A NEW FILTER APPROXIMATION AND DESIGN ALGORITHM

Li Ping, R. K. Henderson and J. I. Sewell

Department of Electronics and Electrical Engineering
University of Glasgow, Glasgow G12 8QQ, UK

ABSTRACT

Two complementary algorithms, applying Newton's method to approximation and passive ladder filter design, are presented. Amplitude responses with a specified weighting or derivative value can be computed. The numerical ill-conditioning of conventional ladder synthesis methods are avoided by an extended version of Orchard's algorithm. The techniques have been employed in a filter compiler and many practical difficulties are covered.

INTRODUCTION

General filter amplitude functions are required in many applications, such as transmission line equalisation and sinc(x) correction [1]. Optimisation techniques must be applied to solve the general approximation problem. One of the most important, Newton's method, is now shown to provide a simple and efficient algorithm for this purpose. The well-known Remez exchange algorithm [2] is revealed as a modified version of the Newton algorithm. A further generalisation is introduced whereby a high degree of tangency with a bounding function may be assigned to certain points (touch points). A family of responses is produced which can be continuously varied between Butterworth, Chebyshev, Inverse Chebyshev and elliptic, offering a useful compromise between amplitude and group delay [3,4]. By smoothing the rapid amplitude rippling normally found near the passband edge the sensitivity behavior is expect to improve in this region. Demands on delay-correcting all-pass stages, known to be a significant source of noise for circuits implemented in CMOS technology, can be lessened. The capability of general amplitude approximation can also be applied to prewarp the response to compensate for distortions due to improper realisation or non-ideal circuit effects, particularly in SC realisations.

Results from the approximation procedure can be passed directly to an extended version of Orchard's iterative [5] method for passive ladder design. The component values are adjusted to match the magnitude of the reflection function at the touch points. Hurwitz factorisation and insertion-loss synthesis [6,7] are avoided, with their associated accuracy loss and ill-conditioning. Traditional methods, however, are still useful in providing initial guesses of component values for iterative design. The combination of the two methods, provides both efficiency and accuracy, especially for high order filters. The work is implemented in a new filter design package [8].

APPROXIMATION METHOD

Consider the problem of finding a realisable transfer function, $T(j\omega)$ which is usually a rational polynomial, to approximate some frequency response specification. An identity is introduced to avoid working with complex variables

$$|T(j\omega)|^2 = T(j\omega)T(-j\omega) = \hat{T}(x)|_{x=-\omega^2} \tag{1}$$

Consider the problem of fitting a polynomial $p(x)=a_nx^n+\ldots+a_0$ in between two boundary functions $u(x)>l(x)$. At a series of points, the so-called *touch points*, $p(x)$ will touch $u(x)$ and $l(x)$ alternately, which implies that $p(x)$ will have the same zero and derivative values up to certain order as $u(x)$ or $l(x)$. At M points (the touch points) on the upper and the lower function, $\{x_{ti} \mid a<x_{ti}<x_{ti+1}<b\}$

$$p^{(r)}(x_{ti}) = u^{(r)}(x_{ti}) \text{ or } l^{(r)}(x_{ti}) \quad r=0,1,2,\ldots,\mu_i \tag{2}$$

where $i=1,2,3,\ldots,M$. The exact locations of $\{x_{ti}\}$ are unknown but the sequence $\{\mu_i\}$ is specified Fig. 1. For convenience we fix the two end points by $p(a)=A$ and $p(b)=B$ (A and B are in between $u(x)$ and $l(x)$) then altogether there are N_c specifications on the values and the derivatives of $p(x)$, where

$$N_c = 2 + \sum_{i=1}^{M} (\mu_i+1) \tag{3}$$

The aim of the curve fitting problem is to find the lowest order approximating polynomial which fits the specification of (2). The unknown positions of $\{x_{ti}\}$ provide M degrees of freedom, which can be used to reduce the order of the polynomial from the nominal problem order N_c. Thus N_c-M of the specifications can be chosen as constraints to form a polynomial of order N, where

$$N + 1 = N_c - M \tag{4}$$

The remaining M specifications must be met by adjusting the M positions of the touch points.

A N^{th} order polynomial can always be interpolated by N+1 constraints. *Osculatory Newton interpolation* [9] can be used to interpolate a number of derivatives with certain computational advantages over other interpolation methods. In particular, round-off errors can be minimised by a zig-zag path method [9]. Assume that for the specifications,

i) all μ_i are odd,

ii) the touch points are assigned alternately to $u(x)$ and $l(x)$, i.e. $\{x_{ti} \mid i=1,3,\ldots,M_u\}$ and $\{x_{ti} \mid i=2,4,\ldots,M_l\}$ are the set of touch points on $u(x)$ and $l(x)$ respectively (where $M_u=M$ and $M_l=M-1$ if M is odd and $M_u=M-1$ and $M_l=M$ if M is even).

Interpolate $p(x)$ such that,

$$p^{(r)}(x_{ti}) = u^{(r)}(x_{ti}) \qquad r = 0,1,\ldots,\mu_i-1 \quad i = 1,3,5,\ldots,M_u \tag{5a}$$

$$p^{(r)}(x_{ti}) = l^{(r)}(x_{ti}) \qquad r = 0,1,\ldots,\mu_i-1 \quad i = 2,4,6,\ldots,M_l \tag{5b}$$

$$\text{and} \qquad p(a) = A \qquad p(b) = B \tag{6}$$

Thus exactly N_c-M specifications are met by interpolation. It now remains to adjust $\{x_{ti}\}$ to make $\{p^{(\mu_i)}(x_{ti})\}$ satisfy the other specifications.

Definitions:

upper error function $e_u(x) = p(x) - u(x)$
lower error function $e_l(x) = l(x) - p(x)$
mid function $m(x) = (l(x) + u(x))/2$
search function $s(x) = \max[\, e_u(x),\, e_l(x) \,]$

$$\textit{combined error function } e(x) = \begin{cases} -e_u(x) & \text{if } p(x)>m(x) \\ e_l(x) & \text{if } p(x)\leq m(x) \end{cases} \tag{7}$$

CH2692-2/89/0000-1063 $1.00 © 1989 IEEE

From assumption i) $\{\mu_i - 1\}$ are restricted to be even, so in general the touch points are now points of inflection, Fig.2 and s(x) will change sign in the neighbourhood of each touch point $\{x_{ti}\}$. If the polynomial is manipulated such that p(x) does not cross u(x) or $l(x)$ at these points, then p(x) must possess an extra order of tangency to u(x) or $l(x)$, having then up to the μ_ith order tangency at $\{x_{ti}\}$ required for p(x) to be a solution. At this stage, max[s(x)]=0 in the neighbourhood of $\{x_{ti}\}$. Notice that from assumption ii) there must be at least one minima of e(x), denoted as x_{mi}, on $[x_{ti-1}, x_{ti+1}]$. Therefore if $x_{mi}=x_{ti}$ is achieved then p(x) is a solution, Fig.2. Some approximation scheme can be adopted to generate an adjustment $\{\Delta x_{ti}\}$

$$\{x_{t\,i}\} \leftarrow \{x_{t\,i} + \Delta x_{t\,i}\} \qquad i=1,2,3,\dots,M \qquad (8)$$

to reduce $\{e(x_{mi})\}$. Obviously $\{\Delta x_{ti}\}$ can be generated by a technique based on Newton's method which is found by solving a Jacobian system

$$\begin{bmatrix} g_1(x_{m1}) & \cdot\cdot & g_M(x_{m1}) \\ \cdot & & \cdot \\ \cdot & & \cdot \\ g_1(x_{mM}) & \cdot\cdot & g_M(x_{mM}) \end{bmatrix} \begin{bmatrix} \Delta x_{t1} \\ \cdot \\ \cdot \\ \Delta x_{tM} \end{bmatrix} = \begin{bmatrix} e(x_{m1}) \\ \cdot \\ \cdot \\ e(x_{mM}) \end{bmatrix} \qquad (9)$$

where $$g_i(x) = \partial p(x)/\partial x_{t\,i} \qquad (10)$$

The computational cost of setting up the Jacobian matrix J and solving the Newton system is usually large ($O(n^3)$). Efficient methods to obtain the derivatives $g_i(x_{mj})$ and to solve for the touch point increments $\{\Delta x_{ti}\}$ follow.

Lemma 1: *Define a N^{th} order polynomial q(x) subject to the following N+1 interpolation conditions*

$$q(x_{mi}) = e(x_{mi}) \qquad i = 1,2,\dots,M \qquad (11a)$$

$$q^{(r)}(x_{t\,i}) = 0 \qquad r = 0,1,\dots,\mu_i-2 \qquad i = 1,2,\dots,M \qquad (11b)$$

and

$$\delta_i(x) = \frac{(x-x_{t\,i})q(x)}{\mu_i\ e(x)} \qquad (11c)$$

then the Newton system (9) can be solved for the touch point increments $\{\Delta x_{ti}\}$ by

$$\Delta x_{t\,i} = \lim_{x \to x_{t\,i}} \delta_i(x) \qquad (12)$$

The proof is presented elsewhere [10]. As both numerator and denominator of (11c) tend to zero at x_{ti}, each touch point increment Δx_{ti} can only be calculated from the limiting values of the increment polynomial q(x) and error function e(x) in the proximity of the touch point, $x_{ti}+h$. Involving only repeated interpolation, the computational cost of the whole procedure is very small. The $O(n^3)$ step of solving the Newton system has been reduced to an $O(n^2)$ interpolation. Δx_{ti} can be approximately evaluated by $\delta_i(x)$ at a point close to x_{ti}, if this point is selected as $x=x_{mi}$, then a very simple adjustment to the touch point positions is revealed, {notice (11a)}

$$\Delta x_{t\,i} = \delta_i(x_{mi}) = \frac{(x_{mi}-x_{t\,i})\ q(x_{mi})}{\mu_i\ e(x_{mi})} = \frac{x_{mi}-x_{t\,i}}{\mu_i} \qquad (13)$$

In the special case of the curve fitting problem with all $\mu_i=1$, then (13) results in the well-known Remez method [2,11,12] which updates the variable vector by

$$\{x_{t\,i}\} \leftarrow \{x_{mi}\} \qquad i=1,2,\dots,M \qquad (14)$$

This indicates that the interpolation ordinates are simply exchanged with the locations of the extrema and re-interpolated (Fig. 2). For the case of $\mu_i > 1$, the simple exchange process of (14) is unsuitable. Instead the adjustment given by (13) is applicable,

$$\{x_{t\,i}\} \leftarrow \{x_{t\,i} + (x_{mi} - x_{t\,i})/\mu_i\} \qquad i=1,2,\dots,M \qquad (15)$$

This can be seen as a generalisation of the Remez method of (14) in which instead of moving the ordinate all the way to the extremum it is moved by a fraction of the distance dependant on the order of the touch point. When the $\{x_{ti}\}$ are close to the solution, $\{x_{mi}\}$ is also close to $\{x_{ti}\}$, and the adjustment given by (13) becomes similar to that given by a Newton method. This confirms that the Remez method achieves the good convergency of Newton iteration on approach to the solution.

In most cases the boundary functions are only given by values and the derivatives are not available. Although the derivatives can be calculated by numerical differentiation, it is notoriously inaccurate for high orders. The polynomial obtained by a Newton interpolation may become totally unreliable at the nearby region of a high order touch points. A better conditioned method is to interpolate the polynomial by a *cluster* of first order touch points (points with $\mu_i'=1$). The error caused by this approximate method can be controlled and made much smaller than the allowed ripple (the separation of u(x) and $l(x)$).

A lowpass approximation function employing high order touch points in both passband and stopband is illustrated in Fig.3. In the passband a 5th order point at 0Hz is followed by 4th, 2nd and 6th order points spread over the passband, while the stopband contains a 6th order point creating a deep notch at the lower edge.

The responses of various 9th order lowpass filters with equiripple stopband attenuation fixed to 65dB, passband edge frequency 200Hz and stopband edge frequency 250Hz are shown in Fig.4. Elliptic and inverse Chebyshev are the two classical solutions to the filter problem. In the delay response, the elliptic function has a very large peak which is much reduced by the smoother inverse Chebyshev function. By creating high order touch points and rounding the passband response delay peaks between these extremes can be created with improved passband response.

ITERATIVE LADDER DESIGN

Orchard has proposed a very simple but efficient algorithm to design an RLC ladder from a given reflection function $\rho(\omega)$ [5]. The structure of the ladder is prescribed and only the component values remain to be determined. A set of real and imaginary parts, $\{\mathrm{Re}[\rho(\omega_{ti})], \mathrm{Im}[\rho(\omega_{ti})]\}$ are used to set up the objective function vector, F, for Newton type iteration and component values, $\{y_i\}$, form a vector of variables Y. The core of Orchard's algorithm is an elegant, well conditioned method to compute F and the Jacobian matrix of derivatives by chain matrix calculations.

In the case of certain classical approximations, where the points of maximum or minimum transmission ($\rho(j\omega)=0$ or $\rho(j\omega)=1$) are known, the explicit calculation of $\rho(j\omega)$ is not necessary for Orchard's algorithm. However in general, Orchard's method requires the formation of $\rho(j\omega)$ by Hurwitz factorisation of $|\rho|^2$, which is an ill-conditioned procedure. In the following an extension of Orchard's method is described which works with more general forms of $|\rho|^2$ but eliminates any root finding requirement.

The value of $\hat{\rho} = |\rho|^2$ and its derivatives at the touch points $\{x_{ti}\}$ can be chosen as the objective function for the Newton's sheme. The derivatives of $|\rho|^2$ with respect to the element values, $\{y_k\}$, are required for the construction of the Jacobian matrix and are given by (let $x_{ti}=-\omega_{ti}^2$)

$$\frac{\partial}{\partial y_k}\left(\frac{d^r|\rho(\omega_{t\,i})|^2}{dx^r}\right) = \left[\begin{array}{l} r = 0,\ \dots,\ \mu i \\ i = 1,2,\dots,M \end{array}\right.$$
$$2\frac{d^r}{d\omega^r}\left\{\mathrm{Re}\left[\bar{\rho}(j\omega_{t\,i})\frac{\partial\rho(j\omega_{t\,i})}{\partial y_k}\right]\right\} \qquad (16)$$

Notice that here $\bar{\rho}$ and $\partial\rho/\partial y_k$ are obtained from the approximate network with guessed component values, which can be generated by Orchard's algorithm and then the remaining part of (16) can be calculated by a numerical differentiation. Here it is also found efficient to use 'cluster' method mentioned earlier.

No additional programming effort or computation cost is incurred by the extended algorithm. Both the original and extended Orchard's algorithms have good convergency for lowpass functions. For difficult specifications a synthesis program can be used to provide initial guesses of component values and the iterative algorithm to refine the accuracy. The whole program is still relatively simple since complicated accuracy preservation measures [6,7] in the synthesis stage can be dropped.

Continuation methods [13] are particularly useful in this problem to maintain the convergency of Newton iteration. It is important that the values of the components obtained are realiseable i.e. positive and not too large or small. Continuation methods ensure that if the iteration is started with a set reasonable initial values, then it usually terminates with a set of reasonable solution values.

In the case of very high order design (above 50) where ordinary synthesis programs would meet severe numerical difficulties a special approach may be taken. Observation of standard filter tables [14] reveals that there is some pattern of progression in element values of filters having similar ripple specifications for different orders. This raises the possibility of predicting the element values for higher order ladders from lower order ones. The component values for high order ladders are predicted by a third order extrapolation technique. Because the structures of even and odd order ladders differ topologically, it is better to increment the order by 2 at each step to retain the even or odd property. Filters up to 100th order have been designed in this way without loss of accuracy.

Ladders designed to be simulated in active RC or switched capacitor (SC) technology encounter many nonideal factors such as finite opamp gain-bandwith product or parasitic capacitance or switch resistance effects which influence the circuit behaviour [15]. These result in a distortion of the desired frequency response. Such non-ideal effects can be compensated for by reapproximation and redesign of the prototype ladder. The whole procedure is a straight-forward feed-back loop which has the advantage of simplicity and easy control. The amount of computation is based on the original specification order, not on the number of components in the filter circuit, as in a full optimisation scheme. Since there are typically many more components than filter order iterative re-design is very efficient.

Interestingly, a special difficulty arises when a ladder, designed with maximum power transmission at the touch points, must be predistorted. According to Orchard's observation $d|\rho|^2/dy_i$ is zero at these frequencies making the ladder especially insensitive to changes of the element values made by the iterative algorithm [16]. A simple solution is to adjust the terminating resistor a little to destroy the maximum power transmission property and encourage convergency.

An example of 20th order ladder design is illustrated in Fig.5, showing an upward slope in the passband and unequal stopband attenuations, Fig.6. The filter has been purposely designed to have two negative capacitors used for efficient SC simulation [10]. The ripples near the passband edge have been reduced to improve sensitivity. The component values obtained are to an accuracy of 10^{-7}, limited only by the numerical differentiation step for Eq. (16).

CONCLUSIONS

A Newton type approximation algorithm has been presented and shown to be a generalisation of the well-known Remez method. The magnitude response can be extended from the classical flat form to allow a continuous compromise for the consideration of phase. This also provides a means of compensation and optimisation to improve circuit performance for later stages. An extended version of Orchard's algorithm is then introduced for prototype RLC ladder design. By avoiding polynomial factorisation and insertion-loss synthesis high accuracy can be obtained whilst keeping the program simple.

ACKNOWLEDGEMENT

The support of GEC Research and the Science Research Council is gratefully acknowledged.

REFERENCES

[1] H. G. Martinez and T. W. Parks, "Design of recursive digital filters with optimum magnitude and attenuation poles on the unit circle", IEEE Transactions on Acoustics, Speech and Signal Processing, vol. ASSP-26, no. 2, pp. 150-156, April 1978.

[2] E. Ya. Remez, "General computation methods of Tchebysheff approximation", Kiev USSR, Atomic Energy Translation 4491, pp. 1-85, 1957.

[3] A.S. Sedra and P.O. Brackett," Filter theory and design: active and passive," Matrix, Portland, Oregon, 1978.

[4] R. Gregorian, G.C. Temes, "Design techniques for digital and analog all-pass circuits", IEEE Trans. Circuits and Systems, vol. CAS-25, no. 12, pp. 981-988, Dec. 1978.

[5] H.J.Orchard, "Filter design by iterated analysis", IEEE Trans. Circuits and Syst.,CAS-32, No. 11, pp. 1089-1096, Nov. 1985.

[6] H. J. Orchard and G. C. Temes, "Filter design using transformed variables," IEEE Trans., Circuit Theory, vol. C7-15, pp. 385-408, 1968.

[7] R. Saal and E. Ulbrich, "On the design of filters by synthesis", IRE Trans. Circuit Theory, vol. CT-5, pp.284-317, Dec. 1958.

[8] R. K. Henderson, Li Ping and J.I.Sewell, "PANDDA : A program for advanced network design : digital and analogue", Digest of IEE Saraga Colloquium on Electronic Filters, pp. 4/1-4/8, London 1988.

[9] F. B. Hildebrand, "Introduction to numerical analysis", McGraw-Hill, New York, 1956.

[10] Li Ping, R. K. Henderson and J. I. Sewell, "Filter design algorithms and novel structues", submitted for publication.

[11] M. T. McCallig and B. J. Leon, "Constrained ripple design of FIR digital filters", Trans. on Circuits and Systems, vol. CAS-25, no. 11, pp. 893-901, Nov. 1978.

[12] J. H. McClellan and T. W. Parks, "A computer program for designing optimum FIR linear-phase digital filters", IEEE Trans. on Audio and Electroacoustics, vol. AU-21, no. 6, pp. 506-524, Dec. 1973.

[13] S. D. Conte and C. De Boor, "Elementary numerical analysis : an algorithmic approach", McGraw-Hill, New York, 1980.

[14] R. Saal, "Handbook of filter design", Telefunken, W. Berlin, 1979.

[15] M. S. Ghausi, K. R. Laker, "Modern filter design : active RC and switched- capacitor", Prentice-Hall, Englewood Cliffs, New Jersey, 1981.

[16] H.J. Orchard,"Inductorless filters," Electron. Lett., vol. 2, pp. 224-225, June 1966.

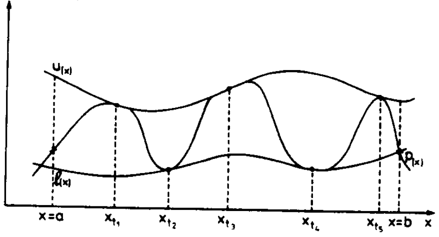

Fig.1 Minimax curve fitting with high order touch points

(217) 333-4789 Further information can be obtained from Dr. W. Kenneth Jenkins.

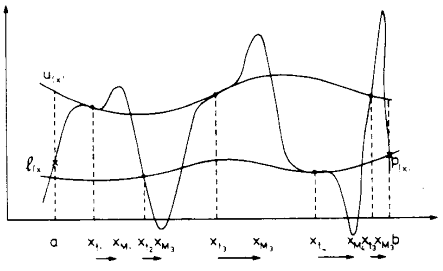

Fig.2 Bilateral approximation

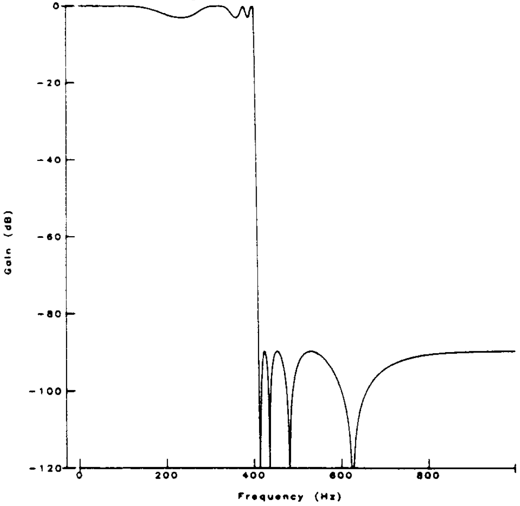

Fig.3 Rational 16th order approximation with high order touch points

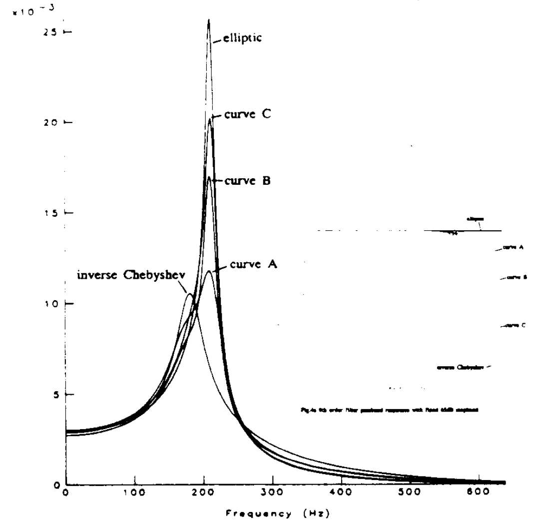

Fig.4a [illegible] order filter passband response with fixed [illegible] stopband

Fig.4b Variation of delay response

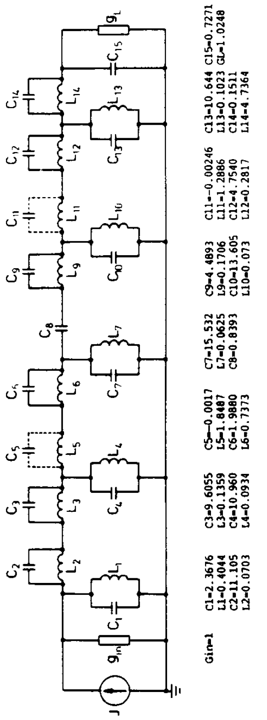

Fig.5 20th order ladder prototype

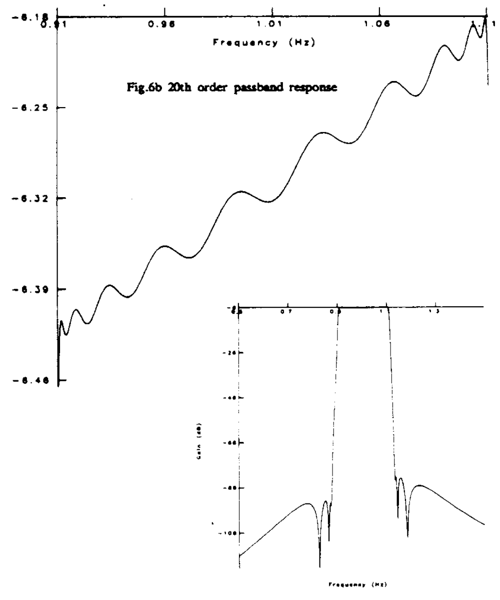

Fig.6b 20th order passband response

Fig.6a 20th order filter response