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, Berlin, 1992.

[15] J. Arabas, *Lectures on Evolutionary Algorithms*, WNT, Warsaw, 2001.

[16] N. Benvenuto, M. Marchesi, G. Orlandi, F. Piazza, and A. Uncini, "Finite wordlength digital filter design using an annealing algorithm", *Int. Conf. on Acoustics, Speech, and Signal Processing* 2, 861–864 (1989).

[17] M. Nakamoto, T. Yoshiya, and T. Hinamoto, "Finite wordlength design for IIR digital filters based on the modified least-square criterion in the frequency domain", *Int. Symposium on Intelligent Signal Processing and Communication Systems*, ISPACS 1, 462–465 (2007).

[18] N. Karaboga and B. Cetinkaya, "Performance comparison of genetic algorithm based design methods of digital filters with optimal magnitude response and minimum phase", *46th IEEE Midwest Symposium on Circuits and Systems* 1, CD-ROM (2003).

[19] N.D. Venkata and B.L. Evans, "Optimal design of real and complex minimum phase digital FIR filters", *IEEE Int. Conf. on Acoustics, Speech, and Signal Processing* 1, CD-ROM (1999).

[20] S.J. Orfanidis, *Introduction to Signal Processing*, Prentice-Hall, Englewood Cliffs, 1995.

[21] H. Ding, J. Lu, X. Qiu, and B. Xu, "Anadaptive speech enhancement method for siren noise cancellation", *Applied Acoustics* 65, 385-–399 (2004).

[22] *TMS320C54x DSP Library Programmer's Reference*, Texas Instruments, Dallas, 2001.
[23] *TMS320C55x DSP Library Programmer's Reference*, Texas Instruments, Dallas, 2002.
[24] W.S. Gan and S.M. Kuo, "Teaching DSP software development: from design to fixed-point implementations", *IEEE Trans. on Education* 49 (1), 122–131 (2006).
[25] G.S. Baicher, "Optimization of finite word length coefficient IIR digital filters through genetic algorithms – a comparative study", *Lecture Notes on Computer Science* 4222, 641–650 (2006).
[26] N. Karaboga and B. Cetinkaya, "Design of minimum phase digital IIR filters by using genetic algorithm", *Proc. 6th Nordic Signal Processing Symposium* 1, CD-ROM (2004).
[27] A. Slowik, "Evolutionary design of digital filters with non-standard amplitude characteristics and finite bit word length", *Proc. IX Nat. Conf. on Electronics* 1, 255–261 (2010).