Transactions Briefs

Design of Nonrecursive Filters Satisfying Arbitrary Magnitude and Phase Specifications Using a Least-Squares Approach

Sunder S. Kidambi and Ravi P. Ramachandran

Abstract—**A method is described which can be used to design nonrecursive filters satisfying prescribed magnitude and phase specifications. The method is based on formulating the absolute mean-square error between the frequency response of the practical filter and the desired response as a quadratic function. The coefficients of the filters are obtained by solving a set of linear equations. It is shown that our method, in general, has an order of magnitude lower computational complexity than the eigenfilter method. For the design of allpass filters, in particular, the computational complexity is three orders of magnitude lower than the eigenfilter method. In addition, our method yields a lower mean-square error.**

I. Introduction

The design of linear-phase finite impulse response (FIR) filters is generally carried out using the McClellan-Parks (MP) algorithm [1] and least-squares methods [2]-[9]. The MP approach is based on the Remez exchange algorithm and yields filters that are optimal in the minimax sense. In [3]-[6], FIR filters are designed using a least-squares method, called the eigenfilter method, in which the mean-square error is formulated in a quadratic form. The filter coefficients are obtained by computing the eigenvector corresponding to the smallest eigenvalue of a real, symmetric and positive-definite matrix. In [2], [7]-[9], the least-squares method is based on formulating the mean-square error as a quadratic function. Here, the filter coefficients are obtained simply by solving a system of linear equations.

A linear-phase FIR filter has symmetry/antisymmetry constraints imposed on its impulse response. Consequently, for a given filter length, the group delay assumes a constant value for all frequencies. Furthermore, a large length FIR filter is needed to satisfy a narrow transition-band specification thereby leading to a rather high group delay. On the other hand, a minimum phase FIR filter can be designed to achieve a lower group delay but does not provide a constant group delay for all frequencies including those in the passband. Therefore, to achieve an arbitrary constant group delay filter, neither a linear phase nor a minimum phase filter can be used. Other nonlinear phase characteristics, such as mandated by allpass phase equalizers, can only be achieved by filters satisfying arbitrary magnitude and phase specifications.

In [10]-[14], several methods have been proposed to simultaneously approximate magnitude and phase specifications. In [10], a complex Chebyshev approximation is first converted into a real approximation problem. The solution to an overdetermined set of linear equations obtained by linear programming techniques [15]-[17] yields the filter coefficients. A linear programming approach is also used in [11] to design FIR allpass phase equalizers. This method requires a large memory space and considerable computing time. In [12], a complex Chebyshev approximation is converted into two real approximation problems. The MP algorithm is applied to the two real problems individually to obtain the filter coefficients. In [13] and [14], the eigenfilter method for the design of linear phase filters is extended to the design of FIR filters whose frequency response approximates a complex-valued function in a least-squares sense. It is shown that the eigenfilter method is computationally efficient and yields filters that are comparable in performance with those obtained in [10] and [11].

It has been shown that the least-squares method in [8] and [9] for the design of linear phase filters is computationally more efficient and leads to a lower mean-square error than the eigenfilter method. In this paper, we extend this method to the design of FIR filters satisfying prescribed magnitude and phase specifications. Our method has, in general, an order of magnitude lower computational complexity than the eigenfilter method [13], [14]. For the case of allpass filters, in particular, our method only requires the evaluation of a vector. Consequently, our method has approximately three orders of magnitude lower computational complexity than the eigenfilter method. Through examples, it is also shown that our method yields a lower mean-square error.

II. Design Procedure for FIR Filters

The frequency response of an FIR digital filter with N taps specified by a real-valued impulse response $h(n)$ is given by

$$\begin{aligned} H(e^{j\omega}) &= \sum_{n=0}^{N-1} h(n)e^{-jn\omega} \\ &= \sum_{n=0}^{N-1} h(n)\cos(n\omega) - j\sum_{n=0}^{N-1} h(n)\sin(n\omega) \\ &= \mathbf{h}^T\mathbf{c}(\omega) - j\mathbf{h}^T\mathbf{s}(\omega) \end{aligned} \tag{1}$$

where

$$\begin{aligned} \mathbf{h} &= [h(0)h(1)h(2)\cdots h(N-1)]^T \\ \mathbf{c}(\omega) &= [1\cos(\omega)\cos(2\omega)\cdots\cos((N-1)\omega)]^T \\ \mathbf{s}(\omega) &= [0\sin(\omega)\sin(2\omega)\cdots\sin((N-1)\omega)]^T. \end{aligned}$$

The phase response of the filter is given by

$$\phi(\omega) = -\tan^{-1}\left(\frac{\mathbf{h}^T\mathbf{s}(\omega)}{\mathbf{h}^T\mathbf{c}(\omega)}\right)$$

and the group delay is given by

$$\tau(\omega) = -\frac{d}{d\omega}\phi(\omega).$$

The desired frequency response $D(\omega)$ having an amplitude response $M(\omega)$ and a phase response $\rho(\omega)$ is given by

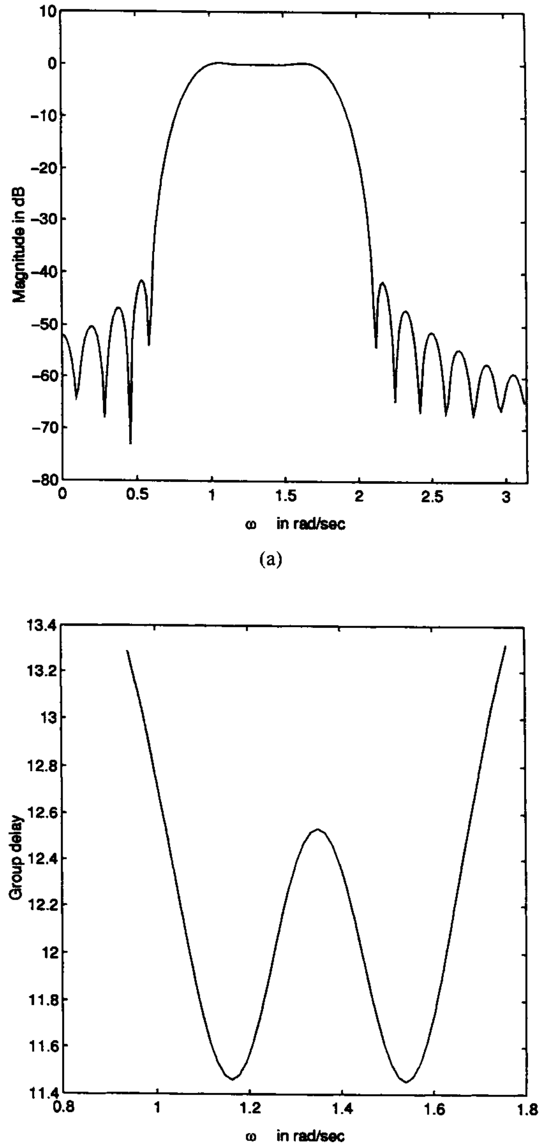
$$D(\omega) = \begin{cases} M(\omega)e^{j\rho(\omega)} = M(\omega)\cos(\rho(\omega)) + jM(\omega)\sin(\rho(\omega)) & \omega \in P \\ 0 & \omega \in S \end{cases} \tag{2}$$

Manuscript received February 4, 1993; revised December 30, 1994. This paper was recommended by Associate Editor Y. C. Lim.

S. S. Kidambi is with Analog Devices, Wilmington, MA 01887 USA.

R. P. Ramachandran is with the Department of Electrical Engineering, Rutgers University, Piscataway, NJ 08855 USA.

IEEE Log Number 9415298.



(b)

Fig. 1. Frequency response of the bandpass filter. (a) Magnitude response. (b) Group delay response in the passband.

where P is the passband, S is the stopband and

$$\tau_\rho(\omega) = -\frac{d\rho(\omega)}{d\omega}$$

is the desired group delay response. Comparing (1) and (2) it can be seen that a nonrecursive filter satisfying prescribed magnitude and phase specifications can be designed provided the coefficients $h(n)$ are determined appropriately.

The mean-square error between $D(\omega)$ and $H(e^{j\omega})$ can be expressed as

$$\begin{aligned} E_{\mathrm{mse}} &= \alpha \int_P |D(\omega) - H(e^{j\omega})|^2 d\omega + \beta \int_S |H(e^{j\omega})|^2 d\omega \\ &= \alpha \int_P \left\{ \Big(M(\omega)\cos(\rho(\omega)) - \mathbf{h}^T \mathbf{c}(\omega) \Big)^2 \right. \\ &\quad \left. + \Big(M(\omega)\sin(\rho(\omega)) + \mathbf{h}^T \mathbf{s}(\omega) \Big)^2 \right\} d\omega \\ &\quad + \beta \int_S \mathbf{h}^T (\mathbf{Q}_c + \mathbf{Q}_s) \mathbf{h} \, d\omega \end{aligned} \tag{3}$$

where $\mathbf{Q}_c = \mathbf{c}(\omega)\mathbf{c}^{\mathbf{T}}(\omega)$ and $\mathbf{Q}_s = \mathbf{s}(\omega)\mathbf{s}^{\mathbf{T}}(\omega)$.

TABLE I
BANDPASS FILTER COEFFICIENTS

n	$h(n)$
0	5.130387523660900e-03
1	9.888451904419063e-03
2	-1.578075087670760e-02
3	-3.395956668768713e-02
4	7.426244047221808e-03
5	2.965599220047781e-02
6	3.559780413337920e-04
7	4.261220691298246e-02
8	7.210925899355972e-02
9	-1.147468270152104e-01
10	-2.332935178712081e-01
11	6.795511270065929e-02
12	3.382773056358781e-01
13	7.225351626564923e-02
14	-2.620347047088969e-01
15	-1.363457704465302e-01
16	9.006787721835299e-02
17	5.424676022631854e-02
18	1.326573210346884e-03
19	4.979996695722889e-02
20	1.296947769629594e-02
21	-5.938791705183812e-02
22	-2.694045681525601e-02
23	1.359597120131944e-02
24	-2.544992623975987e-03
25	5.119408357996645e-03
26	2.560589327968212e-02
27	3.943365630876753e-03
28	-1.658958743785633e-02
29	-5.637372306205530e-03
30	2.395612858651372e-03

By minimizing the error function E_{mse} with respect to the filter coefficients, the required filter can be designed. In minimizing E_{mse}, we set $\frac{\partial E_{\mathrm{mse}}}{\partial h(n)} = 0$, for $0 \le n \le N-1$, to obtain a system of linear equations $(\alpha\mathbf{Q} + \beta\mathbf{R})\mathbf{h} = \alpha\mathbf{d}$, where

$$\mathbf{Q} = \int_P (\mathbf{Q}_c + \mathbf{Q}_s) d\omega \tag{4}$$

$$\mathbf{R} = \int_S (\mathbf{Q}_c + \mathbf{Q}_s) d\omega \tag{5}$$

$$\mathbf{d} = \int_P M(\omega)(\cos(\rho(\omega))\mathbf{c}(\omega) - \sin(\rho(\omega))\mathbf{s}(\omega)) d\omega. \tag{6}$$

It can be noted that $\mathbf{Q}$ and $\mathbf{R}$ are symmetric Toeplitz matrices. Consequently, the system of linear equations can be solved by the computationally efficient and robust Levinson's algorithm which entails only $\mathcal{O}(N^2)$ complexity [19].

III. DESIGN EXAMPLES

Example 1

We design of a bandpass filter having the following specifications:

$$D(\omega) = \begin{cases} \cos(\gamma\omega) - j\sin(\gamma\omega) & \omega \in P = [0.3\pi, 0.56\pi] \\ 0 & \omega \in S = [0, 0.2\pi] \cup [0.66\pi, \pi] \end{cases}.$$

The desired passband group delay is $\gamma = 12$. The magnitude and group delay response of a 31-tap bandpass filter designed using $\alpha = 1$ and $\beta = 10$ are shown in Fig. 1(a) and (b), respectively. The filter coefficients are tabulated in Table I.

IV. Design of FIR Allpass Phase Equalizers

For an allpass phase equalizer, the desired characteristic can be expressed as

$$D(\omega) = e^{j\rho(\omega)} = \cos(\rho(\omega)) + j\sin(\rho(\omega))\ \omega \in P = [0, \pi]. \quad (7)$$

Following the development in the previous section, E_{mse} is minimized and a system of linear equations $\mathbf{Qh} = \mathbf{d}$ is obtained. For the design of allpass filters with specified phase responses, it is not necessary to solve a system of linear equations to obtain the filter coefficients. This is because $\mathbf{Q}$ can be written as

$$\begin{aligned}\mathbf{Q} &= \int_0^{\pi} (\mathbf{Q}_c + \mathbf{Q}_s) d\omega \\ &= \pi \mathbf{I}\end{aligned} \quad (8)$$

where $\mathbf{I}$ is an $N \times N$ identity matrix. Thus the filter coefficients can be obtained simply as

$$\mathbf{h} = \frac{\mathbf{d}}{\pi} \quad (9)$$

and consequently, the computational effort is significantly reduced. In addition, if $\rho(\omega)$ is symmetric or antisymmetric with respect to $\pi/2$, the computational complexity is further reduced since only half the number of coefficients need to be determined. Below we shall consider the design of two allpass phase equalizers advanced in [11] using our method.

A. Symmetric Phase Characteristics

The desired phase characteristic is given by

$$\rho(\omega) = -\frac{N-1}{2}\omega + \hat{\rho}(\omega) \quad (10)$$

where the first term on the right-hand side is the linear phase term and $\hat{\rho}(\omega)$ is a function of ω symmetric about $\pi/2$. It must be mentioned that the number of filter taps N is odd. It follows that [11]

$$h\left(\frac{N-1}{2} - n\right) = h\left(\frac{N-1}{2} + n\right) \text{when } n \text{ is even}$$
$$h\left(\frac{N-1}{2} - n\right) = -h\left(\frac{N-1}{2} + n\right) \text{when } n \text{ is odd.}$$

From (1), the allpass phase equalizer can be characterized by

$$e^{j\frac{N-1}{2}\omega} H(e^{j\omega}) = \sum_{n=0}^{U} a(n)\cos(2n\omega) + j\sum_{n=1}^{V} b(n)\sin((2n-1)\omega) \quad (11)$$

where $U = \lfloor \frac{N-1}{4} \rfloor$, $V = \lfloor \frac{N+1}{4} \rfloor$,

$$a(n) = \begin{cases} h\left(\frac{N-1}{2}\right) & n = 0 \\ 2h\left(\frac{N-1}{2} - 2n\right) & n = 1, 2, \ldots, U \end{cases}$$

and

$$b(n) = 2h\left(\frac{N-1}{2} - 2n + 1\right)\ n = 1, 2, \ldots, V.$$

Consequently (11) can be written as

$$e^{j\frac{N-1}{2}\omega} H(e^{j\omega}) = \mathbf{a}^T \hat{\mathbf{c}}(\omega) + j\mathbf{b}^T \hat{\mathbf{s}}(\omega) \quad (12)$$

where

$$\begin{aligned}\mathbf{a} &= [a(0)a(1)\ldots a(U)]^T \\ \mathbf{b} &= [b(1)b(2)\ldots b(V)]^T \\ \hat{\mathbf{c}}(\omega) &= [1\cos(2\omega)\ldots\cos(2U\omega)]^T \\ \hat{\mathbf{s}}(\omega) &= [\sin(\omega)\sin(3\omega)\ldots\sin((2V-1)\omega)]^T.\end{aligned}$$

From (11), the mean-square error associated with the real part can be written as

$$E_R = \int_0^{\pi} \left(\cos(\hat{\rho}(\omega)) - \mathbf{a}^T \hat{\mathbf{c}}(\omega)\right)^2 d\omega. \quad (13)$$

By setting $\frac{\partial E_R}{\partial a(n)} = 0$, for $0 \le n \le U$, we get $\hat{\mathbf{Q}}_a \mathbf{a} = \hat{\mathbf{d}}_a$, where the elements of $\hat{\mathbf{Q}}_a$ are given by

$$\hat{Q}_a(n, m) = \begin{cases} \pi & n = m = 0 \\ \frac{\pi}{2} & n = m \ne 0 \\ 0 & n \ne m \end{cases} \quad (14)$$

for $0 \le n, m \le U$. The elements of $\hat{\mathbf{d}}_a$ are given by

$$\hat{d}_a(n) = \int_0^{\pi} \cos(\hat{\rho}(\omega))\cos(2n\omega) d\omega \quad n = 0, 1, \ldots, U. \quad (15)$$

From (14) and (15), the elements of $\mathbf{a}$ are given by

$$a(n) = \begin{cases} \frac{\hat{\mathbf{d}}_a(n)}{\pi} & n = 0 \\ \frac{2\hat{\mathbf{d}}_a(n)}{\pi} & n = 1, 2, \ldots, U \end{cases}. \quad (16)$$

Similarly, the mean-square error associated with the imaginary part can be written as

$$E_I = \int_0^{\pi} \left(\sin(\hat{\rho}(\omega)) - \mathbf{b}^T \hat{\mathbf{s}}(\omega)\right)^2 d\omega. \quad (17)$$

Again, by setting $\frac{\partial E_I}{\partial b(n)} = 0$, for $1 \le n \le V$, we get $\hat{\mathbf{Q}}_b \mathbf{b} = \hat{\mathbf{d}}_b$, where the elements of $\hat{\mathbf{Q}}_b$ are given by

$$\hat{Q}_b(n, m) = \begin{cases} \frac{\pi}{2} & n = m \\ 0 & n \ne m \end{cases} \quad (18)$$

for $1 \le n, m \le V$. The elements of $\hat{\mathbf{d}}_b$ are given by

$$\hat{d}_b(n) = \int_0^{\pi} \sin(\hat{\rho}(\omega))\sin((2n-1)\omega)\, d\omega\ n = 1, 2, \ldots, V. \quad (19)$$

From (18) and (19), the elements of $\mathbf{b}$ are given by

$$b(n) = \frac{2\hat{\mathbf{d}}_b(n)}{\pi} \quad n = 1, 2, \ldots, V. \quad (20)$$

Example 2: *Chirp Allpass Phase Equalizer*

The desired phase characteristic of the chirp allpass phase equalizer is given by

$$\rho(\omega) = -\frac{N-1}{2}\omega - \frac{8}{\pi}\left(\omega - \frac{\pi}{2}\right)^2. \quad (21)$$

Consequently, the group delay is given by

$$\tau_\rho(\omega) = \frac{N-1}{2} + \frac{16}{\pi}\left(\omega - \frac{\pi}{2}\right). \quad (22)$$

The amplitude and group delay response of a 61-tap phase equalizer are shown in Fig. 2(a) and (b), respectively. Fig. 2(c) shows the variation of the group-delay error (difference between the ideal group delay and the obtained group delay) with frequency.

B. Antisymmetric Phase Characteristics

When $\hat{\rho}(\omega)$ is antisymmetric with respect to $\pi/2$, the coefficients satisfy the following condition [11]:

$$h\left(\frac{N-1}{2} - n\right) = h\left(\frac{N-1}{2} + n\right) = 0 \text{ when } n \text{ is odd} \quad (23)$$

where, again, N is odd. In designing such allpass filters, $\mathbf{Q}$ and $\mathbf{h}$ are given as in (8) and (9) except that the rows and columns corresponding to the zero-valued coefficients are deleted.

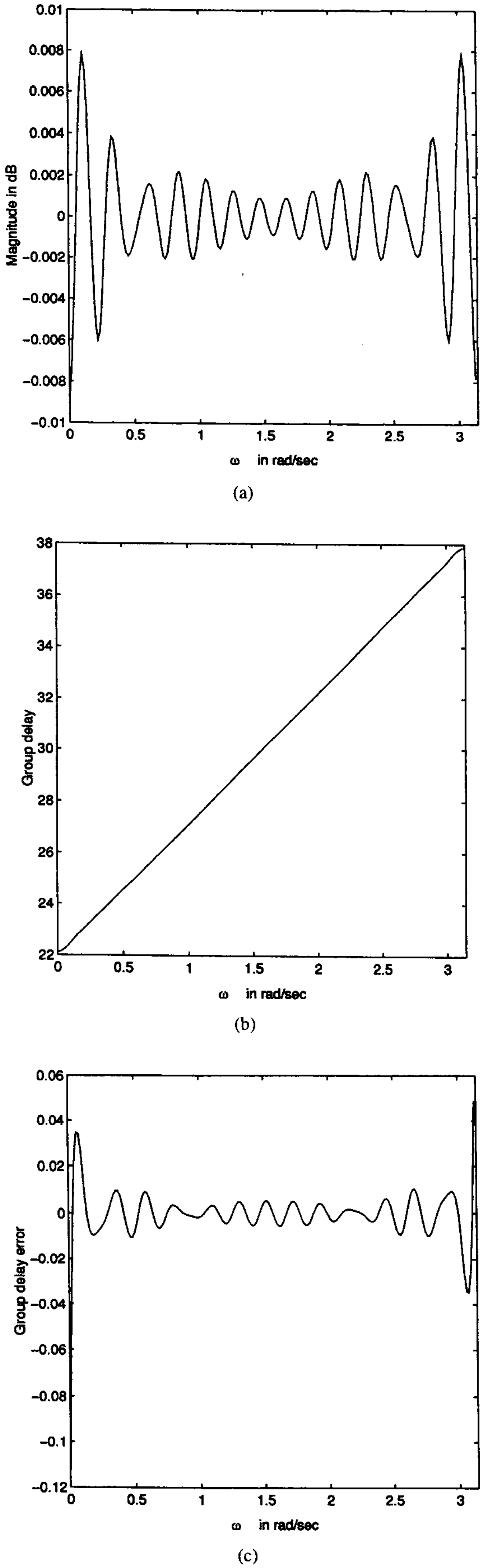


Fig. 2. Frequency response of the Chirp allpass filter. (a) Magnitude response. (b) Group delay response. (c) Group delay error.

Example 3: *Sine-Delay Allpass Phase Equalizer*

The desired characteristic of the sine-delay allpass equalizer is given by

$$\rho(\omega) = -\frac{N-1}{2}\omega + 2\pi(1-\cos\omega). \tag{24}$$

Consequently, the group delay is given by

$$\tau_\rho(\omega) = \frac{N-1}{2} - 2\pi\sin\omega. \tag{25}$$

The amplitude and group delay response of a 61-tap phase equalizer are shown in Fig. 3(a) and (b), respectively. Fig. 3(c) shows the variation of the group-delay error with frequency.

V. Performance Results

In this section, we compare our design method with the eigenfilter approach in terms of the mean-square error $E_{\rm mse}$, the peak error given by

$$E_M = \max_{\omega\in P\cup S} |D(\omega) - H(e^{j\omega})| \tag{26}$$

and the peak passband group-delay error given by

$$E_\tau = \max_{\omega\in P} |\tau_\rho(\omega) - \tau(\omega)| \tag{27}$$

for all the examples in [13]. A comparison of the two methods with respect to $E_{\rm mse}$, E_M and E_τ is shown in Table II. The reference frequencies for all the designs using the eigenfilter method have been chosen as in [13].

A. Error Measure

Our method formulates a better error measure than the eigenfilter method in that we explicitly minimize the mean-square error between the ideal response and the frequency response of the obtained filter. In contrast, the eigenfilter method does not take the ideal response into account. Rather, it uses a scaled version of the desired frequency response where the scaling factor is $H(e^{j\omega_0})/D(\omega_0)$ and ω_0 is an arbitrary reference frequency. As a consequence, our method yields a lower mean-square error. However, the differences in the value of $E_{\rm mse}$ obtained for both methods are small. Also, the differences in the values of E_M and E_τ obtained for both methods are small.

B. Computational Complexity

For our method, the filter parameters are obtained by a system of linear equations involving a symmetric Toeplitz matrix $\mathbf{G} = (\alpha\mathbf{Q} + \beta\mathbf{R})$. Such a system of linear equations can be solved by the computationally efficient and robust Levinson's algorithm [19] which involves only $\mathcal{O}(N^2)$ complexity.

In the eigenfilter approach [13], the mean-square error is formulated as

$$E_{\rm mse} = \mathbf{h}^T\mathbf{P}\mathbf{h}$$

where $\mathbf{P}$ is a real, symmetric and positive-definite matrix. The coefficients of the filters are obtained as the eigenvector that corresponds to the smallest eigenvalue of $\mathbf{P}$. In order to compute the smallest eigenvalue and its corresponding eigenvector, generally, an iterative inverse power method is used [3]. At the $(k+1)$th iteration, a vector $\mathbf{x}_{k+1}$ is computed from the previous iterate $\mathbf{x}_k$ as

$$\mathbf{y}_{k+1} = \mathbf{P}^{-1}\mathbf{x}_k \tag{28}$$

$$\mathbf{x}_{k+1} = \mathbf{y}_{k+1}/ \parallel \mathbf{y}_{k+1} \parallel \tag{29}$$

where $\parallel \mathbf{y}_{k+1} \parallel$ denotes the L_2 norm of $\mathbf{y}_{k+1}$. If $\parallel \mathbf{x}_{k+1} - \mathbf{x}_k \parallel \leq \epsilon$ (typically ϵ is about 10^{-6}), then $\mathbf{x}_{k+1}$ is a good approximation of the

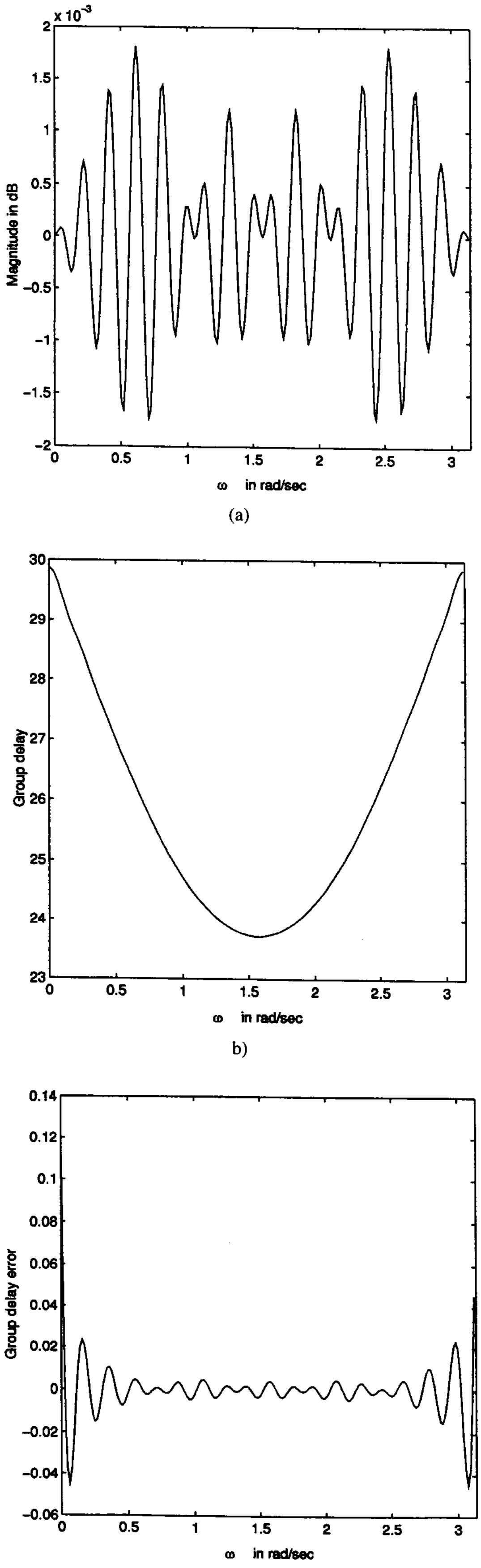


Fig. 3. Frequency response of the Sine-delay allpass filter. (a) Magnitude response. (b) Group delay response. (c) Group delay error.

TABLE II
COMPARISON OF THE TWO METHODS WITH RESPECT TO THE MEAN-SQUARE, PEAK MAGNITUDE AND PEAK GROUP-DELAY ERRORS FOR ALL THE EXAMPLES IN [13]

Examples	E_{mse}		E_M		E_τ	
	Our method	Eigenfilter method	Our method	Eigenfilter method	Our method	Eigenfilter method
Lowpass	6.414e-05	6.525e-05	6.706e-02	6.503e-02	1.007e-00	1.029e-00
Bandpass	4.209e-04	4.279e-04	1.148e-01	1.197e-01	1.319e-00	1.338e-00
Differentiator	2.439e-05	2.443e-05	4.325e-02	4.339e-02	4.587e-02	4.609e-02
Hilbert Transformer	1.474e-06	1.479e-06	1.189e-02	1.195e-02	5.102e-01	5.132e-01
Chirp allpass	1.803e-07	1.806e-07	1.769e-03	1.773e-03	1.172e-01	1.174e-01
Sine-delay allpass	2.934e-07	2.935e-07	1.583e-03	1.584e-03	1.290e-01	1.290e-01

eigenvector corresponding to the smallest eigenvalue of $\mathbf{P}$. We can rewrite (28) as $\mathbf{x}_k = \mathbf{P}\mathbf{y}_{k+1}$. By solving a system of linear equations, we can obtain $\mathbf{y}_{k+1}$ and subsequently $\mathbf{x}_{k+1}$. Since $\mathbf{P}$ is real, symmetric and positive-definite but not Toeplitz, the solution of the system of linear equations involves $\mathcal{O}(N^3)$ complexity. Moreover, the eigenfilter method requires solving a system of linear equations several times before obtaining the eigenvector corresponding to the smallest eigenvalue. As is evident, our method has an order of magnitude lower complexity than the eigenfilter method. For the design of allpass filters, our method involves only the evaluation of $\mathbf{d}$. Consequently, our method has approximately three orders of magnitude lower computational complexity than the eigenfilter method.

VI. CONCLUSION

In this brief, a method to design FIR filters satisfying arbitrary magnitude and phase specifications has been presented. In this method, the absolute mean-square error between ideal and actual frequency responses is explicitly minimized. The filter coefficients are found in a noniterative and computationally efficient manner. The mean-square error achieved by our method is lower than that achieved by the eigenfilter approach.

REFERENCES

[1] J. H. McClellan, T. W. Parks, and L. R. Rabiner, "A computer program for designing optimum FIR linear phase digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 506-526, Dec. 1973.
[2] D. W. Tufts and J. T. Francis, "Designing digital lowpass filters: Comparison of some methods and criteria," *IEEE Trans. Audio Electroacoust.*, vol. AU-18, pp. 487–494, Dec. 1970.
[3] P. P. Vaidyanathan and T. Q. Nguyen, "Eigenfilters: A new approach to least-squares FIR filter design and applications including Nyquist filters," *IEEE Trans. Circuits Syst.*, vol. CAS-34, pp. 11–23, Jan. 1987.
[4] T. Q. Nguyen, "The eigenfilter for the design of linear-phase filters with arbitrary magnitude response," *IEEE Int. Conf. Acoust., Speech, Signal Process.*, Toronto, ON, Canada, pp. 1981–1984, May 1991.
[5] S. C. Pei and J. J. Shyu, "Design of FIR Hilbert transformers and differentiators by eigenfilter," *IEEE Trans. Circuits Syst.*, vol. 35, pp. 1457-1461, Nov. 1988.
[6] ——, "Eigenfilter design of higher-order digital differentiators," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 37, pp. 505–511, Apr. 1989.
[7] S. Sunder, W.-S. Lu, A. Antoniou, and Y. Su, "Design of digital differentiators satisfying prescribed specifications using optimization techniques," *Proc. IEE*, vol. 138, pt. G, pp. 315–320, June 1991.
[8] S. Sunder and R. P. Ramachandran, "Least-squares design of higher-order nonrecursive differentiators," *IEEE Trans. Signal Process.*, vol. 42, pp. 956–961, Apr. 1994.
[9] ——, "A unified and efficient least-squares design of linear phase nonrecursive filters," *Signal Processing*, vol. 36, pp. 41–53, Mar. 1994.

[10] X. Chen and T. W. Parks, "Design of FIR filters in the complex domain," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, pp. 144–153, Feb. 1987.
[11] K. Steiglitz, "Design of FIR digital phase networks," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, pp. 171-176, Apr. 1981.
[12] S.-C. Pei and J.-J. Shyu, "Design of real FIR filters with arbitrary complex frequency responses by two real Chebyshev approximations," *Signal Processing*, vol. 26, pp. 119–129, Jan. 1992.
[13] ——, "Eigenapproach for designing FIR filters and allpass phase equalizers with prescribed magnitude and phase response," *IEEE Trans. Circuits Systems-II*, vol. 39, pp. 137–146, Mar. 1992.
[14] T. Q. Nguyen, "The design of arbitrary FIR digital filters using the eigenfilter method," *IEEE Trans. Signal Process.*, vol. 41, pp. 1128–1139, Mar. 1993.
[15] K. Glashoff and K. Roleff, "A new method for Chebyshev approximation of complex-valued function," *Math Comp.*, vol. 36, pp. 233–239, 1981.
[16] J. Barrodale and C. Phillips, "Solution of an overdetermined system of linear equations in the Chebyshev norm, algorithm 495," *ACM Trans. Math Software*, vol. 1, pp. 264–270, 1975.
[17] R. L. Streit and A. H. Nuttal, "A general Chebyshev complex function approximation procedure and an application to beam forming," *J. Acoust. Soc. Amer.*, vol. 72, pp. 181–190, 1982.
[18] L. R. Rabiner and B. Gold, *Theory and Applications of Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.
[19] S. L. Marple, Jr., *Digital Spectral Analysis with Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

Transient Suppression at the Boundary for 2-D Digital Systems

W. W. Edmonson and W. Alexander

***Abstract*— We present an approach to the suppression of transients due to unspecified initial conditions or truncation of data for 2-D digital systems. The approach is based upon the use of Roesser's 2-D state space model.**

I. Introduction

Many practical digital systems require the use of appropriate boundary values or initial conditions. The classical approach of assigning a value of zero to initial conditions and boundary values often leads to undesirable transients in the output. *Ad hoc* approaches to this problem usually involve extending the data by replication or by the use of window functions [1]. The boundary value transient problem is more prominent for systems which have an impulse response of long duration, e.g., edge enhancement and high pass filters.

In this brief, we present an approach to this problem based upon associating transients with changes in the state of the system. We develop a suppression algorithm based upon not permitting the state of the system to change at the boundaries. Although this approach makes the system shift variant [1], it gives very desirable results for most applications. Appropriate boundary values can minimize the transients due to initial conditions. This problem has been addressed previously in [2] for image restoration and in [3] and [4] for 2-D recursive filters.

Manuscript received October 25, 1993; revised August 12, 1994. This paper was recommended by Associate Editor W.-S. Lu.

W. W. Edmonson is with the Department of Electrical Engineering, University of Florida, Gainesville, FL 32604 USA.
W. Alexander is with the Department of Electrical Engineering, North Carolina State University, Raleigh, NC 20059 USA.

IEEE Log Number 9415297.

We have chosen to restrict the development of the transient suppression algorithm to the recursively computable, discrete, linear, shift-invariant system (DLSI) with first quadrant support as represented by the 2-D state space (SS) model. A recursively computable system with support other than first quadrant support can be mapped into a corresponding system with first quadrant support [1]. Thus, our results can apply to any recursively computable system.

II. The 2-D State Space Representation

We use a modified Roesser's state space model [5], [6] which represents a recursively computable 2-D DLSI system with quarter plane support and is given by

$$\begin{aligned}\begin{bmatrix} Q_H(n_1,n_2) \\ Q_V(n_1,n_2)\end{bmatrix} &= \mathbf{A}\begin{bmatrix} Q_H(n_1-1,n_2) \\ Q_V(n_1,n_2-1)\end{bmatrix} + \mathbf{B}f(n_1,n_2)\\ g(n_1,n_2) &= \mathbf{C}\begin{bmatrix} Q_H(n_1-1,n_2) \\ Q_V(n_1,n_2-1)\end{bmatrix} + \mathbf{D}f(n_1,n_2)\end{aligned} \tag{1}$$

where $f(n_1, n_2)$ and $g(n_1, n_2)$ are the input and output scalars, respectively. $A, B, C,$ and D represent the SS parameters of appropriate dimensions.

We simplify the analysis of the boundary state equations by representing the SS parameters in block matrix form, shown below as

$$\mathbf{A} = \begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22}\end{bmatrix};\ \mathbf{B} = \begin{bmatrix}\mathbf{B}_1 \\ \mathbf{B}_2\end{bmatrix};\ \mathbf{C} = [\mathbf{C}_1\, \mathbf{C}_2]. \tag{2}$$

We also define $\hat{\mathbf{A}}_i$ as the matrix made up of the ith matrix subcolumn of $\mathbf{A}$ with all other submatrices equal to null matrices of the appropriate dimensions. Thus, for the 2-D system

$$\hat{\mathbf{A}}_1 = \begin{bmatrix}\mathbf{A}_{11} & \Theta \\ \mathbf{A}_{21} & \Theta\end{bmatrix}; \hat{\mathbf{A}}_2 = \begin{bmatrix}\Theta & \mathbf{A}_{12} \\ \Theta & \mathbf{A}_{22}\end{bmatrix} \tag{3}$$

where Θ is a null matrix of appropriate dimensions. In a similar manner, we define $\hat{\mathbf{C}}_i$ as

$$\hat{\mathbf{C}}_1 = [\mathbf{C}_1\, \Theta]; \hat{\mathbf{C}}_2 = [\Theta\, \mathbf{C}_2]. \tag{4}$$

We can then express the 2-D state space model in the form

$$\begin{aligned} Q(n_1,n_2) &= \hat{\mathbf{A}}_1 Q(n_1-1,n_2)\\ &\quad + \hat{\mathbf{A}}_2 Q(n_1,n_2-1) + \mathbf{B}f(n_1,n_2)\\ g(n_1,n_2) &= \hat{\mathbf{C}}_1 Q(n_1-1,n_2)\\ &\quad + \hat{\mathbf{C}}_2 Q(n_1,n_2-1) + \mathbf{D}f(n_1,n_2)\end{aligned} \tag{5}$$

where

$$Q(n_1,n_2) = \begin{bmatrix} Q_H(n_1,n_2) \\ Q_V(n_1,n_2)\end{bmatrix}. \tag{6}$$

III. 2-D Boundary State Space Equations

We can associate a transient with a change in the state of the system. Thus, we can suppress any transient associated with a boundary by not allowing a change in the state as the system crosses that boundary. Initial conditions are required for the (0, 0), $(0, n_2)$ and the $(n_1, 0)$ boundaries for the 2-D system with first quadrant support. Thus, we require

$$Q(0,0) = Q(-1,0) = Q(0,-1) \tag{7}$$

$$Q(n_1,0) = Q(n_1,-1);\quad Q(0,n_2) = Q(-1,n_2). \tag{8}$$

1057-7130/95$04.00 © 1995 IEEE