Design of 1-Dimentional FIR Filter using Modified Widrow-Hoff Neural Network

Amit Mishra
Jaypee University of Engineering and Technology
Guna, Madhya Pradesh, India

Khushboo Pachauri
Jaypee University of Engineering and Technology
Guna, Madhya Pradesh, India

Zaheeruddin
Jamia Millia Islamia (A Central University)
New Delhi, India

ABSTRACT

This paper is intended to provide an alternative optimization approach for the design of one-dimensional finite impulse response filter based on modified Widrow-Hoff neural network. This technique is based on minimization of weighted square-error function in frequency domain. Design guidelines and implementation approach was presented along with the proof of convergence theorem for the stability of neural network algorithm. Few examples which include single and multiband digital finite impulse response filters are presented; comparisons to existing methods are made. Computational complexity of various neural-based methods are also compared. As simulation results illustrates, the proposed neural network based method is capable of achieving an excellent performance for digital filter design.

Keywords:

FIR filter, weigted square-error function, modified Widrow-Hoff neural network, convergence theorem.

1. INTRODUCTION

Linear phase finite impulse response (FIR) digital filters are frequently used in signal processing applications because of their generalized stability and freedom from phase distortion. The problem of designing linear phase FIR filters has been studied extensively and solved in a number of different ways [1]−[4].
Much effort has been spent on designing filter based on windowing methods and frequency sampling [5]. The windowing method is the earliest and simplest approach of FIR filter designing. In this approach, a truncated ideal low pass filter having a certain bandwidth is generated, and then a chosen window is applied to achieve certain stop band attenuation. The use of windows offers very little design flexibility e.g. in low pass filter design, the pass band edge frequency generally cannot be specified exactly since the window smears the discontinuity in frequency.
In frequency sampling method, evenly spaced samples of a desired frequency response are created, and the IDFT is computed to obtain an impulse response. This method is useful for the design of non-prototype filters where the desired magnitude response can take any irregular shape but has drawback i.e. the frequency response obtained by interpolation is equal to the desired frequency response only at the sampled points whereas at the other points, there will be a finite error present. Because both the methods can not accurately control border frequencies of pass band and stop band in the practical application and are based on fixed formulation and not iterative, as a result, many researchers have presented some optimal design approaches.
Optimal FIR filter techniques were initiated in early 1970s. Mainly Remez Exchange Algorithm was the basis of these techniques and it has built for FIR filters. The algorithm proposed by Parks and McClellan [6], [7] uses the Remez Exchange method to find the optimal approximation for the magnitude response. After that, many algorithms have been developed based on Linear Programming (LP) [8], [9], Quadratic Programming and Heuristic methods in Artificial Intelligence (AI) Tools, such as Neural Networks [10]−[15].
Remez Exchange Algorithm and linear Programming are optimum in the sense that these methods achieve both a given discrimination and a specified selectivity with a minimum length of the filter impulse response. Unfortunately both the schemes are computationally intensive as the filter length is increased. The weighted least-square (WLS) methods [16], [17] show much flexible utilization for any type of filter design analytically but these approaches are typically based on linear algebra methods that requires computationally intensive matrix inversion.
In this paper, an efficient method based on neural networks proposed for the design of linear phase FIR filters. The design problem was formulated based on the approximation of magnitude response using modified weighted Widrow-Hoff neural network architecture.
In Sect. 2, motivation and some properties of modified Widrow-Hoff neural network are briefly reviewed in order to perform the filter design problem along with the linear transformation ability of proposed network. Then a convergence theorem and algorithm to implement the FIR digital filters using ANN is proposed along with a comparison of computational complexity of proposed method with other neural network based models in Sect. 3. In Sect. 4, the designed examples and simulated results are described to demonstrate the effectiveness of the proposed algorithm. Finally, the conclusions are stated in Sect. 5.

2. FILTER DESIGN USING MODIFIED WIDROW-HOFF NEURAL NETWORK (MWHNN)

The standard Widrow-Hoff neural network described as ADALINE (ADAptive LInear NEuron) is based on Least Mean Square (LMS) algorithm.
Proposed model is a modified version of the standard Widrow-Hoff network in the sense that this does not include the bias term. If a single layer artificial neural network with no bias input has

Table 1. The parameters of the four types of filter

Type	I	II	III	IV
s	s=0	s=0	s=1	s=1
N	Odd	Even	Odd	Even
M	$\frac{N-1}{2}$	$\frac{N}{2}$	$\frac{N-1}{2}$	$\frac{N}{2}$
n_0	0	1	1	1
a_n	$\begin{cases} h(M), & n=0 \\ 2h(M-n), & 1 \le n \le M \end{cases}$	$2h(M-n)$, $1 \le n \le M$	$2h(M-n)$, $1 \le n \le M$	$2h(M-n)$, $1 \le n \le M$
$\varphi_n(\omega)$	$cos(n\omega)$	$cos(n-\frac{1}{2})\omega$	$sin(n\omega)$	$sin(n-\frac{1}{2})\omega$

a linear transfer function, then the transformation from the input vector to the output vector is a linear transformation. This makes our proposed model suitable for filter design with the capability of linear transformation [18].

2.1 Motivation of Filter Design

The frequency response of a linear phase digital filter [5] is given as

$$\mathbf{H}(\omega) = \sum_{n=0}^{N-1} h(n)e^{-j\omega n} = e^{-j\omega \frac{N-1}{2}} e^{js\frac{\pi}{2}} \mathbf{A}(\omega), \qquad (1)$$

where $h(n)$ ($n = 0$ to $N-1$) is the impulse response, ω is frequency, N is the filter length, and

$$s = \begin{cases} 0, & \text{if } h(n) \text{ is symmetric} \\ 1, & \text{if } h(n) \text{ is antisymmetric} \end{cases}$$

The freuency response of real-valued amplitude response $A(\omega)$ can be expressed as the general form,

$$\mathbf{A}(\omega) = \sum_{n=n_0}^{M} a_n\, \varphi_n(\omega) \qquad (2)$$

where a_n and $\varphi_n(\omega)$ are the filter coefficient vector and appropriate trigonometrical function respectively.
These parameters can be divided into four types of filters, according to whether the filter length is even or odd and whether the impulse response is symmetric or antisymmetric. All the results are shown in Table 1. From Eq. (2), $\mathbf{A}(\omega)$ for type I filter [5] can be expressed as

$$\mathbf{A}(\omega) = \sum_{n=n_0}^{M} a_n\, \varphi_n(\omega) = \sum_{n=n_0}^{M} a_n\, cos(n\omega) \qquad (3)$$

where M=$\frac{N-1}{2}$ and n_0=0. Now, sample uniformly $\mathbf{A}(w)$ in frequency axis to get its discrete values $\mathbf{A}(w_l)$.

$$\mathbf{A}(\omega_l) = \sum_{n=0}^{M} a_n cos(n\omega_l) = \mathbf{a}\, \Phi_n(\omega_l) \qquad (4)$$

where $\mathbf{A}(\omega_l)$ is the magnitude response corresponding to the sampling point at w_l and L is the number of point sampled between 0 to π i.e. $w_l \in [0-\pi]$. The sampling point (ω_l) can be expressed as $\omega_l = \frac{l}{L}\pi$, where $l = 0, 1, \ldots L$. The trigonometric function matrix $\Phi_n(\omega_l)$ can be evaluated as-

$$\Phi_n(\omega_l) = \mathbf{b} = \begin{bmatrix} 1 & cos(\omega_1) & \ldots & cos(n\omega_1) \\ 1 & cos(\omega_2) & \ldots & cos(n\omega_2) \\ \vdots & \vdots & \vdots & \vdots \\ 1 & cos(\omega_L) & \ldots & cos(n\omega_L) \end{bmatrix} \qquad (5)$$

Now, Eq. (4) can be written as

$$\mathbf{A}(\omega_l) = \mathbf{b}^T\mathbf{a} \qquad (6)$$

The error response is expressed as

$$\mathbf{e}_k = \mathbf{A}_d - \mathbf{A}(\omega_l) \qquad (7)$$

where $\mathbf{A}_d$ is desired magnitude response.
In this work, the performance index is chosen as weighted square error function (**P**) to converge the training algorithm at its minima and provide stability to proposed neural network model.
The proposed performance index **P** (weighted square−error function) can be defined as

$$\mathbf{P} = \frac{1}{W}\sum_{l=1}^{L} W(l)\mathbf{e}_k^2(l) \qquad (8)$$

where $W(l)$ is a weight coefficient ($W(l) > 0$), and $W = \sum_{l=1}^{L} W(l)$.
In Eq. (8) the weight factor $\frac{\sum_{l=1}^{L} W(l)}{W}$ is multiplied with sqaure error term to make performance index **P** as a weighted square−error function.
To minimize the performance index of proposed Widrow-Hoff neural network the concept of Least Mean Square (LMS) algoritm is used. The number of neuron in hidden layer are $M+1$ (i.e. $\frac{N+1}{2}$). The weight vector of hidden layer is denoted by **a** and recursively updated as

$$\mathbf{a}_{k+1} = \mathbf{a}_k - \eta\frac{\partial \mathbf{P}}{\partial \mathbf{a}_k} = \mathbf{a}_k + 2\,\eta\,\mathbf{b}\,\mathbf{e}_k\,\frac{\mathbf{W}_d}{W} \qquad (9)$$

where $\eta > 0$ is the learning rate and $\mathbf{W}_d$ can be expressed as

$$\mathbf{W}_d = \begin{bmatrix} W(1) & 0 & \ldots & 0 \\ 0 & W(2) & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \ldots & 0 & W(M) \end{bmatrix} \qquad (10)$$

The term $\frac{\mathbf{W}_d}{W}$ in Eq. (9) is a weight factor which depends on the selection of weight coefficient vectors using Eq. (18).

2.2 Proposed Neural Network Model

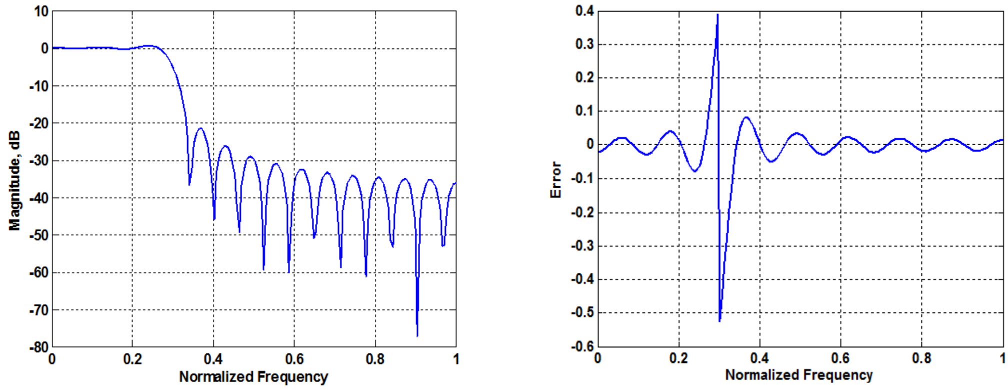
The proposed model for FIR digital filter design is shown as in Fig. 1, where **a** and **b** are the input vector and weight matrix of the NN model respectively. $\mathbf{A}(\omega_l)$ is the amplitude response of the model and $\mathbf{e}_k$ is the error of network, where $\mathbf{A}_d$ represents the desired magnitude response. Here, L and n are the number of points sampled at frequency axis and required number of hidden neuron in the ANN model respectively.
During training phase the error $\mathbf{e}_k$ is evaluated as the difference of desired and current amplitude response of the neural network and using error back propagation algorithm the weights are updated with appropriate choice of learning rate η .

3. CONVERGENCE AND COMPUTATIONAL COMPLEXITY ANALYSIS

In order to ensure the convergence of proposed model, it is important to select appropriate value of learning rate η. In this sec-

Table 2. Computational complexity involved in NN based models

Operation $\longrightarrow$ Algorithm $\downarrow$	Integer multiplication	Integer addition	Sigmoid function
Conventional Least−square method [16]	$\approx n^2$ L	–	–
Bhattacharya and Antoniou method [10]	$2nL(1+n)+n(2L+1)$	$2nL(1+n)+n(2L+1)$	$n+2L+n(2L+1)$
Yuo-Dar Jou method [14]	$n(1+n)$	$n(1+n)$	n
Neural network optimization method [19]	$n(1+n)$	$n(1+n)$	n
Proposed algorithm	$n(n)$	$n(n-1)$	–



(a) (b)

Fig. 2. Design of low-pass FIR filter using modified Widrow-Hoff NN with filter length N=31, ω_p=0.3 π, ω_s=0.3 π, and sampling grid L=180. (a) Amplitude response, (b) Designed error response (Normalized to π)

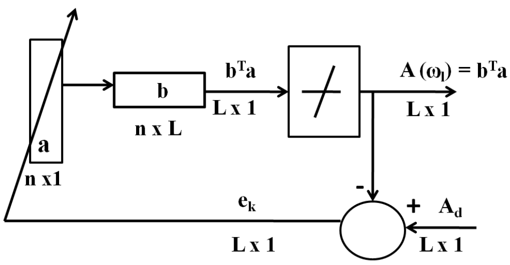


Fig. 1. Proposed modified Widrow-Hoff Neural Network (MWHNN) model

tion a proof of convergence theorem is presented to show the convergence and stability of the proposed neural network algorithm. The computational complexity involved in various neural network based models is also discussed.

3.1 Convergence Theorem

THEOREM 1. *Algorithm of the neural network is convergent, if learning rate satisfies* $0 < \eta < \frac{2W}{(N+1)W(l)}$, *where* $W(l) > 0$, N *is odd integer,* $N-1$ *is the order of FIR filter and* $\frac{N+1}{2}$ *is the number of hidden neurons used in neural network.*

PROOF. Lets define performance index **P** as a Lyupunov function shown in Eq. (8).
Therefore, $\triangle\mathbf{P}$ can be expressed as

$$\triangle\mathbf{P} = \mathbf{P}_{k+1} - \mathbf{P}_k = \frac{1}{W}\sum_{l=1}^{L} W(l)\left[e_{k+1}^2(l) - e_k^2(l)\right], \quad (11)$$

where l=1, 2, ... L.
The term $\triangle e_k(l)$ can be expressed as

$$\triangle e_k(l) = \left[\frac{\partial e_k(l)}{\partial \mathbf{a}}\right]^T \triangle\mathbf{a} \quad (12)$$

from Eq. (9), we get,

$$\triangle\mathbf{a} = -\eta\frac{2}{W}W(l)e_k(l)\frac{\partial e_k(l)}{\partial \mathbf{a}} \quad (13)$$

Therefore Eq. (12) can be expressed as -

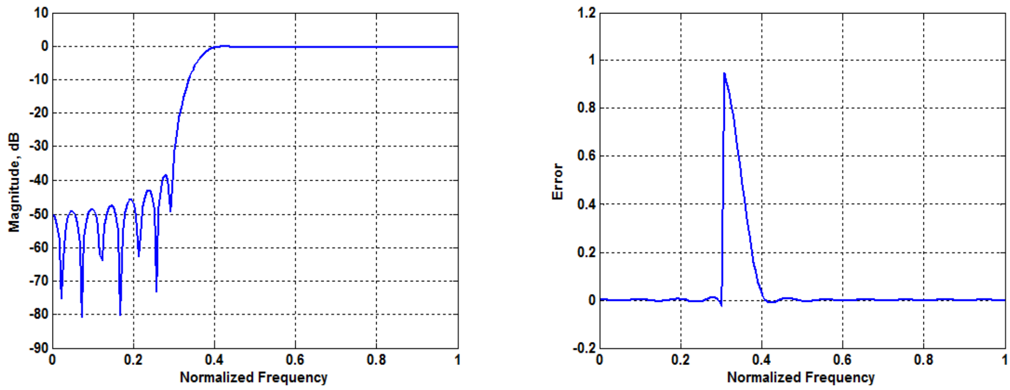
$$\begin{aligned}\triangle e_k(l) &= -\eta\frac{2}{W}W(l)e_k(l)\left[\frac{\partial e_k(l)}{\partial \mathbf{a}}\right]^T\left[\frac{\partial e_k(l)}{\partial \mathbf{a}}\right] \\ &= -\eta\frac{2}{W}W(l)e_k(l)\|\frac{\partial e_k(l)}{\partial \mathbf{a}}\|^2 \quad (14)\end{aligned}$$

where $\|\cdot\| = \sum|\cdot|^2$ is the square of Euclidean norm.
From Eq. (11)

$$\begin{aligned}\triangle\mathbf{P} &= \frac{1}{W}\sum_{l=1}^{L} W(l)\left[\{e_k(l) + \triangle e_k(l)\}^2 - e_k^2(l)\right] \\ &= \frac{4}{W^2}\sum_{l=1}^{L} W^2(l)e_k^2(l)\|\frac{\partial e_k(l)}{\partial\,\mathbf{a}}\|^2 \\ &\quad \left[-\eta + \eta^2\frac{1}{W}W(l)\|\frac{\partial e_k(l)}{\partial\,\mathbf{a}}\|^2\right] \quad (15)\end{aligned}$$

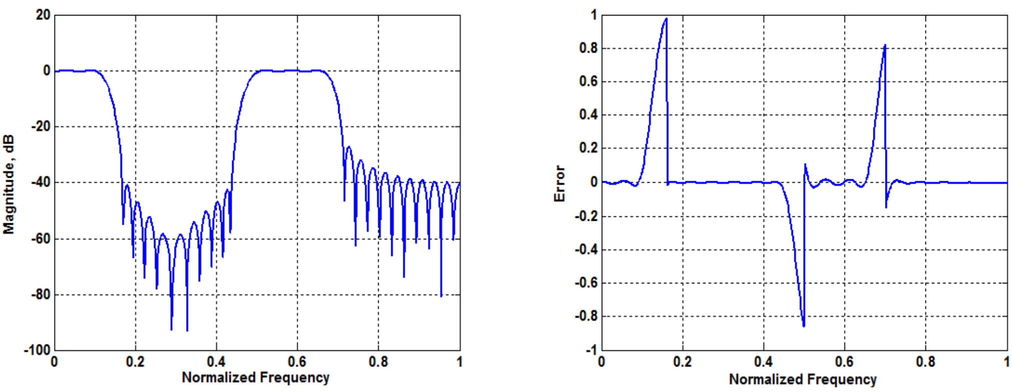
From Eq. (15), if

$$-\eta + \eta^2\frac{1}{W}W(l)\|\frac{\partial e_k(l)}{\partial\,\mathbf{a}}\|^2 < 0 \quad (16)$$



(a) (b)

Fig. 3. Design of high-pass FIR filter using modified Widrow-Hoff NN with filter length N=41, ω_s=0.3017 π, ω_p=0.3994 π, and sampling grid L=180. (a) Amplitude response, (b) Designed error response (Normalized to π)



(a) (b)

Fig. 4. Design of multiband FIR filter using modified Widrow-Hoff NN with filter length N=65, $\omega_p = \{0 \le \omega \le 0.1\pi\} \cup \{0.5\pi \le \omega \le 0.66\pi\}$, $\omega_s = \{0.16\pi \le \omega \le 0.44\pi\} \cup \{0.7\pi \le \omega \le \pi\}$, and sampling grid L=501. (a) Amplitude response, (b) Designed error response (Normalized to π)

the term $\triangle \mathbf{P} \le 0$ and algorithm is convergent. Therefore according to Eq. (4) and (7)

$$\begin{aligned} \|\frac{\partial e_k(l)}{\partial\, \mathbf{a}}\|^2 &= \|\frac{\partial e_k(l)}{\partial A(\omega_l)} \frac{\partial A(\omega_l)}{\partial\, \mathbf{a}}\|^2 \\ &= \sum_{i=0}^{M} |\Phi_n(\omega_l)|^2 = \frac{N+1}{2} \end{aligned} \quad (17)$$

by substituting the value of $\|\frac{\partial e_k(l)}{\partial\, \mathbf{a}}\|^2$ from Eq. (17) to (16), we get $0 < \eta < \frac{2W}{(N+1)W(l)}$.
The derivative of performance index $\triangle \mathbf{P} \le 0$, if learning rate satifies $0 < \eta < \frac{2W}{(N+1)W(l)}$, and the algorithm converges. If $\triangle \mathbf{P} = 0$, from Eq. (13) and (15) we have $\triangle \mathbf{a} = 0$, therefore the NN model is stable and hence, theorem is proved completely. □

3.2 Selection of Weight coefficient Vector

The appropriate values of weight coefficient vector improve the performance of ANN filter design therefore selection of weight vector is the key step of proposed algorithm and should be chosen in the same range as shown in Eq. (18).

$$W(\omega_l) = \begin{cases} \omega_p \ge 1, & \text{in the pass-band} \\ \omega_{p\_edge} \ge 1, & \text{on the pass-band edge} \\ \omega_{s\_edge} \ge 1, & \text{on the stop-band edge} \\ \omega_t = 1, & \text{in the transition-band} \\ \omega_s \ge 1, & \text{in the stop-band} \end{cases} \quad (18)$$

The values of weight coefficient vectors ($W(\omega_l)$) are selected more than unity to minimize the ripples present in pass band and stop band of magnitude response of filter. The transition band does not demand any ripple minimization therefore weight coefficient are kept unity in this region. The ripples are minimized during updation of weight vector ($\mathbf{a}_{k+1}$) of hidden layer of modified Widrow-Hoff Neural Network using Eq. (9).

The following algorithm summarize the proposed design of FIR filter using modified Widro-Hoff Neural Network.
<step 1> Set initial iteration number $k = 0$, stop criterion ε, learning rate η according to convergence theorem and weight coefficient vector $W(\omega_l)$ using equation Eq. (18).
<step 2> Sample the desired magnitude response $\mathbf{A}_d$ uniformly on the frequency sample point at $\omega_l = \frac{l}{L}\,\pi$, where l=1,2,... L.
<step 3> Initialize random weight vector $\mathbf{a}$ for training of proposed neural network.

<step 4> produce amplitude response $\mathbf{A}(\omega_l)$ of neural network. Compare it with desired response $\mathbf{A}_d$ and compute error $\mathbf{e}_k$ and performance index **P** using Eq. (7) and (8).
<step 5> Update the weight vector **a** of the network according to Eq. (9).
<step 6> Check the stop criterion. If $\mathbf{P} \leq \varepsilon$, condition satisfied, we terminate the design process. Otherwise, set $k = k + 1$ and go to step 4 for the next iteration.

3.3 Computational Complexity

In Table 2, a complexity comparison of different schemes and algorithms is summarized, where n and L are the number of neurons (filter coefficients) and number of frequency points on desired response (sampling grid) respectively.
The neural network based models proposed by Bhattacharya [10], Yuo-Dar Jou [14] and Zhao [19] are used feedback neural network, compacted feedback neural network and continuous Hopfield neural network respectively. Table 2 shows that the proposed method requires a less number of computation as compared to other techniques.

Table 3. Performance comparison for lowpass FIR filter design

Design Result ⟶	Maximum peak ripple (dB)	
Algorithm ↓	Passband	Stopband
conventional least-squares [16]	0.7803	21.2296
Bhattacharya and Antoniou [10]	0.6805	20.2199
Yue-Dar Jou and Fu-Kun Chen [14]	0.6282	21.4116
Proposed method	0.6241	21.587

Table 4. Performance comparison for highpass FIR filter design

Design Result ⟶	Maximum peak ripple (dB)	
Algorithm ↓	Passband	Stopband
Parks-McClellan Transformation [6]	0.0985	37.99
Neural network optimization [19]	0.0981	38.13
Proposed method	0.0954	38.49

4. SIMULATIONS AND COMPARISONS

In this section, Matlab programs are used to design three examples of 1-Dimentional FIR digital filters, including the least-squares using the implementation of MWHNN to evaluate the performance of the proposed technique. The weight coefficient vector is set according to Eq. (18) so as to obtain least square approximation in each example.
Example-1. (Low-pass filter): The desired amplitude response A_d is a low-pass FIR filter with unity gain in the pass-band $\{0 \leq \omega \leq \omega_p = 0.30\pi\}$, zero gain in the stop-band $\{\omega_s = 0.30\pi \leq \omega \leq \pi\}$ with a sampling frequency of π. The filter length is chosen to be N=31 (i.e. type I filter) with sampling grid of L=180 [14]. The MWHNN iterates for 1000 times to converge to a low-pass filter with learning rate η=0.1. Fig. 2 illustrates the amplitude response of the low pass FIR filter. Table 3 shows the performance comparison of the low-pass filter design with conventional least-squares [16], Bhattacharya [10], You-Dar Jou [14], and the proposed method.

Example-2. (High-pass filter): For this simulation a high-pass FIR filter with zero gain in the stop-band $\{0 \leq \omega \leq \omega_p = 0.3017\pi\}$, unity gain in the pass-band $\{\omega_s = 0.3994\pi \leq \omega \leq \pi\}$ with a sampling frequency of π is selected as an example from literature [19]. The filter length is chosen to be N=41 (i.e. type I filter) with sampling grid of L=180. The MWHNN iterates for 1000 times to converge to a high-pass filter with learning rate η=0.1. Fig. 3 illustrates the amplitude response of the low pass FIR filter. Table 4 shows the performance comparison of the high-pass filter design with H. Zhao [19], Parks-McClellan transfromation [6] and the proposed method.

Example-3. (Multiband filter): The desired amplitude response A_d is a 65−tap multiband filter with unity gain in the pass-band region $\omega_p = \{0 \leq \omega \leq 0.1\pi\} \cup \{0.5\pi \leq \omega \leq 0.66\pi\}$, zero gain in the stop-band region $\omega_s = \{0.16\pi \leq \omega \leq 0.44\pi\} \cup \{0.7\pi \leq \omega \leq \pi\}$. The filter length (N) and sampling grid (L) are chosen as 65 and 501 respectively [13]. The MWHNN iterates for 2000 times to converge to a multiband filter with learning rate η =0.1. Fig. 4 illustrates the amplitude response of the multiband filter. Table 5 shows the performance comparison of the multiband filter design results with X. P. lai [3], X. Wang [13] and the proposed method.
From all three examples, It is evident that the proposed network shows a better convergance to a optimal solution and it's performance is superior to other methods.

5. CONCLUSIONS

The proposed MWHNN based approach is an alternatively computationally effective, weighted least square technique for designing FIR digital filters. Few examples are simulated and compared on the basis of maximum peak ripples in passband and stopband of FIR filter. Furthermore, the required number of neurons in MWHNN is independent of the sampling grid in the frequency domain and approximately proportional to the filter length. Therefore, higher order filter can further be designed efficiently by using the proposed neural network approach.

6. REFERENCES

[1] S. Sunder, "An efficient weighted least-squares design of linear phase nonrecursive filters," *IEEE Trans. Circuits Syst. II*, 4th ed. vol. 42, no. 5, pp. 359–361, 1995.

[2] M. Okuda, M. Ikehara and S. Takahashi, Fast and stable least-squares approach for the design of linear phase FIR filters, IEEE Signal Process. Lett., vol. 46, no. 6, pp. 1485-1493, (1998).

[3] X. P. Lai, "Constrained Chebyshev Design of FIR filters," *IEEE Trans. Circuits Syst. II*, vol. 51, no. 3, pp. 143–146, 2004.

[4] S. C. Pei, and H. S. Lin, Minimum-Phase FIR filter design using real cepstrum, IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 10, pp. 1113-1117, (2006).

[5] J. G. Proakis and D. G. Manolakis, *Digital Signal Processing*, 4th ed. Pearson Prentice Hall International, Inc, 1996.

[6] L. R. Rabiner, J. H. McClellan, and T. W. Parks, "FIR digital filter design techniques using weighted Chebyshev approximation," *Proc. IEEE*, vol. 63, pp. 595–610, 1975.

[7] P. Zahradnik and M. Vlcek, Analytical design method for optimal equiripple comb FIR filters, IEEE Trans. Circuits Syst. II, Exp. Briefs, vol. 53, no. 2, pp. 112-115, (2005).

[8] H. D. Tuan, T. T. Son, H. Tuy and T. Nguyen, "New linear-programming-based filter design," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 52, no. 5, pp. 276–281, 2005.

[9] A. T. Chottera and G. A. Jullien, A linear programming approach to recursive digital filter design with linear phase, IEEE Trans. Circuits Syst., vol. 29, pp. 139-149, (1982).

[10] D. Bhattacharya, and A. Antoniou, "Real time design of FIR filters by feedback neural networks ," *IEEE Signal Process. lett.*, vol. 3, no. 5, pp. 158–161, 1996.

Table 5. Performance comparison for multiband FIR filter design

Design Result →	Maximum peak ripple (dB)			
Algorithm ↓	Passband-1	Passband-2	Stopband-1	Stopband-2
X. P. Lai [3]	more than 0.5	more than 0.5	35	32
X. Wang [13]	0.2222	0.4118	45.75	37.951
Proposed method	0.1563	0.2318	49.68	38.39

[11] D. Bhattacharya, and A. Antoniou, Design of equiripple FIR filter using feedback neural network, IEEE Trans. Circuits Syst-II, vol. 45, no. 4, pp. 527-531, (1998).

[12] Yue-Dar Jou, Least-squares design of digital differentiators using neural networks with closed-form derivations, IEEE Signal Process. Lett., vol. 12, no. 11, pp. 760-763, (2005).

[13] X. Wang, Xianzhi Meng and Yigang He, "A Novel Neural Networks-Based Approach for Designing FIR Filters," *Proc. IEEE.*, vol. 1, pp. 4029–4032, 2006.

[14] Yue-Dar Jou and Fu-Kun Chen, " Least-Squares Design of FIR Filters Based on a Compacted Feedback Neural Network," *IEEE trans. Circuits and Systems- II*, vol. 54, no. 5, pp. 427–431, 2007.

[15] S. A. Hosseini and F. Farokhi, An Optimum Design of Linear Phase FIR Filters by the Generalized Brain-State-in-a-Box Neural Network Model, Proc. IEEE., vol. 1, pp. 253-258, (2009).

[16] W. P. Zhu, M. O. Ahmad and M. N. Swamy, " Weighted least-square design of design of FIR filters using fast iterative matrix inversion algorithm," *IEEE Trans. Circuits Syst-I*, vol. 49, no. 11, pp. 143–146, 2002.

[17] Y. C. Lim, J. H. Lee, C. K. Chen and R. H. Yang, A weighted least square algorithm for quasi-equiripple FIR and IIR digital filter design, IEEE Trans. Signal Process., vol. 40, no. 3, pp. 551-558, (1992).

[18] M. T. Hagan, H. B. Demuth and M. Beale, *Neural Network Design*, 4th ed. Cengage Learning India Pvt. ltd, 2011.

[19] H. Zhao and J. Yu, "A Novel Neural Network-based Approach for Designing Digital Filters," *IEEE International Symposium on Circuits and Systems*, pp. 2272–2275, 1997.