ALLPASS FILTER DESIGN AND APPLICATIONS

Markus Lang

Dept. of ECE, Rice University, Houston, TX-77251, email: lang@ece.rice.edu

ABSTRACT

The problem of allpass filter design for phase approximation and equalization in the Chebyshev sense is solved by using a generalized Remez algorithm. Convergence to the unique optimum is guaranteed and is achieved rapidly in the actual implementation. The well-known numerical problems for higher degree filters are analyzed and solved by a simple approach. Possible applications are: design of filters with a desired phase response (e.g., a delay element), the design of phase equalizers, or the design of recursive filters with magnitude prescriptions using parallel allpass filters. For the latter the algorithm can be modified to allow arbitrary tolerance schemes for the magnitude response.

1. INTRODUCTION

Phase approximation might be considered a somewhat academic problem but can be used in a variety of scenarios. Obviously it is applicable for any recursive allpass filter design problem. It might also be used for allpole filter design or FIR filter design if one is only interested in a phase prescription. Some of the numerous allpass filter applications are: approximation of a prescribed phase, e.g., a linear phase (fractional delay element), equalization of the phase of a given system, design of a Hilbert transformer, or design of recursive filters with a desired magnitude response using a parallel allpass structure. The latter application is very appealing since 1) it facilitates the implementation of a degree n recursive (e.g., low pass) filter with only n multiplications, 2) it permits the realization of the complementary (e.g., high pass) filter with no extra multiplications, and 3) it exhibits small sensitivity to coefficient quantization.

The most common approaches for the phase approximation problem are 1) maximally flat (mostly at zero frequency) approximation, 2) L_2 approximation, or 3) L_∞ approximation. For the first problem there exist straightforward analytic solutions (see [8] for references). Since there are only pointwise prescriptions, one has limited control over the approximation quality in the interval of interest. This does not hold for an L_2 or an L_∞ approximation. There an iterative procedure is required.

In the recent literature the L_2 approach seems to have been dominated [2, 9, 10, 6]. However, there are severe problems. It is not clear whether the solution is unique or how the optimal solution is characterized. Consequently none of the proposed algorithms is shown to converge to the L_2 solution. These difficulties arise because the approximation problem is nonlinear.

On the contrary, it can be shown [3, 7, 8] that for the L_∞ problem the solution is uniquely characterized by the alternation theorem. Furthermore, there exists an efficient Remez type algorithm which is guaranteed to converge. It is worthwhile mentioning that most publications dealing with the L_∞ problem do not utilize these results. Consequently they either propose ad hoc generalizations of the standard Remez algorithm for linear approximation problems to the nonlinear case with possible convergence problems [4, 11, 12] or they use iterative least squares approaches [2, 6] that are inefficient compared to the approach in this paper.

In the following section we pose the problem mathematically and relate it to some important applications mentioned above. In Sec. 3 we give a Remez type algorithm which is guaranteed to converge to the L_∞ solution in contrast to all alternative approaches in the literature. An analysis of the numerical problems and their resolution are presented in Sec. 4 for the first time. Another novelty is the possibility of designing recursive filters with an arbitrary magnitude tolerance scheme on the basis of the parallel allpass structure, as described in Sec. 5

2. THE PROBLEM

The problem under consideration is the design of a real valued allpass filter with the transfer function

$$H_A(z) = \frac{z^n P(z^{-1})}{P(z)}, \quad P(z) = z^n + \sum_{\nu=0}^{n-1} p_\nu z^\nu \tag{1}$$

such that its phase ($=-\arg\{H_A(e^{j\Omega})\}$)

$$b_A(\Omega) = -n\Omega + 2\arctan \frac{\sum_{\nu=0}^{n} p_\nu \sin(\nu\Omega)}{\sum_{\nu=0}^{n} p_\nu \cos(\nu\Omega)} \tag{2}$$

approximates a prescribed phase $b_{pre}(\Omega)$ in the Chebyshev sense. More generally we want to minimize the L_∞ norm or Chebyshev norm of the weighted phase error

$$e_b(\Omega) = G_b(\Omega)\,[b_A(\Omega) - \tau_0\Omega + b_0 - b_{pre}(\Omega)] \tag{3}$$

with respect to the coefficients p_ν and the parameters τ_0, b_0. These two additional parameters may be used for the equalization problem [7, 8, 9] and the adaptation of a phase offset for bandpass filters, respectively. If not needed they are simply set to zero in Eq. (3). Note that the approximation problem is nonlinear since the parameters p_ν appear nonlinearly in the error function.

Applications: Fig. 1 shows two important applications of an allpass filter design. In the case of phase equalization (Fig. 1(a)) of a low pass filter one is interested in determining a stable $H_A(z)$ such that $-\arg\{H(e^{j\Omega})\} = b_g(\Omega) + b_A(\Omega) \approx \tau_0\Omega$ (Ω in pass band of $H_g(e^{j\Omega})$) holds.

0-7803-2431-5/95 $4.00

$b_g(\Omega)$ denotes the phase of $H_g(e^{j\Omega})$ and τ_0 is some unspecified slope used to improve the approximation. If one identifies $b_g(\Omega)$ and $b_{pre}(\Omega)$, sets $b_0 = 0$, and $G_b(\Omega) = 1$, this equalization problem is equivalent to minimizing $e_b(\Omega)$ in Eq. (3).

A second interesting application, depicted in Fig. 1(b), is the parallel allpass structure [3, 4, 7, 8, 11] with the overall transfer function

$$H_{1,2}(z) = [H_{A1}(z) \pm H_{A2}(z)]/2. \tag{4}$$

The corresponding magnitude response

$$Q_1(\Omega) = |H_1(e^{j\Omega})| = |\cos(\Delta b(\Omega)/2)|, \tag{5}$$

is completely determined by $\Delta b(\Omega)$, the phase difference between allpass 2 and allpass 1. I.e., in order to describe a low pass frequency response $Q_1(\Omega)$, $\Delta b(\Omega)$ has to approximate 0 in the pass band and π in the stop band. Moreover, $Q_1(\Omega)$ completely determines

$$Q_2(\Omega) = |H_2(e^{j\Omega})| = \sqrt{1 - Q_1^2(\Omega)} \tag{6}$$

Thus it is possible to design recursive digital filters with magnitude prescriptions by solving a phase approximation problem.

3. THE SOLUTION

In [1] Barrar and Loeb present a generalized Remez algorithm which converges to the optimal solution if the approximating function possesses certain properties. It can be shown that the allpass phase function belongs to this class [3, 7, 8]. The algorithm consists of two basic steps 1) determination of $m+1$ (m = number of approximation parameters) extremal frequencies Ω_i of $e_b(\Omega)$ with alternating signs and 2) interpolation and adaptation (increase) of the actual error level δ. After convergence, δ corresponds to the Chebyshev error. In most cases the Remez algorithm converges within 6 iteration steps.

Since step 1) is easily performed it need not be discussed in more detail. The interpolation step 2), however, requires the solution of

$$\sum_{\nu=0}^{n} p_\nu \sin\left[\nu\Omega_i - \frac{1}{2}\left(\frac{(-1)^i\delta}{G_b(\Omega_i)} + b_{pre}(\Omega_i) + (\tau_0 + n)\Omega_i - b_0\right)\right] = 0, \quad i = 1\ldots m+1 \tag{7}$$

for $p_0 \ldots p_{n-1}$, τ_0, b_0, and δ [7, 8]. In contrast to the classical Remez algorithm this is a nonlinear system of equations. It is efficiently solved with Newton's method. There are typically less than 5 Newton steps required. Each step corresponds to the solution of an $m+1$ by $m+1$ system of linear equations.

Initialization: Similar to the classical Remez algorithm one has to find an initial solution with the alternation property of the error function $e_b(\Omega)$. This corresponds to demanding $\delta = 0$ in Eq. (7) (for m frequencies only). In the special case of fixed τ_0 and b_0, Eq. (7) becomes a system of linear equations and is thus easily solved. Otherwise (τ_0 or b_0 variable) one has to solve a system of nonlinear equations which is solved similarly to the interpolation step 2) in the Remez algorithm. The same initialization for τ_0, b_0 as in [9] can be used. In contrast to the initialization in [7] this approach explicitly includes the required alternation property of $e_b(\Omega)$. Furthermore, it yields a considerable improvement of the computation time compared to [7].

Example 1: The phase of an elliptic low pass filter with pass band frequency $\Omega_p = 0.1\pi$ and degree 7 is equalized by an allpass filter of degree $n = 9$. Fig. 2 shows the resulting phase error and the overall group delay response (corresponding to $H(e^{j\Omega})$ in Fig. 1(a)). For comparison, the results of an approximate L_2 solution according to [9] is included in Fig. 1(b), exhibiting a larger Chebyshev error. The design takes about 2s on a SPARC10 workstation.

4. NUMERICAL PROBLEMS

It is well-known [5, 7, 8] that the maximal usable allpass degree is limited by numerical problems. Jing [5] observes this problem especially in the context of phase equalization. He gives a heuristic (yet incorrect) explanation for this behaviour and suggests the design of a filter of larger degree where one additionally prescribes zeros of $P(z)$ on the unit circle. According to his experience the "best number" of these additional zeros can be determined by minimizing a certain function. Although the proposed method improves the numerical behaviour it is important to note that the numerical problems are not analyzed in [5]. Also, the function he proposes to minimize is given without justification.

In contrast to this approach the numerical problems are analyzed here and resolved by a similar but well-founded, more powerful and more effective method without the need of minimizing any function.

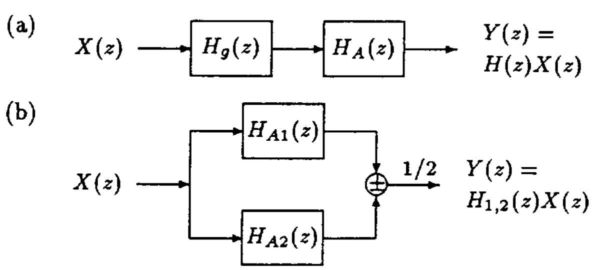

Figure 1: Two important applications of phase approximation. (a) Phase equalization of a given system. (b) Recursive filter design using a parallel allpass structure.

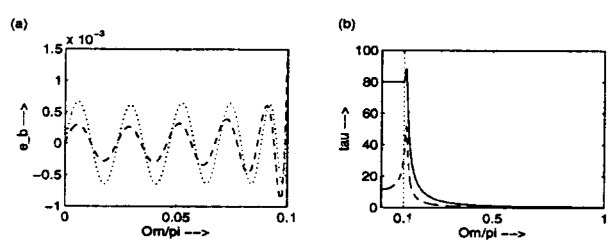

Figure 2: Example 1, phase equalization. (a) Phase error of Chebyshev (dotted) and approximate L_2 (dashed) solution. (b) Group delay of $H_g(z)$ (dashed) and the overall system $H(z)$ (solid).

Analysis: It is possible to find a way to considerably reduce the numerical problems by analyzing the problem described in the following. The phase of the filter in example 1 shall be equalized with an allpass filter of degree $n = 12$. In Fig. 3(a) the phase error function resulting at some step of the initialization algorithm is depicted. As is clearly seen in the figure the computed phase error shows a noisy behaviour. It is obvious that one can not properly perform the first step of the Remez algorithm (determination of $m+1$ extrema with alternating sign).

This problem occurs because the phase of the actual allpass filter can not be computed with sufficient numerical accuracy. Since the allpass filter transfer function is completely determined by the denominator polynomial $P(z)$, the problems in computing the phase must emanate from numerical inaccuracies in the course of evaluating the polynomial $P(z)$.

For the subsequent analysis it is assumed that the polynomial $P(z)$ has the roots

$$z_i = re^{j\varphi_i}, \quad i = 1 \dots n \tag{8}$$

$$\varphi_i = \Omega_p\left(-1 + 2\frac{i-1}{n-1}\right), \quad i = 1 \dots n, \tag{9}$$

i.e., they are equally distributed on an arc of radius r and angle $2\Omega_p$. This is a good approximation to the actual distribution for large allpass degrees (lowpass equalization).

The chain of causes for the numerical problems consists of the following three elements:

1. The numerical evaluation of the polynomial $P(z)$ in the points $z = e^{j\Omega}$ has to be performed within the given computing accuracy ϵ. The resulting error may be interpreted as follows: a polynomial $\tilde{P}(z)$ with coefficients perturbed by values in the order of ϵ is evaluated *exactly* for each z. The perturbation randomly varies with the actual value of z.

2. A perturbation of the coefficients p_ν in the order of ϵp_ν results in a perturbation of the roots of the polynomial. In the worst case this perturbation is amplified by the condition number [8]

$$\kappa(P(z)) = \frac{|p_\lambda| r^\lambda}{(2r)^{n-1}} \prod_{\substack{i=1 \\ i \neq \lambda}}^{n} \left|\sin\frac{(\lambda - i)\Omega_p}{(n-1)}\right|^{-1}, \tag{10}$$

where $\lambda = \text{int}((n+1)/2)$ ($\text{int}(x)$ = integer portion of x).

3. The perturbation of the root locations results in a perturbation of the phase response, which can be observed in Fig. 3(a).

A more detailed analysis shows that the by far dominating effect is the one described under **2.** (cf. [8]). To verify the usefulness and correctness of the results above consider the case corresponding to the Fig. 3(a) where $n = 12$, $\Omega_p = 0.1\pi$, and $r \approx 0.94$. The resulting condition number according to Eq. (10) is $4.6 \cdot 10^{11}$. Thus a maximum error of the order of

$$\kappa(P(z))\epsilon = 4.6 \cdot 10^{11} \cdot 2.204 \cdot 10^{-16} \approx 10^{-4} \tag{11}$$

is to be expected which nicely coincides with the observed noise level in Fig. 3(a).

The discussion above leads to the conclusion that the cause of the numerical problems lies in the high concentration of roots on a small arc. This coincides with the well-known fact that among all polynomials those with equally distributed roots on a circle have the smallest condition number. To improve the conditioning of the problem, the polynomial $P(z)$ arising in the course of the algorithm has to be substituted by another polynomial $C(z)$ such that 1) the original polynomial $P(z)$ is a factor of $C(z)$ and 2) the roots of $C(z)$ approximately lie on a circle. This is done (in a suboptimal manner) by prescribing the additional zeros of $C(z)$ on the unit circle, outside the passband of the filter to be equalized. This works because these additional zeros do not change the phase response. The required number and location of these zeros can be explicitly computed from Ω_p and n. Note that the additional zeros are maintained throughout the algorithm. Consequently, Eq. (7) has to be formulated in terms of $C(z)$, instead of $P(z)$. Upon convergence, the desired polynomial $P(z)$ can be found by factoring $C(z)$.

Example 2: The approach described above allows the usage of considerably increased allpass degrees. The maximum possible n for the example 1 without the modifications is 10 with Chebyshev error $\delta = 4.09 \cdot 10^{-4}$. By applying the proposed method it is possible to use a degree $n = 20$ which results in $\delta = 2.77 \cdot 10^{-7}$. The corresponding phase error is depicted in Fig. 3. For some cases the improvement is considerably higher. The maximum degree for equalizing the lowpass filter with the (incomplete) specifications in [5] is 150 with $\delta = 1.06 \cdot 10^{-4}$ (compared to 25 without modifications).

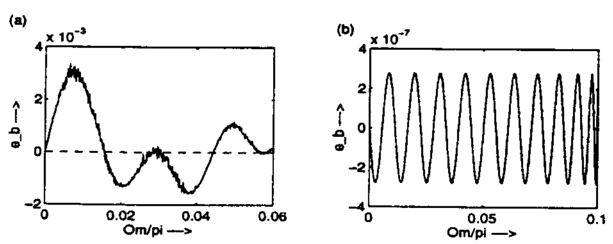

Figure 3: Phase error resulting from the phase equalization of a degree 7 low pass filter with $\Omega_p = 0.1\pi$. (a) Intermediate result of the original initialization algorithm with allpass degree $n = 12$. (b) Final result of the modified Remez algorithm with allpass degree $n = 20$.

5. ARBITRARY MAGNITUDE TOLERANCES

The design of recursive filters with arbitrary tolerance prescriptions for the magnitude response is based on the parallel allpass structure depicted in Fig. 1(b). The method is shown for a low pass filter described by $H_1(z)$.

The desired magnitude tolerance function $\delta_{dH}(\Omega)$ can be related to a desired magnitude weighting function

$$G_{dH}(\Omega) = \delta_{dH}(\Omega_r)/\delta_{dH}(\Omega), \quad \Omega \in \mathcal{B}, \tag{12}$$

where Ω_r denotes some arbitrary reference frequency and $\mathcal{B}$

the set of interest. Based on the Eqs. (4, 5) the relationship

$$G_H(\Omega) = \frac{\delta_{Hr}}{\delta_H(\Omega)} = \begin{cases} \frac{\sin^2(\delta_{br}/4)}{\sin^2(\delta_b(\Omega)/4))}, & \Omega \in \mathcal{B}_p, \\ \frac{2\sin^2(\delta_{br}/4))}{\sin(\delta_b(\Omega)/2))}, & \Omega \in \mathcal{B}_s, \end{cases} \quad (13)$$

between the magnitude weighting function $G_H(\Omega)$, the magnitude tolerance $\delta_H(\Omega)$, and the phase tolerance $\delta_b(\Omega)$ can be derived [8]. $\mathcal{B}_p$ and $\mathcal{B}_s$ denote the pass and the stop band interval and $\delta_{br} = \delta_b(\Omega_r)$, $\delta_{Hr} = \delta_H(\Omega_r)$. Note that $G_H(\Omega)$, $\delta_H(\Omega)$, and $\delta_b(\Omega)$ are the weighting and tolerance functions corresponding to some actual filter response (and not to the desired ones, mentioned above).

A Taylor series expansion of the right hand side of Eq. (13) yields [8]

$$G_H(\Omega) \approx G_b(\Omega) \cdot \begin{cases} G_b(\Omega)\left(1 + \frac{\delta_b^2(\Omega) - \delta_{br}^2}{48}\right), & \Omega \in \mathcal{B}_p, \\ \left(\frac{\delta_{br}}{4} + \frac{\delta_{br}\delta_b^2(\Omega)}{96} - \delta_{br}^3\right), & \Omega \in \mathcal{B}_s, \end{cases} \quad (14)$$

where $G_b(\Omega) = \delta_{br}/\delta_b(\Omega)$. The relation (14) can be used to construct an iterative update of the phase weighting function $G_b(\Omega)$. This is done by solving (14) for $G_b(\Omega)$, replacing $G_H(\Omega)$ by the desired weighting $G_{dH}(\Omega)$ and introducing an iteration index. That is

$$G_b^{(k+1)}(\Omega) = \begin{cases} \sqrt{\frac{48 G_{dH}(\Omega)}{48 + (\delta_b^{(k)}(\Omega))^2 - (\delta_{br}^{(k)})^2}}, & \Omega \in \mathcal{B}_p, \\ \frac{192 G_{dH}(\Omega)}{\delta_{br}^{(k)}\left[48 + 2(\delta_b^{(k)}(\Omega))^2 - (\delta_{br}^{(k)})^2\right]}, & \Omega \in \mathcal{B}_s. \end{cases} \quad (15)$$

The algorithm for designing a filter with a desired magnitude weighting function $G_{dH}(\Omega)$ can be summarized as follows:

1. Determine L_∞ solution for $\Delta b(\Omega)$ using $G_b^{(0)}(\Omega) = 1$.
2. $k \rightarrow k+1$.
3. Determine the actual phase tolerance $\delta_b^{(k-1)}(\Omega) = \delta_{br}^{(k-1)}/G_b^{(k-1)}(\Omega)$ and compute the new weighting function $G_b^{(k)}(\Omega)$ according to Eq. (15).
4. Compute a Chebyshev solution using the new weighting function and the allpass coefficients of the previous iteration step as an initial solution.
5. Stop if the algorithm has converged, repeat at **2.** otherwise.

Example 3: A recursive filter is designed based on the structure depicted in Fig. 1 with $H_{A1}(z) = z^{-12}$ (leading to a linear phase of $H_{1,2}(e^{j\Omega})$) and $H_{A2}(z)$ of degree 13. The pass band and stop band frequencies are $\Omega_p = 0.4\pi$ and $\Omega_s = 0.5\pi$, respectively. Fig. 4(a) shows the overall magnitude responses $|H_1(e^{j\Omega})|$ (solid, low pass) and $|H_2(e^{j\Omega})|$ (dotted, high pass). Fig. 4(b) depicts the resulting magnitude error of $|H_1(e^{j\Omega})|$ which keeps exactly the desired tolerance scheme (dotted line). The dashed line indicates the magnitude error resulting from a constant tolerance prescription in both bands.

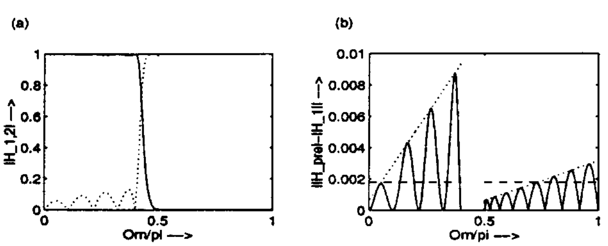

Figure 4: Example 3, parallel allpass structure. (a) Magnitude responses. (b) Magnitude error of $|H_1(e^{j\Omega})|$.

6. SUMMARY

This paper describes a more general approach to phase approximation than known in the literature (inclusion of phase offset). It is shown by examples that this general formulation facilitates the solution of many problems related to phase approximation. A generalized Remez algorithm is used to achieve a Chebyshev solution. Convergence and optimality are guaranteed. The initialization is considerably improved compared to [7] (the only paper dealing with this problem). An analysis of the numerical problems is performed for the first time. It facilitates the construction of an algorithm which allows the design of allpass filters with a considerably increased degree. Furthermore there is an algorithm presented for the design of recursive filters with arbitrary prescriptions for the magnitude error based on the parallel allpass structure. A journal paper with a more detailed discussion is in preparation. MATLAB files are available from the author or from the World Wide Web: `http://jazz.rice.edu`.

7. REFERENCES

[1] Barrar, R. B. and H. L. Loeb. On the Remez algorithm for non-linear families. *Num. Math., Springer*, 15:382–391, 1970.

[2] Chen, C.-K.,, and J.-H. Lee. Design of digital all-pass filters using a weighted least squares approach. *IEEE Tr. CAS II*, 41:346–350, May 1994.

[3] Gerken, M., H. W. Schüßler, and P. Steffen. On the design of digital filters consisting of a parallel connection of allpass sections and delay elements. *AEÜ*, 49(1):1–11, 1995.

[4] Ikehara, M., H. Tanaka, and H. Kuroda. Design of IIR digital filters using allpass networks. *IEEE Tr. CAS II*, 41:231–235, March 1994.

[5] Jing, Z. A new method for digital all-pass filter design. *IEEE Tr. ASSP*, 35:1557–1564, November 1987.

[6] Laakso, T. I., Truong Q. N., and R. D. Koilpillai. Designing allpass filters using the eigenfilter method. In *IEEE Proc. ICASSP*, pp III–77–III–80, 1993.

[7] Lang, M. Optimal weighted phase equalization according to the l_∞-norm. *EURASIP SP*, 27:87–98, April 1992.

[8] Lang, M. *Ein Beitrag zur Phasenapproximation mit Allpässen (Phase Approximation using Allpass Filters, in German).* PhD thesis, University of Erlangen-Nürnberg, Germany, 1993.

[9] Lang, M., and T. I. Laakso. Simple and robust method for the design of allpass filters using least squares phase error criterion. *IEEE Tr. CAS II*, pp 40–48, 1994.

[10] Pei, S.-C., and J.-J. Shyu. Eigenfilter design of 1-D and 2-D IIR digital all-pass filters. *IEEE Tr. SP*, 42:966–968, April 1994.

[11] Renfors, M., and T. Saramäki. A class of approximately linear phase digital filters composed of allpass subfilters. In *IEEE Proc. ISCAS*, pp 678–681, 1986.

[12] Schüßler, H. W., and J. Weith. On the design of recursive Hilbert transformers. In *IEEE Proc. ICASSP*, pp 876–879, 1987.