TWO NOVEL METHODS FOR COMPLEX ALLPASS FILTERS DESIGN USING REMEZ EXCHANGE ALGORITHM

Soo-Chang Pei[1] *Chien-Cheng Tseng*[2]

[1]Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R. O. C.
Email address: pei@cc.ee.ntu.edu.tw
[2]Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, R. O. C.
Email address: tcc@csp.ee.ntu.edu.tw

ABSTRACT

In this paper, two novel methods are proposed for designing complex allpass filters with equiripple phase responses using Remez exchange algorithm. One is based on solving simultaneous equation, the other is based on the maximum eigenvalue problem. In each method, satisfactory solution is obtained through a few iterations without any initial guess of the solution. One example is presented to show the effectness of the methods.

1. INTRODUCTION

In many signal processing applications, it is desired to design an allpass filter with prescribed phase or group delay characteristic [1]-[4]. Recently, two Remez algorithms are developed to design allpass filters with equiripple phase responses. One is that Ikehara presented a design algorithm to approximate the phase response of a complex coefficient allpass filter by solving linear simultaneous equations [5], the other is that Zhang proposed a design procedure to obtain the filter coefficients of real allpass filter by computing an eigenvector corresponding to maximum eigenvalue [6]. Both algorithms give quite satisfactory results through a few iterations without any initial guess of the solution.

In this paper, two novel methods are proposed for designing complex allpass filters with equiripple phase responses using Remez exchange algorithm. The design procedures of both methods are derived from the same key equation. In the first method, the filter coefficients are obtained by solving simultaneous equations. Although the solution and the number of iterations of this method are almost the same as those of Ikehara method, the computation load of our algorithm is less than Ikehara algorithm. As to the second method, the optimal filter coefficients are obtained by solving maximum eigenvector problem. In fact, the second algorithm is an extension of Zhang algorithm. The difference between them is that our algorithm considers complex coefficient allpass filter design, but Zhang algorithm concerns real coefficient allpass filter design.

2. COMPLEX ALLPASS FILTER

The transfer function of a N-th order complex allpass filter is defined as

$$A(z) = z^{-N}\frac{\sum_{n=0}^{N} a^*(n)z^n}{\sum_{n=0}^{N} a(n)z^{-n}} \qquad (1)$$

where $a(n)$ is a set of complex filter coefficients given by $a(n) = a_r(n) + ja_I(n)$ and $*$ denote complex conjugate. Since $|A(e^{j\omega})| = 1$ for all ω, we have

$$A(e^{j\omega}) = e^{j\theta(\omega)} \qquad (2)$$

where $\theta(\omega)$ is the phase response of $A(z)$. Assuming that the desired phase response is $\theta_d(\omega)$, the desired allpass filter frequency response $A_d(e^{j\omega})$ is given by

$$A_d(e^{j\omega}) = e^{j\theta_d(\omega)} \qquad (3)$$

The desired phase response of a stable complex allpass filter must have the following properties [5]:
(1) $\theta_d(2\pi) = \theta_d(\pi) - 2N\pi$.
(2) $\theta_d(\omega)$ decrease monotonically with increasing frequency.

Assuming that the phase error $\theta_e(\omega) = \theta(\omega) - \theta_d(\omega)$, then it can show that

$$\hat{E}(e^{j\omega}) = 2\sin(\frac{\theta_e(\omega)}{2}) \qquad (4)$$

where $\hat{E}(e^{j\omega})$ is the amplitude of the complex error $E(e^{j\omega})$ between $A(e^{j\omega})$ and $A_d(e^{j\omega})$ [5]. Since $\sin(x)$ monotonically increases as x increases, if an allpass filter can be designed such that $\hat{E}(e^{j\omega})$ is equiripple, $\theta_e(\omega)$ can be also equiripple and the maximum phase error is minimized. This observation is the basic idea of phase approximation algorithm developed by Ikehara [5].

Instead of using eq(4) to develop design algorithm, we derive another phase error relation as follows. From eq(1), we can show that

$$exp(j\theta_e(\omega)) = \frac{\sum_{n=0}^{N} a^*(n)exp[j(n\omega - \frac{N\omega+\theta_d(\omega)}{2})]}{\sum_{n=0}^{N} a(n)exp[-j(n\omega - \frac{N\omega+\theta_d(\omega)}{2})]} \qquad (5)$$

With some manipulation, we have

$$\Phi(\omega) = \tan(\frac{\theta_e(\omega)}{2}) = \frac{C(\omega)}{D(\omega)} \qquad (6)$$

where numerator and denominator of $\Phi(\omega)$ are given by

$$\begin{aligned} C(\omega) &= \sum_{n=0}^{N} -a_I(n)\cos(n\omega - \frac{N\omega+\theta_d(\omega)}{2}) \\ &+ a_r(n)\sin(n\omega - \frac{N\omega+\theta_d(\omega)}{2}) \\ D(\omega) &= \sum_{n=0}^{N} a_r(n)\cos(n\omega - \frac{N\omega+\theta_d(\omega)}{2}) \\ &+ a_I(n)\sin(n\omega - \frac{N\omega+\theta_d(\omega)}{2}) \end{aligned} \qquad (7)$$

0-7803-3073-0/96/$5.00 ©1996 IEEE

Since $\tan(x)$ monotonically increases as x increases, if an allpass filter can be designed such that function $\Phi(\omega)$ is equiripple, $\theta_e(\omega)$ can be also equiripple and maximum phase error is minimized. In next section, we do not approximate the desired phase response directly, but indirectly by using function $\Phi(\omega)$.

3. TWO DESIGN ALGORITHMS

In this section, two approximation algorithms are proposed for designing allpass filter with equiripple phase response by using function $\Phi(\omega)$. One is based on solving simultaneous equation, the other is based on maximum eigenvalue problem.

3.1 Design Algorithm 1

In order to use Remez exchange algorithm, we select $2(N+1)$ appropriate frequency points $\omega_i(i = 0, 1, \cdots, 2N+1)$ in the specified frequency range. Then, an allpass filter is designed such that filter coefficients satisfies following condition

$$\Phi(\omega_i) = (-1)^i \delta \tag{8}$$

where δ is an initial error. Combine eq(6)(7)(8), we obtain

$$\sum_{n=0}^{N} -a_I(n)\cos(n\omega_i - \frac{N\omega_i + \theta_d(\omega_i)}{2}) + a_r(n)\sin(n\omega_i - \frac{N\omega_i + \theta_d(\omega_i)}{2}) = (-1)^i \delta D(\omega_i) \tag{9}$$

Moreover, eq(9) can be represented in matrix form as

$$\begin{bmatrix} \sin\Psi(0,0) & -\cos\Psi(0,0) & \cdots & \sin\Psi(0,N) & -\cos\Psi(0,N) \\ \sin\Psi(1,0) & -\cos\Psi(1,0) & \cdots & \sin\Psi(1,N) & -\cos\Psi(1,N) \\ \vdots & \vdots & \cdots & & \\ \sin\Psi(2N,0) & -\cos\Psi(2N,0) & \cdots & \sin\Psi(2N,N) & -\cos\Psi(2N,N) \\ \sin\Psi(2N+1,0) & -\cos\Psi(2N+1,0) & \cdots & \sin\Psi(2N+1,N) & -\cos\Psi(2N+1,N) \end{bmatrix} \begin{bmatrix} a_r(0) \\ a_i(0) \\ \vdots \\ a_r(N) \\ a_i(N) \end{bmatrix} = \begin{bmatrix} \delta D(\omega_0) \\ (-1)\delta D(\omega_1) \\ \vdots \\ (-1)^{2N}\delta D(\omega_{2N}) \\ (-1)^{2N+1}\delta D(\omega_{2N+1}) \end{bmatrix} \tag{10}$$

where $\Psi(i,n) = n\omega_i - \frac{N\omega_i + \theta_d(\omega_i)}{2}$. The eq(10) has the same form as eq(16) in [5] except the vector in right side is modified. Thus, similar iterative approximation algorithm can be developed as follows:

Procedure of Algorithm 1:

Step 1. Read order N, initial δ, and the desired phase response $\theta_d(\omega)$.
Step 2. Select initial extremal frequencies $\omega_i(i = 0, 1, \cdots, 2N+1)$ equally spaced in the specified frequency range.
Step 3. Set $D(\omega_i) = 1$, $i = 0, 1, \cdots, 2N+1$.
Step 4. Compute a set of coefficients of a complex allpass filter by solving simultaneous equation eq(10).
Step 5. Calculate $\Phi(\omega)$ and search the extremal frequencies of $\Phi(\omega)$. Store them into ω_i'.
Step 6. If $|\omega_i' - \omega_i| \le \epsilon$, then go to *Step 10.* Otherwise go to *Step 7.* Note that ϵ is a prescribed small number.
Step 7. Let $\omega_i = \omega_i'$ and set $\delta = \frac{1}{2(N+1)} \sum_{n=0}^{2N+1} |\Phi(\omega_i)|$.
Step 8. Compute $D(\omega_i)$ according to eq(7).
Step 9. Go to *Step 4.*
Step 10. Compute the transfer function $A(e^{j\omega})$ and plot the phase response.

In this algorithm, we start with $D(\omega) = 1$ and continue $D(\omega)$ calculated by an obtained set of coefficients on the previous pass. As to the convergence, this algorithm has same behavior as algorithm in [5].

3.2 Design Algorithm 2

In order to formulate the approximate problem into a maximum eigenvalue problem, substitute eq(6) into eq(8) to yield

$$C(\omega_i) = (-1)^i \delta D(\omega_i) \tag{11}$$

We can rewrite eq(11) in matrix form as

$$\mathbf{P}\mathbf{a} = \delta \mathbf{Q}\mathbf{a} \tag{12}$$

where matrix $\mathbf{P}$,$\mathbf{Q}$ and vector $\mathbf{a}$ are given by

$$\mathbf{P} = \begin{bmatrix} \sin\Psi(0,0) & -\cos\Psi(0,0) & \cdots & \sin\Psi(0,N) & -\cos\Psi(0,N) \\ \sin\Psi(1,0) & -\cos\Psi(1,0) & \cdots & \sin\Psi(1,N) & -\cos\Psi(1,N) \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \sin\Psi(2N,0) & -\cos\Psi(2N,0) & \cdots & \sin\Psi(2N,N) & -\cos\Psi(2N,N) \\ \sin\Psi(2N+1,0) & -\cos\Psi(2N+1,0) & \cdots & \sin\Psi(2N+1,N) & -\cos\Psi(2N+1,N) \end{bmatrix}$$

$$\mathbf{Q} = \begin{bmatrix} \cos\Psi(0,0) & \sin\Psi(0,0) & \cdots & \cos\Psi(0,N) & \sin\Psi(0,N) \\ -\cos\Psi(1,0) & -\sin\Psi(1,0) & \cdots & -\cos\Psi(1,N) & -\sin\Psi(1,N) \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ \cos\Psi(2N,0) & \sin\Psi(2N,0) & \cdots & \cos\Psi(2N,N) & \sin\Psi(2N,N) \\ -\cos\Psi(2N+1,0) & -\sin\Psi(2N+1,0) & \cdots & -\cos\Psi(2N+1,N) & -\sin\Psi(2N+1,N) \end{bmatrix}$$

$$\mathbf{a} = [\ a_r(0) \quad a_I(0) \quad \cdots \quad a_r(N) \quad a_I(N)\ ]^t$$

Follow the same reason in real case [6], we have $\mathbf{P}$ is a nonsingular matrix. Thus, the eq(12) can be rewritten as

$$\frac{1}{\delta}\mathbf{a} = \mathbf{P}^{-1}\mathbf{Q}\mathbf{a} \tag{13}$$

The eq(13) has the same form as eq(10) in [6] except matrix $\mathbf{P}$,$\mathbf{Q}$, and vector $\mathbf{a}$ have different definition. Thus, a similar iterative algorithm can be stated as follows:

Procedure of Algorithm 2:

Step 1. Specify order N and the desired phase response $\theta_d(\omega)$.
Step 2. Select initial extremal frequencies $\omega_i(i = 0, 1, \cdots, 2N+1)$ equally spaced in the specified frequency range.
Step 3. Compute matrix $\mathbf{P}$, $\mathbf{Q}$ and find the maximum eigenvector of $\mathbf{P}^{-1}\mathbf{Q}$ to obtain the coefficient vector $\mathbf{a}$.
Step 4. Calculate $\Phi(\omega)$ and search the extremal frequencies of $\Phi(\omega)$. Store them into ω_i'.
Step 5. If $|\omega_i' - \omega_i| \le \epsilon$, then go to *Step 6.* Otherwise set $\omega_i = \omega_i'$ and go to *Step 3.* Note that ϵ is a prescribed small number.
Step 6. Compute the transfer function $A(e^{j\omega})$ and plot the phase response.

4. DESIGN EXAMPLE AND CONCLUSION

In this section, one design example is presented to compare the performance between our approaches and Ikehara's method. The desired phase response considered here is given by

$$\theta_d(\omega) = -9\omega + 2\pi\sin(\frac{\omega}{2}) \qquad 0 \leq \omega \leq 2\pi \qquad (14)$$

A ninth-order allpass filter is designed, the obtained phase response and phase error are depicted in Fig.1. We can see that the phase error are equiripple with 20 extrema for each proposed algorithm. Table 1 lists the obtained filter coefficients. It is obvious that the proposed methods almost have the same coefficients as Ikehara's method. Moreover, Table 2 illustrates the convergence behavior of the peak phase error. From this reseult, we see that the proposed algorithm 2 has the fastest convergence speed among three design methods. Finally, the CPU time and the resultant maximum phase error are listed in Table 3. It is clear that both proposed algorithms take less design time than Ikehara's method, but the peak phase errors are all identical.

In this paper, two effective methods have been proposed for designing complex IIR allpass filters. One example is presented to show the effectness of the methods.

REFERENCES

[1] P.A.Regalia, S.K.Mitra, and P.P.Vaidyanathan, "The digital allpass filter: A versatile signal processing building block", Proc. IEEE, Vol.76, pp.19-37, Jan. 1988

[2] M. Lang, "Allpass Filter Design and Applications", In IEEE Proc. ICASSP, pp.1264-1267, 1995

[3] A.G.Deczky, "Equiripple and minimum (Chebyshev) approximations for recusive digital filters", IEEE Trans. Acoust. Speech, Signal Proc. Vol.22, pp.98-111, April 1974

[4] Z.Jing, "A new method for digital allpass filter design", IEEE Trans. Acoust., Speech, Signal Proc., Vol.35, pp.1557-1564, Nov. 1987

[5] M.Ikehara, M.Funaishi, and H.Kuroda, "Design of complex all-pass networks using Remez algorithm", IEEE Trans. Circuits and Syst.-II: Analog and Digital Signal Processing, Vol.39, pp.549-556, Aug. 1992

[6] X.Zhang and H.Iwakura, "Novel method for designing digital allpass filters based on eigenvalue problem", Electronics Letters, Vol.29, pp.1279-1281, July 1993

Table 1. The obtained filter coefficients of complex allpass filter.

a(n)	Ikehara method	Algorithm 1	Algorithm 2
a(0)	0.2103299 - 0.4603836 i	0.2103299 - 0.4603836 i	0.2103299 - 0.4603836 i
a(1)	- 0.6144316 - 0.2807849 i	- 0.6144316 - 0.2807849 i	- 0.6144316 - 0.2807849 i
a(2)	- 0.3107566 + 0.3534777 i	- 0.3107566 + 0.3534777 i	- 0.3107566 + 0.3534777 i
a(3)	0.0537756 + 0.2233129 i	0.0537756 + 0.2233129 i	0.0537756 + 0.2233129 i
a(4)	0.0673107 + 0.0624107 i	0.0673107 + 0.0624107 i	0.0673107 + 0.0624107 i
a(5)	0.0293758 + 0.0188086 i	0.0293758 + 0.0188086 i	0.0293758 + 0.0188086 i
a(6)	0.0067660 + 0.0121137 i	0.0067660 + 0.0121137 i	0.0067660 + 0.0121137 i
a(7)	- 0.0022844 + 0.0256163 i	- 0.0022844 + 0.0256163 i	- 0.0022844 + 0.0256163 i
a(8)	0.0264943 + 0.0323728 i	0.0264943 + 0.0323728 i	0.0264943 + 0.0323728 i
a(9)	0.0275723 - 0.0125971 i	0.0275723 - 0.0125971 i	0.0275723 - 0.0125971 i

Table 2. The convergence behavior of the peak phase error.

iteration number peak phase error	1	2	3	4	5	6	7	8	9
Ikehara method	0.58131	0.94884	0.79960	0.24774	0.13904	0.10418	0.10146	0.10135	0.10135
Algorithm 1	0.58131	0.95193	0.81162	0.25177	0.14222	0.10468	0.10144	0.10135	0.10135
Algorithm 2	0.56641	0.86098	0.67147	0.26188	0.13027	0.10247	0.10135	0.10135	0.10135

Table 3. The CPU time and peak phase error.

	Ikehara method	Algorithm 1	Algorithm 2
peak phase error	0.10135	0.10135	0.10135
CPU time	67.83 sec	66.79 sec	57.89 sec

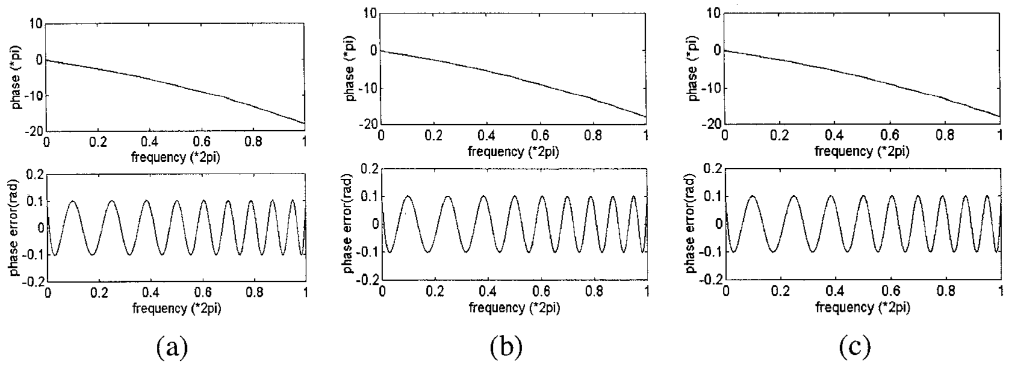

Fig.1 . Phase response and phase error (a) Ikehara method (b) Algorithm 1 (c) Algorithm 2.