A NEW FRAMEWORK FOR DESIGNING LINEAR-PHASE FIR FILTERS BASED ON EXTENDED LEAST SQUARE ERRORS

M. Saadatmand-Tarzjan[1], K. Mohamedpour [2]*

Department of Engineering, Tarbiat Modares University, Nasr Bridge, Tehran, Iran

Department of Electrical Engineering, K.N. Toosi University of Technology, Seyed Khandan, Tehran, Iran

ABSTRACT

In this paper a new iterative algorithm based on least square error method (LS) called extended LS (ELSE) is proposed to automatically design linear-phase FIR filters with any desired frequency response (DFR) and limited deviations. Minimizing a cost function that is defined based on the desired filter specifications including DFR and threshold of acceptable deviations (TAD), is the main idea of ELSE. In contrast to the other typical filter designing methods, defining the transition bands is not necessary in ELSE. Moreover, ELSE regulates the convergence rate by an automated approach that was utilized in neural networks; and it approximates the filter degree using an increasing-decreasing approach. In simulations, ELSE shows significant performance compared to LS though it may be more computationally complex. Also, it supplies the application demands in all of the case studies.

Keywords

Linear-Phase FIR Filters, Least Square Errors, Extended Least Square Errors, Steepest Descent Method.

1. INTRODUCTION

In this paper, an iterative new approach to design linear-phase FIR filters (called simply FIR filters hereafter) with desired frequency response (DFR) and limited deviations is introduced. The aim of the proposed algorithm is decreasing difference between frequency responses (FRs) of the designed and desired filters to below threshold of acceptable deviations (TAD) in each frequency. In more details, in each stage, the current filter is designed by LS using regulated FR (RFR). Then, the deviations of the current FR (CFR) are compared with TAD. At last, RFR is improved to compensate the overshoots.

It is assumed that DFR and TAD are specified in whole the frequency band. Therefore, in contrast to the other algorithms [1, 2], defining the transition bands is not necessary in the proposed approach. In fact, transition bands are defined to decrease the Gibbs effects or to use with Chebyshev approximation in designing. For example, a low-pass filter is

*Corresponding author: *E-mail:* kmpour@kntu.ac.ir

usually designed to remove noise from signal or distinct an appropriate part of it. The suitable pass and stop bands of the filter are determined by the FR of the signal. Almost, in all of the scientific applications, it is impossible to consider a real transition band between the pass and stop bands. In other words, the frequency bands of appropriate and non-appropriate parts of the signal are overlapped and consequently, indicating even one point instead of a transition band to distinct pass and stop bands is enough difficult. In fact, the transition bands are usually considered to remove or decrease the Gibbs effect and do not have a real physical explanation. The filter FR may have large oscillations in the transition bands since designing algorithms usually do not supervise FR in the transition bands. In this case, the designers may decide to improve the designing approach for avoiding these unwanted oscillations [3].

Moreover, someone may want to design a FIR filter with desired FR such as Gaussian or multi-modal. Most of the designing methods of FIR filters can not satisfy optimally these requests [4]. To use these methods in such applications, they should include more constraints such as limited deviations [5-7]. Employing the sophisticated and time-consuming optimization tools [8] such as linear programming [9, 10], simulated annealing [11], and genetic algorithm (GA) [12-14] is another solution. In this paper, the proposed method is developed to overcome the above drawbacks. Simulations show that the suggested approach is enough flexible and reliable to supply all requests of designing FIR filters.

The paper is organized as follows. In Section 2, the least square error approach to design linear-phase FIR filters is stated. Section 3 introduces the proposed algorithm. Section 4 is devoted to the convergence rate manipulation algorithm. In Section 5, the increasing-decreasing method to approximate the filter degree is stated. Simulation results are stated in Section 6. Finally, the paper is concluded in Section 7.

2. DESIGN CRITERIA OF SOSTTC

Let $h[n]$ be a linear-phase FIR filter from degree *2L*. Therefore, we have $h[n] = h[-n]$ (n=-*L*,…,-1,0,1,…,*L*) and its FR is computed as follows [15]:

$$H(\omega) = H(e^{j\omega}) = \sum_{n=0}^{L} k[n]\cos(n\omega) \qquad (1)$$

such that,

$$k[n] = \begin{cases} h[0] & n = 0 \\ 2h[n] & \text{Otherwise} \end{cases} \qquad (2)$$

$H(\omega)$ is an even periodic real function. Suppose the value of $H(\omega)$ in the frequencies $0 < \omega_i < 2\pi$ $(i=0,2,…,N\text{-}1)$ called designing frequencies be $A(\omega_i)$. LS computes the coefficients $k[n]$ and consequently $h[n]$ based on minimizing difference between $A(\omega_i)$ and $H(\omega)$. For this aim, the differentials of the cost function in Equation (3) with respect to $k[n]$ should be set to zero according to Equations (4) and (5).

$$E = \frac{1}{2}\sum_{i=0}^{N-1}[A(\omega_i) - H(\omega_i)]^2 = \frac{1}{2}\sum_{i=0}^{N-1}\left[A(\omega_i) - \sum_{n=0}^{L}k[n]\cos(\omega_i n)\right]^2 \tag{3}$$

$$\frac{\partial E}{\partial k[m]} = -\sum_{i=0}^{N-1}\left[A(\omega_i) - \sum_{n=0}^{L}k[n]\cos(\omega_i n)\right]\cos(\omega_i m) = 0 \quad \Rightarrow$$
$$m = 0,1,…,L \tag{4}$$

$$\begin{bmatrix} k[0] \\ k[1] \\ \vdots \\ k[L] \end{bmatrix} = P^{-1}\begin{bmatrix} \sum_{i=0}^{N-1}A(\omega_i) \\ \sum_{i=0}^{N-1}A(\omega_i)\cos(\omega_i) \\ \vdots \\ \sum_{i=0}^{N-1}A(\omega_i)\cos(L\omega_i) \end{bmatrix} \tag{5}$$

The matrix P in Equation (5) is computed as follows:

$$P = \begin{bmatrix} N & \sum_{i=0}^{N-1}\cos(\omega_i) & \cdots & \sum_{i=0}^{N-1}\cos(L\omega_i) \\ \sum_{i=0}^{N-1}\cos(\omega_i) & \sum_{i=0}^{N-1}\cos^2(\omega_i) & \cdots & \sum_{i=0}^{N-1}\cos(\omega_i)\cos(L\omega_i) \\ \vdots & \vdots & & \vdots \\ \sum_{i=0}^{N-1}\cos(L\omega_i) & \sum_{i=0}^{N-1}\cos(\omega_i)\cos(L\omega_i) & \cdots & \sum_{i=0}^{N-1}\cos^2(L\omega_i) \end{bmatrix}$$

Now, Let $N=2T$ and the designing frequencies are chosen as follows:

$$\omega_i = \frac{\pi(i-T)}{T}, \qquad i = 0,1,…,2T-1 \tag{6}$$

So we have:

$$P = \begin{bmatrix} N & 0 & \cdots & 0 \\ 0 & N/2 & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & N/2 \end{bmatrix} \tag{7}$$

Also, $A(\omega)$ is an even function as well as $H(\omega)$. Therefore, Equation (8) can be deduced from Equation (6). Additionally, Equation (9) is concluded from Equations (2) and (5)-(8).

$$\begin{bmatrix} \sum_{i=0}^{N-1}A(\omega_i) \\ \sum_{i=0}^{N-1}A(\omega_i)\cos(\omega_i) \\ \vdots \\ \sum_{i=0}^{N-1}A(\omega_i)\cos(L\omega_i) \end{bmatrix} = 2\begin{bmatrix} 1/2 & 1 & 1 & \cdots & 1 & 1/2 \\ -1/2 & \cos(\omega_1) & \cos(\omega_2) & \cdots & \cos(\omega_{T-1}) & 1/2 \\ \vdots & \vdots & \vdots & & \vdots & 1/2 \\ (-1)^L/2 & \cos(L\omega_1) & \cos(L\omega_2) & \cdots & \cos(L\omega_{T-1}) & 1/2 \end{bmatrix}\begin{bmatrix} A(\omega_0) \\ A(\omega_1) \\ A(\omega_2) \\ \vdots \\ A(\omega_{T-1}) \\ A(\omega_T) \end{bmatrix} \tag{8}$$

$$\begin{bmatrix} h[0] \\ h[1] \\ \vdots \\ h[L] \end{bmatrix} = \frac{2}{N}\begin{bmatrix} 0.5\cos(0\times\omega_0) & \cos(0\times\omega_1) & \cos(0\times\omega_2) & \cdots & \cos(0\times\omega_{T-1}) & 0.5\cos(0\times\omega_T) \\ 0.5\cos(1\times\omega_0) & \cos(1\times\omega_1) & \cos(1\times\omega_2) & \cdots & \cos(1\times\omega_{T-1}) & 0.5\cos(1\times\omega_T) \\ \vdots & \vdots & \vdots & & \vdots & \vdots \\ 0.5\cos(L\times\omega_0) & \cos(L\omega_1) & \cos(L\omega_2) & \cdots & \cos(L\omega_{T-1}) & 0.5\cos(L\times\omega_T) \end{bmatrix}\begin{bmatrix} A(\omega_0) \\ A(\omega_1) \\ A(\omega_2) \\ \vdots \\ A(\omega_{T-1}) \\ A(\omega_T) \end{bmatrix} \tag{9}$$

In other words, we have:

$$h[n] = \sum_{i=0}^{T}a_i A(\omega_i)\cos(n\omega_i), \qquad a_i = \begin{cases} 1/N & i = 0,T \\ 2/N & \text{Otherwise} \end{cases} \tag{10}$$

Equation (10) indicates the designing equations for linear-phase FIR filters using least square errors (LS), if the designing frequencies are selected according to Equation (6). Idiomatically, $A(\omega)$ is called regulated FR (RFR) of LS. In Section 3, it is shown that the designed filter can be improved by adjusting $A(\omega)$ based on a cost function during an iterative algorithm.

LS is an effective simple approach. Until now, researchers propose a wide variety of methods based on LS to include other constraints [5-7]. But, these extensions are not usually as flexible as LS in designing filters with desired FR. Generally, LS can be extended in three manners:

1   Improving the cost function,

2   Optimizing the designing frequencies and their number,

3   Optimizing $A(\omega)$.

The LS extensions are usually designed according to the first two manners while optimizing $A(\omega)$ can be another effective alternative.

3.   THE PROPOSED ALGORITHM

Let $I(\omega)$, $A(\omega)$, $H(\omega)$, and $\delta(\omega)$ indicate desired (ideal) FR (DFR), regulated FR (RFR), FR of the current filter that is designed by LS using $A(\omega)$ (CFR), and the threshold of acceptable deviations (TAD) of $H(\omega)$ from $I(\omega)$, respectively. Obviously, $H(\omega)$ has no error if and only if $|H(\omega)-I(\omega)|$ becomes less than $\delta(\omega)$ in whole frequency band. Consequently, designing $H(\omega)$ such that $|H(\omega)-I(\omega)|$ becomes less than   in whole frequency band as possible is the main purpose of our proposed algorithm.

In [16] and [17], an iterative algorithm is introduced to overcome above aim. According to Equation (11), in stage t of this method,   is changed in the opposite direction of invalid deviations of $H^t(\omega)$ from $I(\omega)$

$$A^t(\omega) = \begin{cases} I(\omega) & t=0 \\ A^{t-1}(\omega)\big[1-\eta\cdot\mathrm{sgn}\big(H^t(\omega)-I(\omega)\big)\cdot\max\big(|H^t(\omega)-I(\omega)|-\delta(\omega),0\big)\big] & \text{Otherwise} \end{cases} \tag{11}$$

As shown in Equation (11), $A^t(\omega)$ is changed oppositely in each frequency (ω) that $|H(\omega)-I(\omega)|>\delta(\omega)$. These frequencies are terminologically called invalid-deviation frequencies or simply invalid deviations. It is obvious that improving $A^t(\omega)$ suitably in the other frequencies near the invalid deviations can improve the performance of the designing algorithm. Our proposed algorithm develops this ability by minimizing the cost function of Equation (12).

$$J^t = \frac{1}{2}\sum_{\omega\in C}\big(|I(\omega)-H^t(\omega)|-\delta(\omega)\big)^2 \tag{12}$$

$$C^t = \left\{\omega_i \,\Big|\, i=0,1,\dots,T; \, |I(\omega_i)-H^t(\omega_i)|>\delta(\omega_i)\right\} \tag{13}$$

The set C^t that is defined in Equation (13) includes invalid deviation frequencies in stage t. Moreover, in Equation (12), ω_i indicates designing frequencies that are defined by Equation (6). According to Equation (14), in stage t, $A^t(\omega)$ is optimized based on steepest descent approach. Note that designing frequencies (ω_i) are fixed so, $A^t(\omega_i)$ (i=0,1,2,...2T-1) can be considered as $2T$ different variables instead of a functional.

$$A^t(\omega_i) = \begin{cases} I(\omega_i) & t=0 \\ A^{t-1}(\omega_i)-\eta^t\,\dfrac{\partial J^t}{\partial A^t(\omega_i)} & \text{Otherwise} \end{cases}, \qquad i=0,1,\dots,T \tag{14}$$

In the above equation, η specifies the convergence rate of the algorithm. The value of η has special importance. If a small value is selected for η, the computational time of the algorithm is increased and the algorithm captures with high probability in the local minima of the cost function. Additionally, selection of a large value for η causes to instability of the algorithm.

Equation (15) is concluded from Equations (1), (2), and (10):

$$H^t(\omega_p) = \sum_{i=0}^{T} a_i A^t(\omega_i)\left(1+2\sum_{n=1}^{L}\cos(n\omega_i)\cos(n\omega_p)\right), \qquad p=0,1,\dots,T \tag{15}$$

Then,

$$\frac{\partial H^t(\omega_p)}{\partial A^t(\omega_i)} = a_i \left(1 + 2 \sum_{n=1}^{L} \cos(n\omega_i)\cos(n\omega_p) \right) \tag{16}$$

So we have,

$$\frac{\partial J^t}{\partial A^t(\omega_i)} = a_i \sum_{\omega \in C} \text{sgn}\left(H^t(\omega) - I(\omega)\right)\left(\left|H^t(\omega) - I^t(\omega)\right| - \delta(\omega)\right)\left[1 + 2\sum_{n=1}^{L}\cos(n\omega_i)\cos(n\omega)\right] \tag{17}$$

Consequently, the proposed designing method can be summarized as follows,

1   is increased by an enough large number for example D=1000.

2   Specify filter degree (L), number of designing frequencies (N), initial convergence rate (η^0), maximum number of iterations (M), and threshold of cost function variations (ϑ_{min}).

3   Indicate DFR ($I(\omega_i)$) and TAD($\delta(\omega_i)$) in the designing frequencies.

4   Set t equal to zero.

5   The algorithm convergence rate is updated as follows:

$$\eta^t = f(t, \eta^{t-1}) \tag{18}$$

6   $A^t(\omega_i)$ is updated using Equations (14) and (17).

7   According to Equation (10), the filter coefficients ($h[n]$) are computed based on LS using $A^t(\omega_i)$.

8   The algorithm cost function is computed according to Equation (12).

9   If $\left|J^t - J^{t-1}\right| < \vartheta_{min}$ or $M < t$ then, it is referred to step 11.

10   Counter t is increased by one and then, it is referred to step 5.

11   The number D that is added to $I(\omega)$ in step 1 is subtracted from $h[0]$. Then, the algorithm is stopped.

As mentioned above, in the proposed approach, firstly, five different parameters including L, N, η^0, M, ϑ_{min}, $I(\omega)$, and $\delta(\omega)$ should be initialized. Filter degree (L) should be initialized by user however; we introduce an increasing-decreasing method to approximate L optimally in Section 5.

The number of designing frequencies (N) has an important effect on finally designed filter. If N is selected too small, the filter is designed without considering a large part of frequency band. Therefore, the frequency response of the designed filter may differ significantly in whole frequency band with respect to the designing frequencies. Additionally, if N is chosen too large, the computational time of the algorithm is considerably increased. However, N depends on the filter complexity but, simulations show that the proposed algorithm can satisfy all designing requests by N from range [100, 500]. In simulations, N is selected as 200.

The convergence rate (η^0) is firstly initialized by user. But, during iterations of the proposed algorithm, it is updated by Equation (18). It is preferred that a small or large value is not selected for η^0. Based on simulations, it seems 0.1 is a good

selection for η^0 in all cases. Specifically, in Section 4, a method for manipulating convergence rate is stated.

The maximum number of iterations (M) and the threshold of cost function variations (ϑ_{min}) specifies the stop criteria of the proposed algorithm that are experimentally set to 100 and 0.01, respectively.

Additionally, desired FR ($I(\omega)$) and the threshold of acceptable deviations ($\delta(\omega)$) are the filter specifications. As mentioned in step 1, $I(\omega)$ is firstly increased by a large number since $A(\omega)$ may become negative in some designing frequencies during convergence of the proposed algorithm. After convergence, the previously added number is subtracted from $h[0]$ according to step 11.

However, in the suggested algorithm, defining the transition bands is not necessary. But it is sufficient to select a large value for $\delta(\omega)$ in the transition bands to define them. Obviously, this kind of definition for transition bands is more reliable than other conventional designing methods since in this approach, the FR deviations are also investigated precisely in the transition bands.

If $\delta(\omega)$ is selected large in the whole frequency band, the proposed algorithm converges at the first iteration. In this case, the suggested algorithm that is called *extended least square errors* (ELSE) is simplified to LS.

4.   CONVERGENCE RATE MANIPULATION

Convergence rate (η) is a critical parameter for ELSE. Large convergence rate causes to instability while small one leads ELSE to local minima. For convergence rate manipulation, we apply the approach that was utilized in [18] to adjust training rate of neural networks. The effectiveness of this approach is also studied in several other applications [19, 20].

In the above approach, direction variations of correction vectors $A^t(\omega)$ in Equation 19, are considered as a measure for regulating the convergence rate.

$$\Delta \mathbf{A}^t = \left[\Delta A^t(\omega_1), \Delta A^t(\omega_2), \ldots, \Delta A^t(\omega_N)\right] \tag{19}$$

such that,

$$\Delta A^t(\omega_i) = A^t(\omega_i) - A^{t-1}(\omega_i) = -\eta^t \frac{\partial J^t}{\partial A^t(\omega_i)}, \tag{20}$$

$$t > 0, \ i = 0,1,\ldots,T$$

Considering correlation of two subsequent correcting vectors ($\Delta\mathbf{A}^t$ and $\Delta\mathbf{A}^{t-1}$), three different cases may appear:

1   If the correlation is near one, the consequent correcting vectors have approximately similar directions. Therefore, the cost function is flat and the convergence rate can be increased.

2   If the correlation is near minus one, the subsequent correcting vectors are approximately in the opposite directions. Therefore, the cost function is uneven and the convergence rate should be decreased.

3   If there is no correlation between the following correcting vectors, the current value of convergence rate is suitable.

Based on the above discussion, η is updated in each stage of ELSE by Equations (21)-(23). In other words, the function $f(\cdot)$ in Equation (18) is defined according to Equation (21).

$$\eta^t = \eta^{t-1}(1 + c \times r^t) \qquad (21)$$

Where,

$$c = \begin{cases} 0.25 & r^t \geq 0 \\ 0.5 & \text{Otherwise} \end{cases} \qquad (22)$$

$$r^t = \begin{cases} 0 & t = 1 \\ \dfrac{\Delta\mathbf{A}^t \cdot \Delta\mathbf{A}^{t-1}}{\left|\Delta\mathbf{A}^t\right| \times \left|\Delta\mathbf{A}^{t-1}\right|} & \text{Otherwise} \end{cases} \qquad (23)$$

In the above equations, r^t indicates the cosine of the angle between the current and previous correcting vectors ($\Delta\mathbf{A}^t$ and $\Delta\mathbf{A}^{t-1}$). As shown in Equation (21), η can be exponentially increased in a few iterations. Rapidly increasing of η may cause to some oscillations during convergence for decreasing η again [18]. This drawback is avoided by considering the cost function variations during convergence. In more details, if the current cost value (J^t) is greater than the previous one (J^{t-1}) by a fixed threshold such as $\vartheta_{stp} = 0.05$ then, the modifications of $A(\omega)$ in the previous iteration is ignored and η is divided by 2.

5.   FILTER DEGREE APPROXIMATION

The filter degree depends on complexity of DFR and TAD. The user usually specifies the filter degree through a trial and error procedure. We use an increasing-decreasing method to approximate the filter degree. In each step of the increasing phase, the filter degree (L_{inc}) is increased by λ_{inc} until a stop criterion is satisfied. Then, in the decreasing phase, the filter degree (L_{dec}) that is initialized by final L_{inc} is decreased one by one until the previous stop criterion is break again. The final L_{dec} (L_{final}) is a suitable approximation for optimal filter degree.

Figure 1 shows the incorporation of the increasing-decreasing method with ELSE. In each step of this algorithm, a filter is designed by ELSE and evaluated by the stop criterion. Moreover, each time, η^0 and $A(\omega)$ of ELSE are initialized by final η and $A(\omega)$ of the previous step. Additionally, the stop criterion is defined as follows:

$$(24)$$

$$\forall\omega_i: \quad \left|H(\omega_i) - I(\omega_i)\right| - \delta(\omega_i) < \theta_{\max}, \qquad i = 0, 2, \ldots, T$$

In more details, the designed filter is acceptable if and only if deviations of its FR in the designing frequencies are less than threshold θ_{max}.

As shown in Figure 1, increasing-decreasing approach converts ELSE to a powerful framework for designing any type of FIR filters with any desired FR. We experimentally offer 50 and 0.001 for λ_{inc} and θ_{max}, respectively.

6.   RESULTS AND DISCUSSIONS

All of the simulation results in this section are performed on an AMD^{TM} *Athlon 1600* MHz using Matlab^{TM} environment.

We compare ELSE with LS as a benchmark. The simulation results for three different filters including one low pass filter, one band stop filter, and one complicated filter are illustrated in Figures 2-4. Each figure includes the following parts:

a•   Design specifications including $I(\omega)$ (DFR) and $\delta(\omega)$ (TAD)

b•   FR of the filters ($H(\omega)$) that are designed by ELSE and LS

c•   Final $A(\omega)$ (RFR) of ELSE

d•   Errors of the ELSE and LS-designed filters that are computed by Equation (25):

$$(25)$$

$$e(\omega) = \text{sgn}\big(I(\omega) - H(\omega)\big) \cdot \max\big(\left|I(\omega) - H(\omega)\right| - \delta(\omega), 0\big)$$

e•   Variations of the cost function and convergence rate during ELSE convergence

Figures 2.d, 3.d, and 4.d show that the proposed algorithm is successful to decrease the invalid FR deviations in comparison with LS for different filters. In more details, the FR deviations of the ELSE-designed filters are significantly smaller than LS designations. Also, considering Figures 2.b, 3.b, and 4.b, it can be concluded that the valid deviations (which are below TAD) of ELSE-designed filters are increased in whole frequency band to decrease the invalid deviations (which are above TAD).

As illustrated in Figures 2.b, 3.b, and 4.b; especially near the sharp variations of DFR such as DFR edges, ELSE increases RFR in the opposite direction of invalid deviations to decrease and remove them.

Figures 2.e, 3.e, and 4.e show variations of the cost function and convergence rate during convergence of the proposed algorithm for different filters. ELSE converges uniformly and rapidly during only 20 iterations. This advantage is caused by convergence rate regulation algorithm. As shown in these figures, oscillations of the cost function reduce the convergence rate while its uniformly variations increase the convergence rate. This mechanism enables the proposed algorithm to increase the convergence rate as possible while avoiding the oscillations. Moreover, the simulations demonstrate that the increasing-decreasing algorithm is enough precise to approximate the filter degree. In more details, its estimations are seldom off by more than one. Indeed, ELSE can successfully supply all the user requests in designing linear-phase FIR filters.
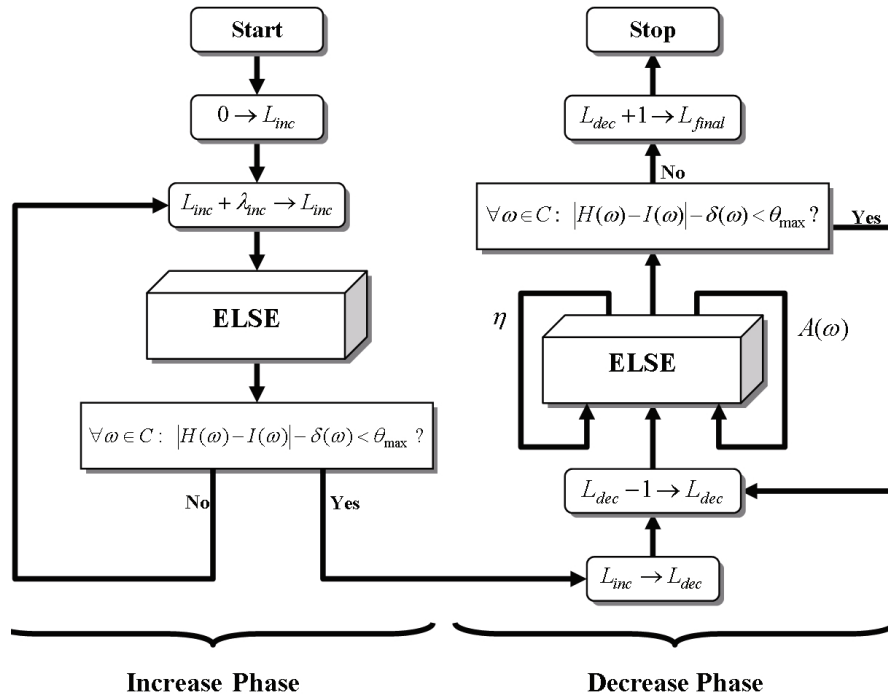
Figure 1. The block diagram of the filter degree approximation algorithm, it converts ELSE to a powerful framework to design any type of FIR filters with any desired FR and limited deviations
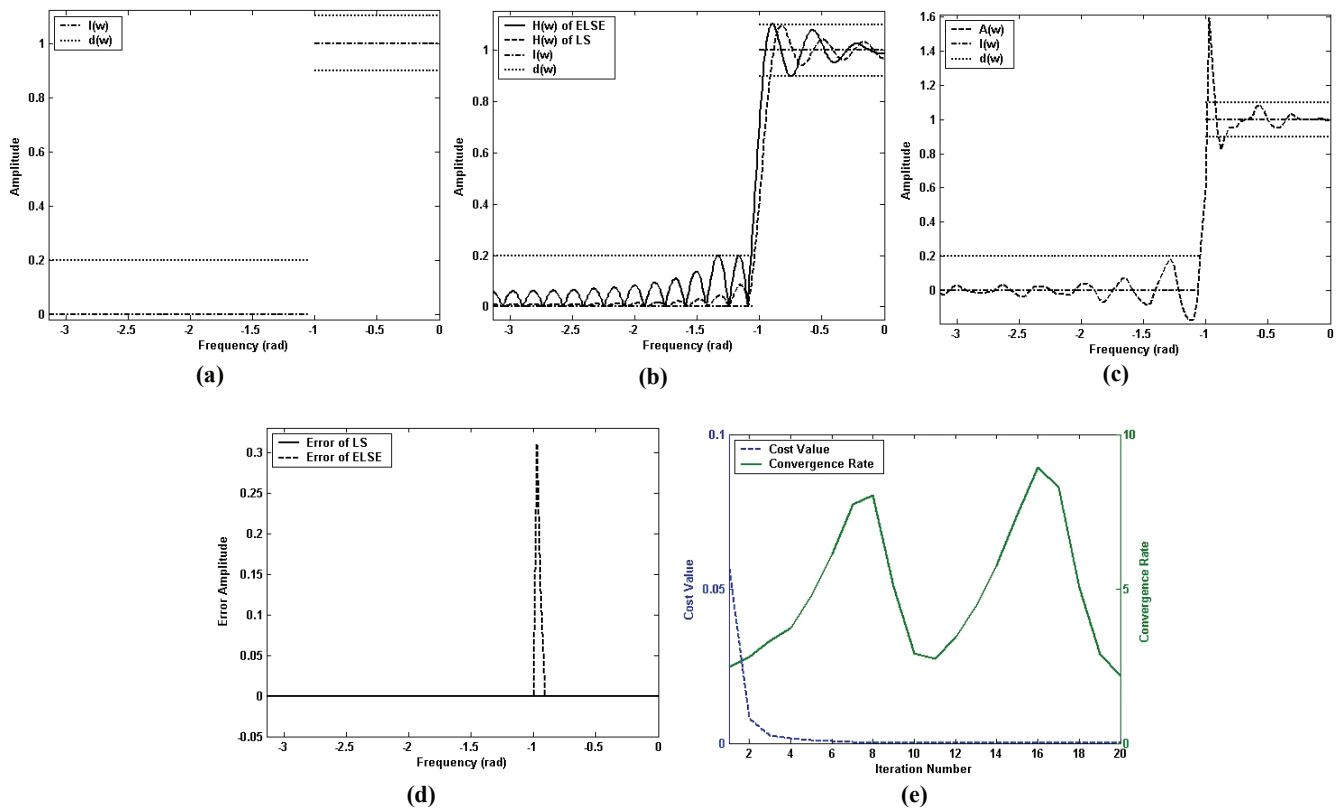

Figure 2. Designing one low-pass filter: (a) designing specifications including DFR and TAD, (b) FR of ELSE and LS-designed filters, (c) final RFR of ELSE, (d) the deviation errors of ELSE and LS-designed filters, (e) the cost function and convergence rate variations during ELSE convergence
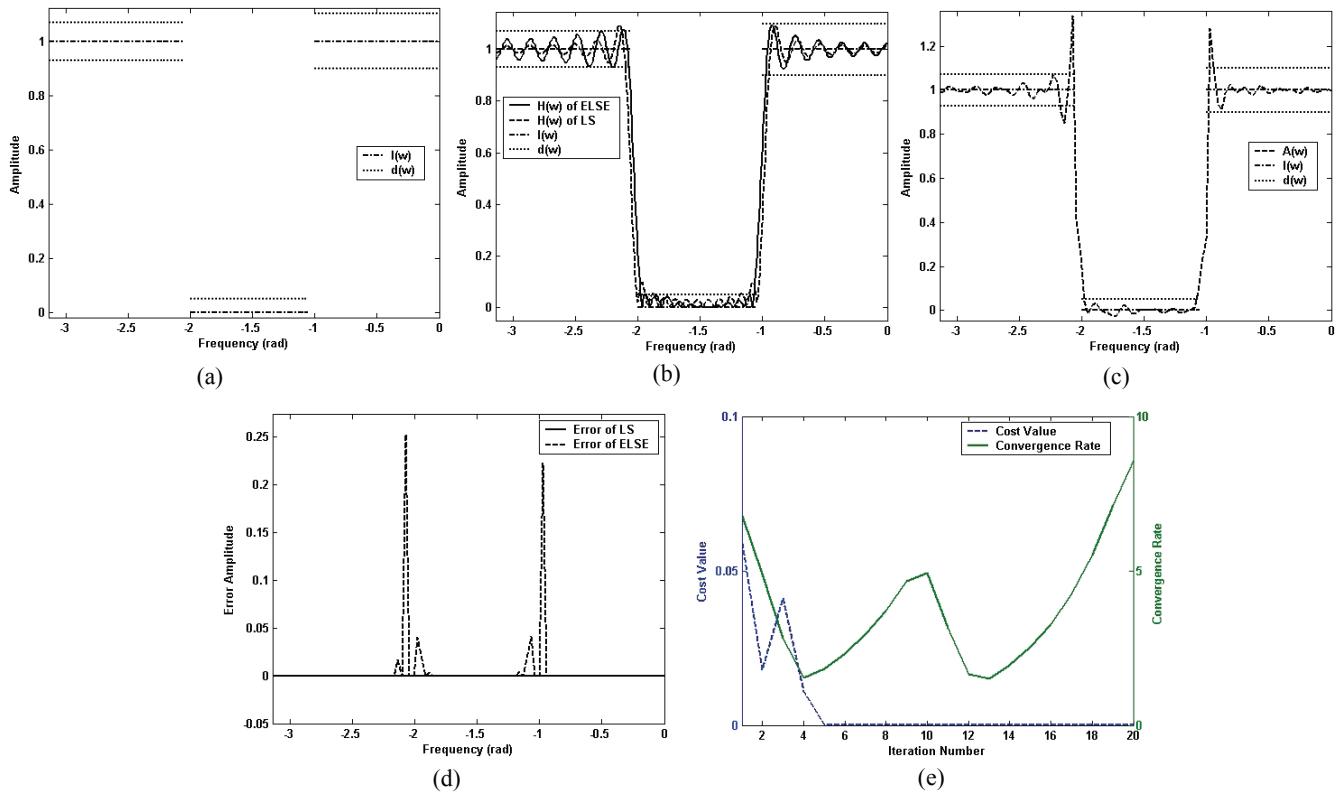
Figure 3. Designing one band-stop filter: (a) designing specifications including DFR and TAD, (b) FR of ELSE and LS-designed filters, (c) final RFR of ELSE, (d) the deviation errors of ELSE and LS-designed filters, (e) the cost function and convergence rate variations during ELSE convergence
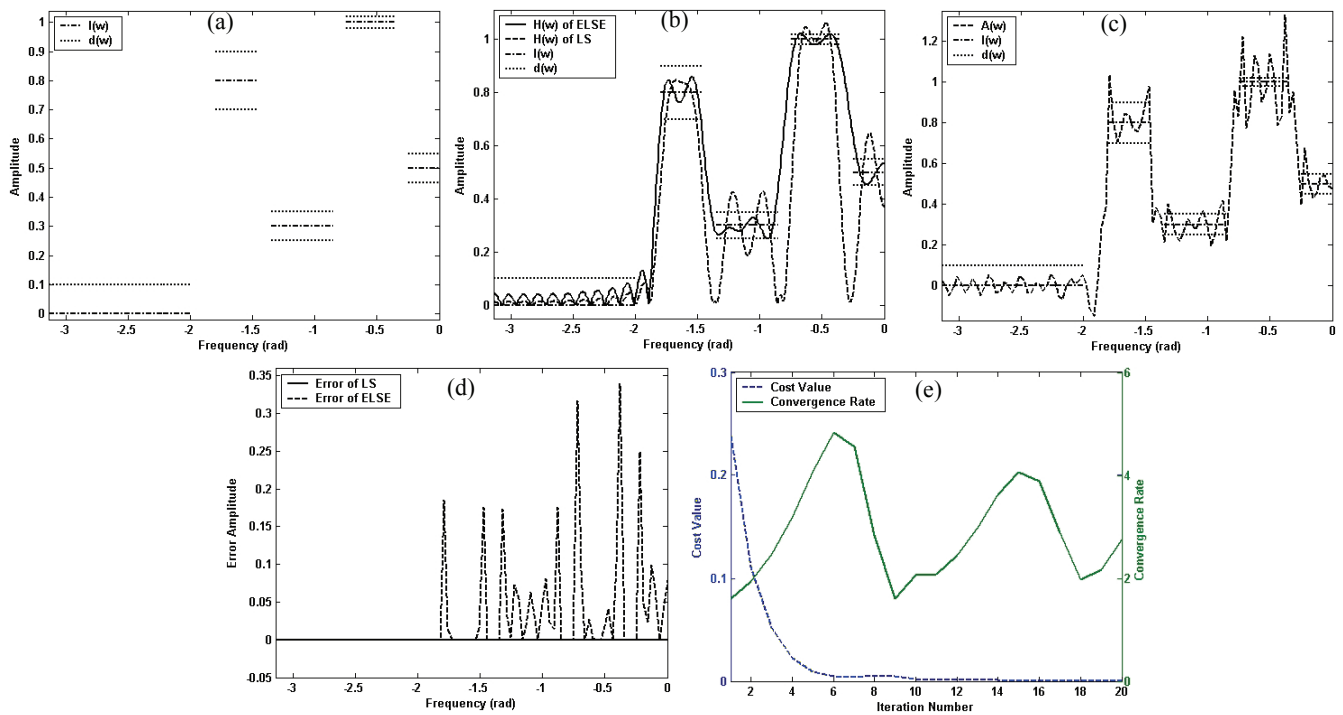

Figure 4. Designing one complicated filter with an irregular shape: (a) designing specifications including DFR and TAD, (b) FR of ELSE and LS-designed filters, (c) final RFR of ELSE, (d) the deviation errors of ELSE and LS-designed filters, (e) the cost function and convergence rate variations during ELSE

7. CONCLUSIONS

In this paper, a new iterative algorithm called ELSE to automatically design linear-phase FIR filters with any desired FR and limited deviations is introduced. ELSE extends LS method by optimally regulating its desirable frequency response. It adjusts the convergence rate by an automated algorithm that was utilized in neural networks; and approximates the filter degree using an increasing-decreasing approach. Also, in contrast to the other typical filter designing approaches, definition of the transition bands is not necessary (it is optional) in ELSE.

ELSE is an effective automated framework to design all types of linear-phase FIR filters with any desired FR and limited deviations. However, in spite of significant performance of ELSE, its computational complexity is heavier than the classical filter designing approaches such as LS. Therefore, ELSE may not be suitable for real-time applications.

The proposed algorithm will be applied to other similar applications such as designing filter banks in order to evaluate its performance compared to other conventional methods. Also, its computational complexity will be more studied

REFERENCES

[1] S. Wu, S. Boyd, & L. Vandenberghe, "FIR filter design via semidefinite programming and spectral factorization", Proc. 35th IEEE Conf. on Decision and Control, 1996, pp. 271-276.

[2] A.V. Oppenheim & R.W. Schafer, "Discrete-Time Signal Processing", 2nd ed., Englewood Cliffs, NJ: Prentice-Hall, UK, 1989, ISBN: 0-13-216292-X.

[3] I.W. Selesnick, M. Lang, & C.S. Burrus, "Constrained least square design of FIR filters without specified transition bands", IEEE Trans. on Signal Processing, Vol. 44, No. 8, Aug. 1996, pp. 1879-1892.

[4] H.H. Dam & K.L. Teo, "The dual parameterization approach to optimal least square FIR filter design subject to maximum error constraints", IEEE Trans. on Signal Processing, Vol. 48, No. 8, Aug. 2000, pp. 2314-2320.

[5] C.S. Burrus, "Multiband least squares FIR filter design", IEEE Trans. on Signal Processing, Vol. 43, No. 2, Feb. 1995, pp. 412-421.

[6] S.-C. Pei & C.-C. Tseng, "FIR filter design based on total least squares error criterion", Proc. IEEE Int'l Sym. on Circuits and Systems (ISCAS'99), Vol. 3, 1999, pp. 283-286.

[7] M.Z. Komodromos, S.F. Russell, P. Tak, & P. Tang, "Design of FIR filters with complex desired frequency response using a generalized Remez algorithm", IEEE Trans. on Circuits and Systems II: Analog and Digital Signal Processing, Vol. 43, No. 2, April 1995, pp. 412-421.

[8] D.W. Redmill, D.R. Bull, and E. Dagless, "Genetic synthesis of reduced complexity filters and filter banks using primitive operator directed graphs," IEE Proceedings on Circuits, Devices, and Systems, Vol. 147, No. 5, Oct. 2000, pp. 303-310.

[9] Y.C. Lim and S.R. Parker, "FIR filter design over a discrete powers-of-two coefficient space," IEEE Trans. Acoustic, Speech, and Signal Processing, Vol. 31, No. 3, Jun. 1983, pp. 583-591.

[10] H. Samueli, "An improved search algorithm for the design of multiplierless FIR filters with powers-of-two coefficients," IEEE Trans. Circuits and Systems, Vol. 36, No. 7, July 1989, pp. 1044-1047.

[11] N. Benvenuto, M. Marchesi, and M. Uncini, "Applications of simulated annealing for the design of special digital filters," IEEE Trans. Signal Processing, Vol. 40, No. 2, 1992, pp. 323-332.

[12] M. Oner, M. Askar, "Increasing design of high complexity FIR filters by genetic algorithms", Proc. 5th Int'l Sym. on Signal, Processing and its Applications (ISSPA'99), Brisbane, Australia, Vol. 2, 1999, pp. 1005-1008.

[13] M. Öner, "A genetic algorithm for optimization of linear phase FIR filter coefficients", Proc. 32nd Asilomar Conf. on Signals, Systems and Computers, 1998, pp. 1397-1400.

[14] D.R. Bull and A. Aladjidi, "The optimization of multiplier-free directed graphs: An approach using genetic algorithms," Proc. IEEE Int'l Sym. Circuits and Systems, Vol. 2, 1994, pp. 197-200.

[15] S. Haykin, "Adaptive filter theory", Englewood Cliffs, NJ: Prentice-Hall, UK, 1996, ISBN: 0-13-322760-X.

[16] K. Mohamed-pour & M. Saadatmand-T., "A new method for linear-phase FIR filter designing with desired shape," Proc. Int'l Sym. on Communication Systems, Networks, and Digital Signal Processing (CSNDSP'04), England, 2004, pp. 452-457.

[17] M. Saadatmand & Mohamed-pour, "A new iterative algorithm to design linear-phase FIR filters based on LS", Proc. 13th Iranian Conf. on Electrical Engineering (ICEE'05), Zanjan, Iran, Vol. 2, 2005.

[18] J.Y.F. Yam & T.W.S. Chow, "Extended least squares based algorithm for training feedforward networks", IEEE Trans. on Neural Networks, Vol. 8, No. 3, May 1997, pp. 806-810.

[19] M. Saadatmand-T., M. Khademi, "Color defect detection of ceramic tiles by a new color clustering algorithm", Proc. 2nd Iranian Conf. on Machine Vision and Image Processing (MVIP'03), Tehran, Iran, Vol. 1, 2003, pp. 108-115.

[20] M. Saadatmand-T., M. Khademi, "A new evolutionary color clustering algorithm to detect color defects of ceramic tiles", Proc. 5th Conf. on Intelligent Systems (CIS'03), Mashhad, Iran, 2003, pp. 228-235.

Biographies

MAHDI SAADATMAND-TARZJAN was born in Iran in 1980. He received his B.S. degree in electrical (control) engineering from Ferdowsi University, Mashad, Iran (2002) and his M.S. degree in biomedical engineering from K. N. Toosi University of Technology, Tehran, Iran (2005). He is currently pursuing the Ph.D. degree in biomedical engineering, Tarbiat Modares University, Tehran, Iran. His fields of interest are signal processing, image processing, machine vision, pattern recognition, and soft computing.

KAMAL MOHAMED-POUR was born in Iran in 1954. He received his B.S. degree from Telecommunication College (PTT), Tehran, Iran (1979), his M.S. degree from Tarbiat Modares University, Tehran, Iran (1987); and his Ph.D. degree from University of Manchester, UK (1996), all in electrical (telecommunication) engineering. He is currently working as associate professor of telecommunication engineering at K. N. Toosi University of Technology, Tehran, Iran. His major fields of interest are digital signal processing, communication systems, compression, and soft computing.