Iterative Reweighted Least-Squares Design of FIR Filters

C. Sidney Burrus, *Fellow, IEEE*, J. A. Barreto, and Ivan W. Selesnick

***Abstract*—This paper develops a new iterative reweighted least squares algorithm for the design of optimal L_p approximation FIR filters. The algorithm combines a variable p technique with a Newton's method to give excellent robust initial convergence and quadratic final convergence. Details of the convergence properties when applied to the L_p optimization problem are given. The primary purpose of L_p approximation for filter design is to allow design with different error criteria in pass and stopband and to design constrained L_2 approximation filters. The new method can also be applied to the complex Chebyshev approximation problem and to the design of 2-D FIR filters.**

I. INTRODUCTION

THE least-squared error and the minimum Chebyshev error criteria are the two most commonly used linear-phase FIR filter design methods [1]. There are many situations where better total performance would be obtained with a mixture of these two error measures or some compromise design that would give a trade-off between the two. We show how to design a filter with an L_2 approximation in the passband and a Chebyshev approximation in the stopband. We also show that by formulating the L_p problem we can solve the constrained L_2 approximation problem [2].

This paper first explores the minimization of the pth power of the error as a general approximation method and then shows how this allows L_2 and L_∞ approximations to be used simultaneous in different frequency bands of one filter and how the method can be used to impose constraints on the frequency response. There are no analytical methods to find this approximation, therefore, an iterative method is used over samples of the error in the frequency domain. The methods developed here [3], [4] are based on what is called an *iterative reweighted least-squared* (IRLS) error algorithm [5]–[7] and they can solve certain FIR filter design problems that neither the Remez exchange algorithm nor analytical L_2 methods can.

The idea of using an IRLS algorithm to achieve a Chebyshev or L_∞ approximation was first developed by Lawson [8] and extended to L_p by Rice and Usow [9], [10]. The basic IRLS method for L_p was given by Karlovitz [11] extended by Chalmers, *et al.* [12], Bani and Chalmers [13], and Watson [14]. Independently, Fletcher, Grant, and Hebden [15] developed a similar form of IRLS but based on Newton's method and Kahng [16] did likewise as an extension of Lawson's algorithm. Others analyzed and extended this work [17], [18], [7], [14]. Special analysis has been made for $1 \leq p < 2$ by [5], [6], [18]–[22] and for $p = \infty$ by [15], [13], [6], [23]–[25]. Relations to the Remez exchange algorithm [26], [27] were suggested by [13], to homotopy [28] by [29], and to Karmarkar's linear programming algorithm [30] by [6], [31]. Applications of Lawson's algorithm to complex Chebyshev approximation in FIR filter design have been made in [32]–[35] and to 2-D filter design in [36]. Reference [37] indicates further results may be forthcoming. Application to array design can be found in [38] and to statistics in [7].

This paper unifies and extends the IRLS techniques and applies them to the design of FIR digital filters. It develops a framework that relates all of the above referenced work and shows them to be variations of a basic IRLS method modified so as to control convergence. In particular, we generalize the work of Rice and Usow on Lawson's algorithm and explain why its asymptotic convergence is slow.

The main contribution of this paper is a new robust IRLS method [3], [4] that combines an improved convergence acceleration scheme and a Newton based method. This gives a very efficient and versatile filter design algorithm that performs significantly better than the Rice-Usow-Lawson algorithm or any of the other IRLS schemes. Both the initial and asymptotic convergence behavior of the new algorithm is examined and the reason for occasional slow convergence of this and all other IRLS methods is discovered.

We then show that the new IRLS method allows the use of p as a function of frequency to achieve different error criteria in the pass and stopbands of a filter. Therefore, this algorithm can be applied to solve the constrained L_p approximation problem. Initial results of applications to the complex and 2-D filter design problem are presented.

Although the traditional IRLS methods were sometimes slower than competing approaches, the results of this paper and the availability of fast modern desktop computers make them practical now and allow exploitation of their greater flexibility and generality.

II. FREQUENCY RESPONSE OF FIR FILTERS

The frequency response of a discrete-time filter is the discrete-time Fourier transform (DTFT) of the impulse response $h(n)$. This is a periodic function of ω with period

Manuscript received December 4, 1992; revised February 24, 1994. This work was supported by Bell Northern Research, the NSF, DARPA, Fundayacucho/Laspau, and by NDSEG fellowship N0014-89J-3204. The associate editor coordinating the review of this paper and approving it for publication was Dr. Robert A. Gobel.

The authors are with the Electrical and Computer Engineering Department and Computational Mathematics Laboratory, Rice University, Houston, TX 77251-1892 USA.

IEEE Log Number 9404779.

1053-587X/94$04.00 © 1994 IEEE

2π given by

$$H(\omega) = \sum_{n=0}^{L-1} h(n)\, e^{-j\omega n}. \tag{1}$$

If the phase response is constrained to be linear with the smallest constant of linearity, this is uniquely expressed in terms of the amplitude and phase response by

$$H(\omega) = A(\omega)\, e^{-jM\omega} \tag{2}$$

where

$$A(\omega) = \sum_{n=0}^{L-1} h(n) \cos(\omega(M-n)) \tag{3}$$

is a real-valued function of ω and the phase constant of linearity

$$M = \frac{L-1}{2} \tag{4}$$

is one half the order of the filter. M is also the constant group delay of the filter. Linear phase requires a symmetric impulse response such that

$$h(n) = h(L-1-n). \tag{5}$$

Using this symmetry allows summing only over half as many terms as in (3) but requires separate formulas for odd and even lengths. For L odd, the amplitude response in (3) becomes

$$A(\omega) = \sum_{n=0}^{M-1} 2\,h(n) \cos(\omega(M-n)) + h(M) \tag{6}$$

or with a change of variables

$$A(\omega) = \sum_{n=1}^{M} 2\,h(M-n) \cos(\omega n) + h(M). \tag{7}$$

These formulas can be made simpler by defining new coefficients so that (6) becomes

$$A(\omega) = \sum_{n=0}^{M} a(n) \cos(\omega(M-n)) \tag{8}$$

where

$$a(n) = \begin{cases} 2h(n) & \text{for } 0 \le n \le M-1 \\ h(M) & \text{for } n = M \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Similar expressions can be derived for even lengths and for odd symmetric filters [1].

To calculate the frequency or amplitude response numerically, one must consider samples of the continuous frequency response function above. LF samples of the amplitude response $A(\omega)$ in (8) are calculated from

$$A(\omega_k) = \sum_{n=0}^{M} a(n) \cos(\omega_k(M-n)) \tag{10}$$

for $k = 0, 1, 2, \cdots, LF-1$. This can be written with matrix notation as

$$A = C\,a \tag{11}$$

where A is an LF by 1 vector of the samples of the real valued amplitude frequency response, C is the LF by $(M+1)$ real matrix of cosines from (8), and a is the $(M+1)$ by 1 vector of filter coefficients related to the impulse response by (9). A similar set of equations can be written for an even length or for odd symmetric filters [1].

III. Minimum Squared Error Approximations

Various approximation methods can be developed by considering different definitions of norm or error measure. Commonly used definitions are L_1, L_2, and Chebyshev or L_∞. Using the L_2 norm, gives the scalar error to minimize

$$\varepsilon = \sum_{k=0}^{LF-1} |A(\omega_k) - A_d(\omega_k)|^2 \tag{12}$$

or in matrix notation using (11), the error or residual vector is defined by

$$\epsilon = C\,a - A_d \tag{13}$$

giving the scalar error of (12) as

$$\varepsilon = \epsilon^T \epsilon. \tag{14}$$

This can be minimized by solution of the normal equations [39], [40], [30]

$$C^T\,C\,a = C^T\,A_d. \tag{15}$$

The weighted squared error defined by

$$\varepsilon = \sum_{k=0}^{LF-1} w_k^2\,|A(\omega_k) - A_d(\omega_k)|^2 \tag{16}$$

or, in matrix notation using (13) and (14) causes (16) to become

$$\varepsilon = \epsilon^T\,W^T\,W\,\epsilon \tag{17}$$

which can be minimized by solving

$$W\,C\,a = W\,A_d \tag{18}$$

with the normal equations

$$C^T\,W^T\,W\,C\,a = C^T\,W^T\,W\,A_d \tag{19}$$

where W is an LF by LF diagonal matrix with the weights w_k from (16) along the diagonal. A more general formulation of the approximation simply requires W^TW to be positive definite. Some authors define the weighted error in (16) using w_k rather than w_k^2. We use the latter to be consistent with the least-squared error algorithms in Matlab [41].

Solving (19) is a direct method of designing an FIR filter using a weighted least-squared error approximation. To minimize the sum of the squared error and get approximately the same result as minimizing the integral of the squared error, one must choose LF to be 3 to 10 or more times the length L of the filter being designed.

IV. Iterative Algorithms to Minimize the L_p Error

There is no simple direct method for finding the optimal approximation for any error power other than two. However, if the weighting coefficients w_k as elements of W in (19) could be set equal to the elements in $|A-A_d|$, minimizing (16) would minimize the fourth power of $|A - A_d|$. This cannot be done in one step because we need the solution to find the weights! We can, however, pose an iterative algorithm which will first solve the problem in (15) with no weights, then calculate the error vector ϵ from (13) which will then be used to calculate the weights in (19). At each stage of the iteration, the weights are updated from the previous error and the problem solved again. This process of successive approximations is called the *iterative reweighted least-squared* error algorithm (IRLS).

The basic IRLS equations can also be derived by simply taking the gradient of the p-error with respect to the filter coefficients h or a and setting it equal to zero [15], [16]. These equations form the basis for the iterative algorithm.

If the algorithm is a contraction mapping [42], the successive approximations will converge and the limit is the solution of the minimum L_4 approximation problem. If a general problem can be posed [43]–[45] as the solution of an equation in the form

$$x = f(x) \tag{20}$$

a successive approximation algorithm can be proposed which iteratively calculates x using

$$x_{m+1} = f(x_m) \tag{21}$$

starting with some x_0. The function $f(\cdot)$ maps x_m into x_{m+1} and, if $\lim_{m\to\infty} x_m = x_0$ where $x_0 = f(x_0)$, x_0 is the fixed point of the mapping and a solution to (20). The trick is to find a mapping that solves the desired problem, converges, and converges fast.

By setting the weights in (16) equal to

$$w(k) = |A(\omega_k) - A_d(\omega_k)|^{(p-2)/2} \tag{22}$$

the fixed point of a convergent algorithm minimizes

$$\varepsilon = \sum_{k=0}^{LF-1} |A(\omega_k) - A_d(\omega_k)|^p. \tag{23}$$

It has been shown [9] that weights always exist such that minimizing (16) also minimizes (23). The problem is to find those weights efficiently.

A. Basic Iterative Reweighted Least Squares

The basic IRLS algorithm is started by initializing the weight matrix defined in (16) and (17) for unit weights with $W_0 = I$. Using these weights to start, the mth iteration solves (19) for the filter coefficients with

$$a_m = [C^T W_m^T W_m C]^{-1} C^T W_m^T W_m A_d. \tag{24}$$

This is a formal statement of the operation. In practice one should not invert a matrix, one should use a sophisticated numerical method [46] to solve the overdetermined equations in (11) The error or residual vector (13) for the mth iteration is found by

$$\epsilon_m = C\, a_m - A_d. \tag{25}$$

A new weighting vector is created from this error vector using (22) by

$$w_{m+1} = |\epsilon_m|^{(p-2)/2} \tag{26}$$

whose elements are the diagonal elements of the new weight matrix

$$W_{m+1} = \mathrm{diag}[w_{m+1}]. \tag{27}$$

Using this weight matrix, we solve for the next vector of filter coefficients by going back to (24) and this defines the basic iterative process of the IRLS algorithm.

It can easily be shown that the a that minimizes (23) is a fixed point of this iterative map. Unfortunately, applied directly, this basic IRLS algorithm does not converge and/or it has numerical problems for most practical cases [7]. There are three aspects that must be addressed. First, the IRLS algorithm must theoretically converge. Second, the solution of (24) must be numerically stable. Finally, even if the algorithm converges and is numerically stable, it must converge fast enough to be practical.

Both theory and experience indicate there are different convergence problems connected with several different ranges and values of p. In the range $2 \le p < 3$, virtually all methods converge [15], [7], [18]. In the range $3 \le p < \infty$, the algorithm diverges and the various methods discussed in this paper must be used. As p becomes large compared to 2, the weights carry a larger contribution to the total minimization than the underlying least-squared error minimization, the improvement at each iteration becomes smaller, and the likelihood of divergence becomes larger. For $p = \infty$ we can use to advantage the fact that the optimal approximation solution to (23) is unique but the weights in (16) that give that solution are not. In other words, different matrices W give the same solution to (24) but will have different convergence properties. This allows certain alteration to the weights to improve convergence without harming the optimality of the results [25]. In the range $1 < p < 2$, both convergence and numerical problems exist as, in contrast to $p > 2$, the IRLS iterations are undoing what the underlying least squares is doing. In particular, the weights near frequencies with small errors become very large. Indeed, if the error happens to be zero, the weight becomes infinite because of the negative exponent in (26). For $p = 1$ the solution to the optimization problem is not even unique. The various algorithms that are presented below are based on schemes to address these problems.

B. The Karlovitz Method

In order to achieve convergence, a second order update is used which only partially changes the filter coefficients a_m in (24) each iteration. This is done by first calculating the unweighted L_2 approximation filter coefficients using (15) as

$$a_0 = [C^T C]^{-1} C^T A_d. \tag{28}$$

The error or residual vector (13) for the mth iteration is found as (25) by

$$\epsilon_m = C\, a_m - A_d \tag{29}$$

and the new weighting vector is created from this error vector using (22) by

$$w_{m+1} = |\epsilon_m|^{(p-2)/2} \tag{30}$$

whose elements are the diagonal elements of the new weight matrix

$$W_{m+1} = \mathrm{diag}[w_{m+1}]. \tag{31}$$

This weight matrix is then used to calculate a temporary filter coefficient vector by

$$\hat{a}_{m+1} = [C^T W_{m+1}^T W_{m+1} C]^{-1} C^T W_{m+1}^T W_{m+1} A_d. \tag{32}$$

The vector of filter coefficients that is actually used is only partially updated using a form of adjustable step size in the following second order linearly weighted sum

$$a_{m+1} = \lambda \hat{a}_{m+1} + (1-\lambda) a_m. \tag{33}$$

Using this filter coefficient vector, we solve for the next error vector by going back to (29) and this defines Karlovitz's IRLS algorithm [11].

In this algorithm, λ is a convergence parameter that takes values $0 < \lambda \le 1$. Karlovitz showed that for the proper λ, the IRLS algorithm using (32) always converges to the globally optimal L_p approximation for p an even integer in the range $4 \le p < \infty$. At each iteration the L_p error has to be minimized over λ which requires a line search. In other words, the full Karlovitz method requires a multidimensional weighted least-squares minimization and a 1-D pth power error minimization at each iteration. Extensions of Karlovitz's work [14] show the 1-D minimization is not necessary but practice shows the number of required iterations increases considerably and robustness in lost.

Fletcher *et al.* [15] and later Kahng [16] independently derive the same second order iterative algorithm by applying Newton's method. That approach gives a formula for λ as a function of p and is discussed later in this paper. Although the number of iterations for convergence of the Karlovitz method is low, indeed, perhaps the lowest of all, the minimization of λ at each iteration causes the algorithm to be very slow in execution.

C. The Rice–Usow–Lawson Method

Lawson [8] developed an IRLS algorithm and gave a good analysis of it for solving the Chebyshev or L_∞ approximation problem. Rice and Usow [9], [10] presented an extended and generalized version of the approach to include the L_p problem and we call their approach the Rice–Usow–Lawson (RUL) algorithm. In this section, we show their algorithm can be derived in a more general form from the basic successive approximation formulation in Section IV-A. Unlike most other methods, this algorithm uses an update of the weights rather than the filter coefficients. It starts with the fixed point of the basic IRLS having a weight vector of

$$w_{m+1} = |\epsilon_m|^{(p-2)/2} \tag{34}$$

as was given in (26).

Rather than using the linear summation form of (32) and (33), a product form for partial updating the weights is posed as

$$w_{m+1} = w_m^{\alpha} |\epsilon_m|^{\beta}. \tag{35}$$

Equating (34) and (35) gives

$$|\epsilon_m|^{(p-2)/2} = |\epsilon_m|^{\alpha(p-2)/2} |\epsilon_m|^{\beta} \tag{36}$$

or

$$|\epsilon_m|^{p-2} = |\epsilon_m|^{\alpha(p-2)+2\beta} \tag{37}$$

which requires the relationship among α, β, and p to be

$$\alpha(p-2) + 2\beta = p - 2. \tag{38}$$

This means (34) will be a fixed point of the map (35) if α and η satisfy (38). Using (35) subject to (38) and for $\alpha = 2\beta$ in place of (26) in the basic IRLS gives Rice and Usow's extension of Lawson's algorithm, the RUL algorithm.

Since p is given by the problem specifications, (38) allows a single parameter to be used to affect convergence of the IRLS algorithm. There are several ways to define this parameter. One that would be a simple generalization of Rice's approach would define the parameter γ by requiring

$$\gamma = \frac{\alpha}{2\beta} \tag{39}$$

which from (38) would give

$$\alpha = \frac{\gamma(p-2)}{\gamma(p-2)+1} \tag{40}$$

and

$$\beta = \frac{\alpha}{2\gamma} = \frac{p-2}{2(\gamma(p-2)+1)}. \tag{41}$$

For various γ, this would result in

$\gamma = 0 \quad \Rightarrow w_{m+1} = |\epsilon_m|^{(p-2)/2}$
Unmodified (26) IRLS
$\gamma = 1/2 \Rightarrow w_{m+1} = (w_m|\epsilon_m|)^{(p-2)/p}$
Modified RUL algorithm
$\gamma = 1 \quad \Rightarrow w_{m+1} = (w_m^2|\epsilon_m|)^{(p-2)/2(p-1)}$
RUL algorithm
$\gamma = 2 \quad \Rightarrow w_{m+1} = (w_m^4\sqrt{|\epsilon_m|})^{(p-2)/2(4p-3)}$
Modified RUL algorithm
$\gamma = \infty \quad \Rightarrow w_{m+1} = w_m$
No iteration!

Experimentation shows best general convergence for $\gamma \approx 0.65$. Other parameterizations than (39) might prove interesting. The RUL algorithm gives fairly good initial convergence but the final convergence is slow for all values of γ.

Because Lawson's algorithm and the RUL generalization use a multiplicative updating of the weights rather than the additive update of a Newton's method discussed in the next section, they cannot achieve quadratic convergence. This explains the observed slow convergence of the basic Lawson's method and lack of success we had in using generalizations to improve its speed of convergence.

D. Newton's Methods

Both Karlovitz's and Lawson's methods use a second order updating of the weights to obtain convergence of the basic IRLS algorithm. Fletcher *et al.* [15] and Kahng [16] use a linear summation for the updating similar in form to (33) and apply it to the filter coefficients in the manner of Karlovitz rather than the weights as Lawson did. Indeed, using our development of Karlovitz's method in Section IV-B, we see that Kahng's method and Fletcher, Grant, and Hebden's method are simply a particular choice of λ as a function of p in Karlovitz's method. They derive

$$\lambda = \frac{1}{p-1} \tag{42}$$

by using Newton's method to minimize ε in (23) to give for (33)

$$a_m = (\hat{a}_m + (p-2)a_{m-1})/(p-1). \tag{43}$$

This defines Kahng's method, which, he says, always converges [16], [45]. He also notes that the summation methods in Section IV-B and IV-C do not have the possible restarting problem that Lawson's method theoretically does. Because Kahng's algorithm is a form of Newton's method, its asymptotic convergence is very good but the initial convergence is poor and very sensitive to starting values.

V. A New Robust IRLS Method

A modification and generalization of an acceleration method suggested independently by Ekblom [17] and by Kahng [16] is developed here and combined with the Newton's method of Fletcher, Grant, and Hebden and of Kahng to give a robust, fast, and accurate IRLS algorithm [3], [4]. It overcomes the poor initial performance of the Newton's methods and the poor final performance of the RUL algorithms.

Rather than starting the iterations of the IRLS algorithms with the actual desired value of p, after the initial L_2 approximation, the new algorithm starts with $p = K*2$ where K is a parameter between one and approximately two, chosen for the particular problem specifications. After the first iteration, the value of p is increased to $p = K^2*2$. It is increased by a factor of K at each iteration until it reaches the actual desired value. This keeps the value of p being approximated just ahead of the value achieved. This is similar to a homotopy where we vary the value of p from 2 to its final value. A small value of K gives very reliable convergence because the approximation is achieved at each iteration but requires a large number of iterations for p to reach its final value. A large value of K gives faster convergence for most filter specifications but fails for some. The rule that is used to choose p_m at the mth iteration is

$$p_m = \min(p, K\,p_{m-1}). \tag{44}$$

Each iteration of our new variable p method is implemented by the basic algorithm described as Karlovitz's method but using the Newton's method based value of λ from Fletcher or Kahng in (42). Both Ekblom and Kahng only used $K = 2$ which is too large in almost all cases.

We also tried the generalized acceleration scheme with the basic Karlovitz method and the RUL algorithm. Although it improved the initial performance of the Karlovitz method, the slowness of each iteration still made this method unattractive. Use with the RUL algorithm gave only a minor improvement of initial performance and no improvement of the poor final convergence.

Our new algorithm uses three distinct concepts:

- The basic IRLS which is a straight forward algorithm with linear convergence [7] when it converges.
- The second order or Newton's modification which increases the number of cases where initial convergence occurs and gives quadratic asymptotic convergence [15], [16].
- The controlled increasing of p from one iteration to the next is a modification which gives excellent initial convergence and allows adaptation for "difficult" cases.

The best total algorithm, therefore, combines the increasing of p given in (44) the updating the filter coefficients using (33), and the Newton's choice of λ in (42). By slowly increasing p, the error surface slowly changes from the parabolic shape of L_2 which Newton's method is based on, to the more complicated surface of L_p. The question is how fast to change and, from experience with many examples, we have learned that this depends on the filter design specifications.

A Matlab program that implements this basic IRLS algorithm is given in the appendix of this paper. It uses an updating of $A(\omega_k)$ in the frequency domain rather than of $a(n)$ in the time domain to allow modifications necessary for using different p in different bands as will be developed later in this paper.

A. Convergence Properties

The basic IRLS algorithm of Section IV-A converges for so few cases and is so slow that it is useless for filter design. Lawson's method converges so slowly as to be impractical. The second order Newton's method version of Section IV-D converge rapidly but still diverge for many practical specifications. It is the new IRLS algorithm described in the preceding section that is practical and is investigated in this section.

It was found that for most specifications, the new algorithm is robust with good initial convergence and excellent quadratic final convergence. The occasional case where the convergence is slow occurs when the final optimal L_p approximation has a different number of passband and/or stopband ripples than does the L_2 approximation or has a small or incomplete ripple [47]. This problem also exists using the Remez algorithm and can be addressed in our new approach by using an adjustable K parameter which adapts itself at each iteration in a manner similar to adjustable step size methods for the numerical solution of differential equations.

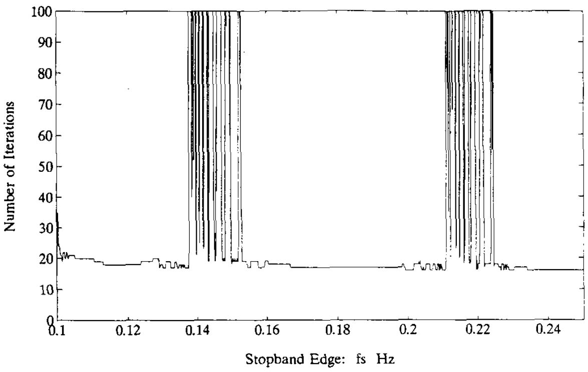

Fig. 1. Number of iterations for convergence.

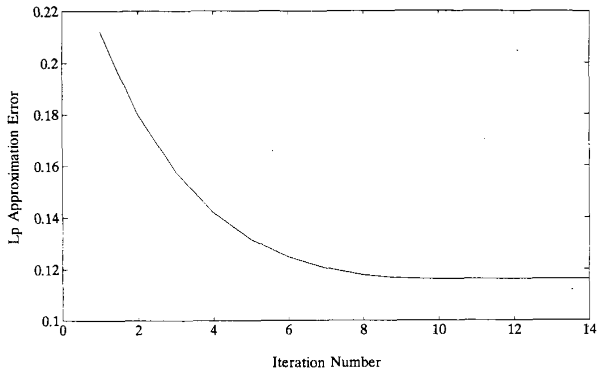

Fig. 2. Convergence characteristics for example 1.

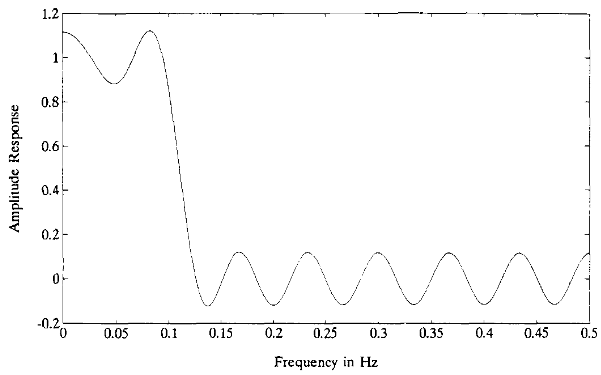

Fig. 3. Frequency response of filter in example 1.

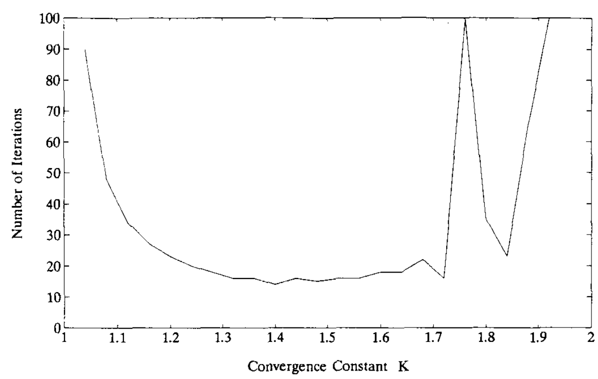

Fig. 4. Number of iterations for convergence for $f_s = 0.12$.

The basic new IRLS algorithm was used to design a large number of FIR filters in order to understand its characteristics better. The Matlab program listed in the appendix was used for these examples. In most cases we define convergence by approximately 14 decimal places of accuracy in the filter coefficients.

In order to understand why the occasional slow convergence occurred, we designed 1500 filters and looked at the number of iterations for convergence for each one while varying the stopband edge from 0.1001 to 0.25 Hz in steps of 0.0001. Fig. 1 illustrates the results. For a filter length $L = 31$, using $LF = 301$ frequency samples, a passband edge of $f_p = 0.1$ Hz, a sampling frequency of $f_0 = 1$ Hz, an error power of $p = 60$, and an acceleration multiplier of $K = 1.3$, we found the number of iterations required was around 18 except in two regions near $f_s = 0.14$ and $f_s = 0.215$ Hz. For these stopband edges, it could take well over 200 iterations. In all cases, the algorithm eventually converged and an optimal solution was found. It was in examining the frequency responses of the filters designed in the various regions that we discovered that it is the changing number of ripples that causes the slow convergence.

Using the same process of examining the number of iterations for convergence as a function of the stopband edge, we discovered that for $K = 1.5$ convergence required around 15 iterations for most cases, but slow convergence occurred for more values of f_s. For $K = 1.1$, around 40 iterations were generally required, slow convergence occurred for fewer values of f_s, and the number of iterations for the slow cases was considerably reduced. For $K = 1.05$ the algorithm converged in 73 iterations for all stopband edges. These experiments clearly show how the choice of K affects convergence of our new algorithm.

First Example: We now consider the filter specification above with a stopband edge that gives fast convergence. Let $f_s = 0.12$ Hz and $K = 1.5$. The first row of Table I shows the number of iterations required for convergence for several values of p. Fig. 2 illustrates the nature of the convergence for $p = 60$. It requires 14 iterations for 14 decimal place convergence. Fig. 3 shows the frequency response of the optimal filter designed. Its response is very near equal ripple using $p = 60$. Fig. 4 illustrates how the speed of convergence depends on the acceleration constant K in (44). For values of K below approximately 1.3 the convergence is primarily controlled by the acceleration process with each iteration of the Newton's method essentially achieving its approximation for the value of p_k of that iteration. For values above 1.4 or so, the convergence is controlled by the IRLS Newton's method which has trouble if p_k gets too far ahead of what each iteration can achieve.

Second Example: We now consider a case where the convergence slows down because one of the ripples is not full size. It uses the same specifications as above but with a stopband

TABLE I
ITERATIONS FOR CONVERGENCE WITH $K = 1.3$

p =	6	10	20	60	100	300
Iterations for example 1	7	8	10	14	17	19
Iterations for example 2	8	11	13	18	20	24
Iterations for example 3	8	10	62	188	298	1326

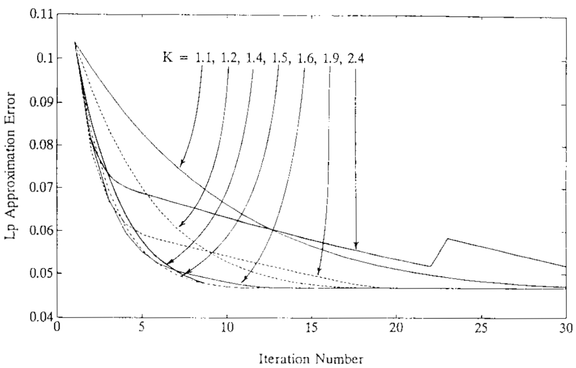

Fig. 5. Convergence for length-31 filter with $f_p = 0.1$, $f_s = 0.134$, and $p = 60$.

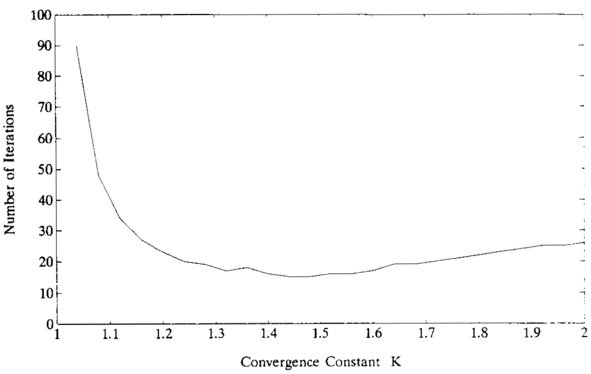

Fig. 6. Number of iterations for convergence, $f_s = 0.134$..

edge of $f_s = 0.134$ and the convergence is illustrated in the second row of Table I. Observing the shape of the plot of approximation error versus iteration for this example illustrates several of its properties. Fig. 5 shows this plot for several values of K. As was true for example 1, for values of K less than approximately 1.5, the convergence is controlled by the acceleration process which allows the quadratic convergence of the Newton's method to be seen. For values of K larger than 1.6 or so, the p_k gets so far ahead of the Newton's method, that the linear convergence of the basic IRLS algorithm is dominant. The frequency response looks very similar to the one for example 1 but the dependency on K is smoother and is shown in Fig. 6.

Third Example: We now use specifications that cause very slow convergence. Here we use $f_s = 0.14$ Hz. Convergence is illustrated in the third row of Table I for various error powers p. In order to see what causes the slow convergence, we use $K = 1.3$ and $p = 18$ to give the plot illustrated in Fig. 7. The frequency response for the filter designed is shown in Fig. 8.

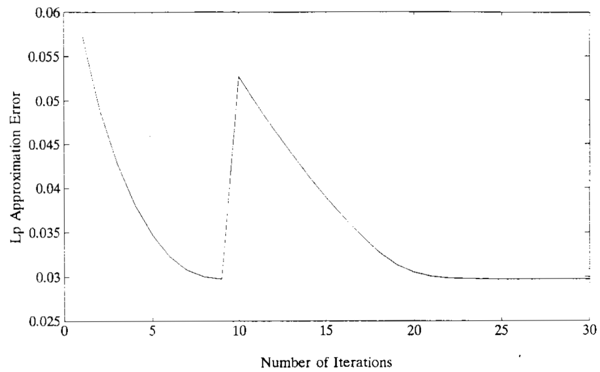

Fig. 7. Convergence characteristics for example 3.

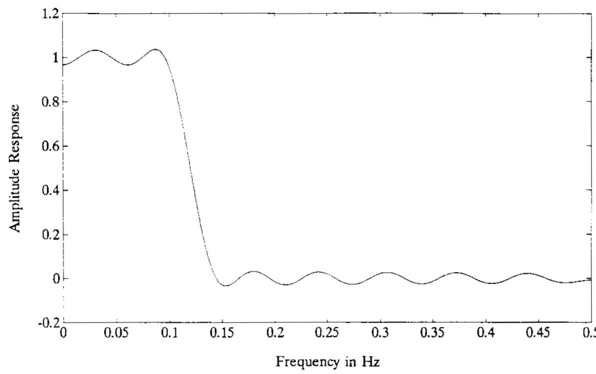

Fig. 8. Frequency response of filter in example 3.

The filter designed at iteration 9 has an L_p error almost as low as the optimal filter but its frequency response is different near $f = 0.5$ Hz. It was the correction of this difference that caused the large jump in the error versus iteration plot. A smaller value of K could have prevented this jump but would have slowed convergence elsewhere. That observation suggests an adaptive control of K might allow fast general convergence and prevent the "jumps" that cause the erratic number of required iterations. The very large number of iterations seen in regions of Fig. 1 and in the third row of Table I are the result of many large jumps in the error plots.

In the third example, for $p < 9$ the number of ripples in the stopband for the L_2 approximation did not change during the iterations and the convergence was relatively insensitive to the value of K. For $p > 10$ the number of stopband ripples changed and the convergence became much more complicated, requiring close control of K. Indeed, it was very informative to plot the frequency response of the filter at each iteration. For the first example and for this example with $p < 8$, the frequency response smoothly changed to its final shape. However, for this example with $p > 10$, the shape of the frequency response went through complicated gyrations between initial and final iteration. This explained our observance of occasional slow convergence of the new algorithm and the difficulty of comparing speed of convergence unless the specifications were exactly the same.

TABLE II
ITERATIONS FOR CONVERGENCE

Length =	31	51	71	101	201	301	401
Iterations	22	22	22	22	22	24	24

To examine the filter lengths that the new algorithm can handle, we designed filters of length up to 401 with the number of frequency samples set at approximately $3L$ for each case. The number of iterations required for complete convergence was almost independent of length. This is shown in Table II.

This can be a bit misleading because each iteration took considerably more time for the longer filter lengths. Still longer filters might be possible on a larger computer. These examples were run on a 486 PC and a Sun SPARC 1+.

The other IRLS algorithms discussed earlier in this paper were also run on a wide variety of examples. They all fell short in comparison with our new algorithm either in poor initial performance where many diverged, or in poor final performance requiring a very large number of iterations. For example the RUL algorithm might get within 1% of convergence in five iterations and then require 300 iterations to obtain the complete 14 decimal place convergence that our new method obtained in 14 iterations.

Special problems occur for $1 \leq p < 2$ [7]. This occurs when there is a large range on the weights caused by errors near zero. The method can be made to converge, but it requires some special modifications and we are still working on this problem.

B. Different Error Criteria in Different Bands

Probably the most important use of the L_p approximation problem posed here is its use for designing filters with different error criteria in different frequency bands. This is possible because the IRLS algorithm allows an error power that is a function of frequency $p(\omega)$ which can allow an L_2 measure in the passband and a Chebyshev error measure in the stopband or any other form. This is important if an L_2 approximation is needed in the passband because Parseval's theorem shows that the time domain properties of the filtered signal will be well preserved but, because of unknown properties of the noise or interference, the stopband attenuation must be less than some specified valued.

The new algorithm described in Section V was modified so that the iterative updating is done to $A(\omega)$ rather than to $a(n)$. Because the Fourier transform is linear, the updating of (33) can also be achieved by

$$A_{m+1}(\omega) = \lambda \hat{A}_{m+1}(\omega) + (1-\lambda)A_m(\omega). \tag{45}$$

The Matlab program listed in the appendix uses this form. This type of updating in the frequency domain allows different p to be used in different bands of $A(\omega)$ and different update parameters λ to be used in the appropriate bands. In addition, it allows a different constant K weighting to be used for the different bands. The error for this problem is changed from

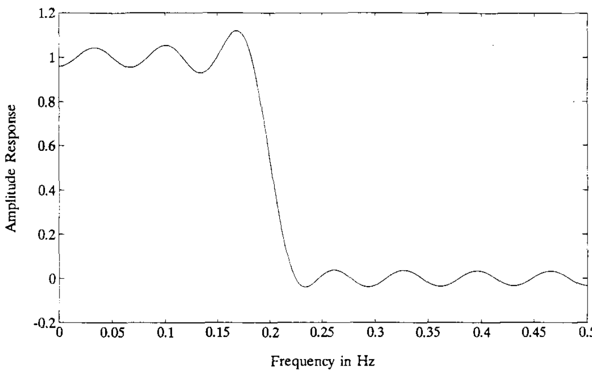

Fig. 9. Frequency response for L_2 passband and L_{60} stopband approximation.

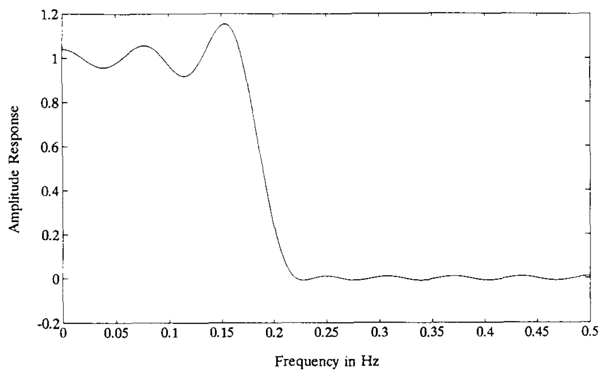

Fig. 10. Frequency response for L_2 passband and weighted L_{60} stopband.

(12) to be

$$\varepsilon = \sum_{k=0}^{k_0} |A(\omega_k) - A_d(\omega_k)|^2 + K \sum_{k=k_0+1}^{LF-1} |A(\omega_k) - A_d(\omega_k)|^p. \tag{46}$$

Fig. 9 shows the frequency response of a filter designed with a passband $p = 2$, a stopband $p = 60$, and a stopband weight of $K = 1$. Fig. 10 gives the frequency response for the same specifications but with $K = 100$. In many cases p above 10 or 20 gives close to Chebyshev results in the stopband. The algorithm will allow several bands and several values of p and K [47]. The convergence properties are not as clean as for the case with a single p and are still being investigated.

C. The Constrained L_p Approximation

In some design situations, neither a pure L_2 nor a L_∞ or Chebyshev approximation is appropriate. If one evaluates both the squared error and the Chebyshev error of a particular filter, it is easily seen that for an optimal least-squares solution, a considerable reduction of the Chebyshev error can be obtained by allowing a small increase in the squared error. For the optimal Chebyshev solution the opposite is true. A considerable reduction of the squared error can be obtained by allowing a small increase in the Chebyshev error. This suggests a better filter might be obtained by some combination of L_2 and L_∞

TABLE III
ITERATIVE REWEIGHTED LEAST SQUARES MATLAB PROGRAM

```
%  a = irls(L,fs,fp,LF,p,K) designs a least p-power error approx
%  lowpass length-L FIR filter using an iterative method which updates
%  the L_2 weights using Kahng's method.  An convergence factor K
%  controls the rate of updating each iteration.  The approx. is over
%  LF freq points from  f = 0 to .5 Hz.  Plots L_p vs iter.  csb 11/5/93
%  Odd length only.
%
R = (L+1)/2;                                           % Odd length
M = (L-1)/2; E2 = []; EC = []; EP = [];                % Group delay M
Np = round(LF*fp/(fp+.5-fs)); Ns = LF-Np;              % Samples bands
dp = fp/Np; ds = (.5-fs)/Ns;                           % Sample spacing
Ad = [ones(Np,1); zeros(Ns,1)];                        % Ideal amplitude
f  = [((0:Np-1)*dp + dp/2), ((0:Ns-1)*ds + fs + ds/2)]';  % Freq. vector
C  = cos(2*pi*(f*[0:R-1])); WC = C;                    % Fourier T. matrix
a  = C\Ad;                                             % Initial L_2 design
A  = C*a;                                              % Actual amplitude
pk = 2;                                                % Initial error power
for k = 1:150                                          % Iterate
   pk = min([p, K*pk]);                                % Convergence control
   e  = A - Ad;                                        % Error vector
   ep = ((sum(abs(e).^p))/LF)^(1/p);                   % Lp error
   e2 = sqrt((e'*e)/LF); ec = max(abs(e));             % L2 and Cheby errors
   w  = abs(e.^((pk-2)/2));                            % Error weights
   w  = w/sum(w);                                      % Normalized weights
   for m = 1:R
      WC(:,m) = w.*C(:,m);                             % Apply weights
   end;
   a  = WC\(w.*Ad);                                    % weighted L_2 sol.
   q  = 1/(pk-1);                                      % Kahng's parameter
   A  = q*C*a + (1-q)*A;                               % Kahng's partial update
   E2 = [E2 e2]; EC = [EC ec]; EP = [EP ep];           % Error vectors
   if p>100, plot(EC); else, plot(EP); end             % Plot E vs iteration
end
```

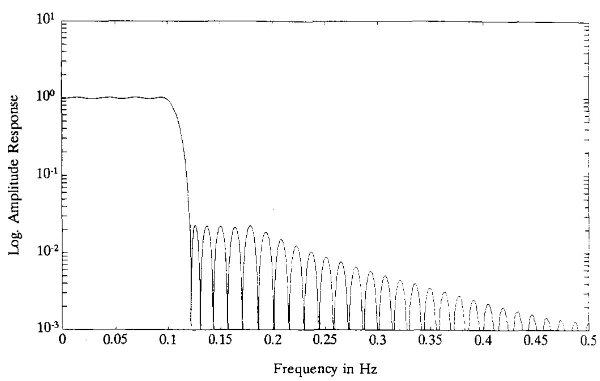

Fig. 11. Frequency response for constrained L_2 stopband approximation.

approximation. This problem is stated and addressed by Adams [2] and by Lang [48], [49].

We have applied the IRLS method to the constrained least-squares problem by adding an error based weighting function to unity in the stopband only in the frequency range where the response in the previous iteration exceeds the constraint. The frequency response of an example is shown in Fig. 11. The IRLS approach to this problem is currently being evaluated and compared to the approach used by Adams. The initial results are encouraging.

D. Application to the Complex Approximation and the 2-D Filter Design Problem

Although described above in terms of a 1-D linear phase FIR filter, the method can just as easily be applied to the complex approximation problem and to the multidimensional filter design problem. We have obtained encouraging initial results from applications of our new IRLS algorithm to the optimal design of FIR filters with a nonlinear phase response. By using a large p we are able to design essentially Chebyshev filters where the Remez algorithm is difficult to apply reliably.

Our new IRLS design algorithm was applied to the two examples considered by Chen and Parks [50] and by Schulist [51], [52] and Preuss [53], [54]. One is a lowpass filter and the other a bandpass filter, both approximating a constant group delay over their passbands. Examination of magnitude frequency response plots, imaginary versus real part plots, and group delay frequency response plots for the filters designed by the IRLS method showed close agreement with published results [55]. The use of an L_p approximation may give more desirable results than a true Chebyshev approximation. Our results on the complex approximation problem are preliminary and we are doing further investigations on convergence properties of the algorithm and on the characteristics of L_p approximations in this context.

Application of the new IRLS method to the design of 2-D FIR filters has also given encouraging results. Here again, it is difficult to apply the Remez exchange algorithm directly to the multidimensional approximation problem. Application of the IRLS to this problem is currently being investigated.

We designed 5×5, 7×7, 9×9, 41×41, and 71×71 filters to specifications used in [36], [56]–[58]. Our preliminary observations from these examples indicate the new IRLS method is faster and/or gives lower Chebyshev errors than any of the other methods [59]. Values of K in the 1.1

to 1.2 range were required for convergence. As for the complex approximation problem, further research is being done on convergence properties of the algorithm and on the characteristics of L_p approximations in this context.

VI. Conclusion

We have proposed applying the iterative reweighted least-squared error approach to the FIR digital filter design problem. We have shown how a large number of existing methods can be cast as variations on one basic successive approximation algorithm called iterative reweighted least squares. From this formulation we were able to understand the convergence characteristics of all of them and see why Lawson's method has experimentally been found to have slow convergence.

We have created a new IRLS algorithm by combining an improved acceleration scheme with Fletcher's and Kahng's Newton type methods to give a very good design method with good initial and final convergence properties. It is a significant improvement over the Rice-Usow-Lawson method.

The main contribution of the paper was showing how to use these algorithms with different p in different frequency bands to give a filter with different pass and stopband characteristics, how to solve the constrained L_p problem, and how the approach is used in complex approximation and in 2-D filter design.

Appendix

The Appendix consists of Table III, in which the iterative reweighted least-squares MATLAB program appears.

Acknowledgment

The authors thank H. Schüssler, T. Parks, R. Gopinath, J. Odegard, M. Lang, and J. Dennis for helpful comments on this work. One of the anonymous reviewers gave detailed comments that were helpful in rewriting the paper. The various algorithms were developed, analyzed, and evaluated using the software system (Matlab) [41].

References

[1] T. W. Parks and C. S. Burrus, *Digital Filter Design*. New York: Wiley, 1987.

[2] J. W. Adams, "FIR digital filters with least-squares stopbands subject to peak-gain constraints," *IEEE Trans. Circuits, Syst.*, vol. 39, pp. 376–388, 1991.

[3] C. S. Burrus and J. A. Barreto, "Least p-power error design of FIR filters," in *Proc. IEEE Int. Symp. Circuits, Syst. ISCAS-92* (San Diego, CA), May 1992, pp. 545–548.

[4] C. S. Burrus, J. A. Barreto, and I. W. Selesnick, "Reweighted least squares design of FIR filters," in *Paper Summaries IEEE Signal Processing Soc. Fifth DSP Workshop* (Utica, IL), Sept. 13–16 1992, p. 3.1.1.

[5] J. Schroederm, R. Yarlagadda, and J. Hershey, "L_p normed minimization with applications to linear predictive modeling for sinusoidal frequency estimation," *Signal Processing*, vol. 24, no. 2, pp. 193–216, Aug. 1991.

[6] S. A. Ruzinsky and E. T. Olsen,"L_1 and L_∞ minimization via a variant of Karmarkar's algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 245–253, Feb. 1989.

[7] R. H. Byrd and D. A. Pyne, "Convergence of the iteratively reweighted least squares algorithm for robust regression," Tech. Rep. 313, Dept. of Math. Sci., The Johns Hopkins Univ., Baltimore, MD, June 1979.

[8] C. L. Lawson, "Contribution to the theory of linear least maximum approximations," Ph.D. dissertation, Univ. of Calif. at Los Angeles, 1961.

[9] J. R. Rice and K. H. Usow, "The Lawson algorithm and extensions," *Math. Comput.*, vol. 22, pp. 118–127, 1968.

[10] J. R. Rice, *The Approximation of Functions, vol. 2*. Reading, MA: Addison-Wesley, 1964.

[11] L. A. Karlovitz, "Construction of nearest points in the L^p, p even, and L^∞ norms, I," *J. Approx. Theory*, vol. 3, pp. 123–127, 1970.

[12] B. L. Chalmers, A. G. Egger, and G. D. Taylor. "Convex L^p approximation," *J. Approx. Theory*, vol. 37, pp. 326–334, 1983.

[13] M.S. Bani and B. L. Chalmers, "Best approximation in L^∞ via iterative Hilbert space procedures," *J. Approx. Theory*, vol. 42, pp. 173–180, 1984.

[14] G. A. Watson, "Convex L^p approximation," *J. Approx. Theory*, vol. 55, no. 1, pp. 1–11, Oct. 1988.

[15] R. Fletcher, J. A. Crant, and M. D. Hebden, "The calculation of linear best L_p approximations," *Comput. J.*, vol. 14, pp. 276–279, 1971.

[16] S. W. Kahng, "Best L_p approximation," *Math. Comput.*, vol. 26, no. 118, pp. 505–508, Apr. 1972.

[17] H. Ekblom, "Calculation of linear best L_p-approximations," *BIT*, vol. 13, no. 3, pp. 292–300, 1973.

[18] G. Merle and H. Späth, "Computational experiences with discrete L_p-apprioximation," *Comput.*, vol. 12, pp. 315–321, 1974.

[19] K. H. Usow, "On L_1 approximation 1: Computation for continuous functions and continuous dependence," *SIAM J. Numer. Anal.*, vol. 4, no. 1, pp. 70–88, 1967.

[20] G. A. Watson, "An algorithm for linear L_1 approximation of continuous functions," *IMA J. Numer. Anal.*, vol. 1, pp. 157–167, 1981.

[21] Y. Li, "A globally convergent method for L_p problems," Tech. Rep. TR 91-1212, Comput. Sci. Dept., Cornell Univ., Ithaca, NY, June 1991.

[22] W.-S. Yu, I.-K. Fong, and K.-C. Chang. "An l_1 -approximation based method for synthesizing FIR filters," *IEEE Trans. Circuits, Syst. II*, vol. 39, pp. 578–561, Aug. 1992.

[23] J. S. Mason and N. N. Chit, "New approach to the design of FIR digital filters," *Proc. Inst. Elect. Eng.*, vol. 134, pt. G, no. 4, pp. 167–180, 1987.

[24] V. R. Algazi and M. Suk, "On the frequency weighted least-square design of finite duration filters," *IEEE Trans. Circuits, Syst.*, vol. CAS-22, pp. 943–953, Dec. 1975.

[25] Y. C. Lim, J.-H. Lee, C. K. Chen, and R.-H. Yang, "A weighted least squares algorithm for quasi-equripple FIR and IIR digital filter design," *IEEE Trans. Signal Processing*, vol. 40, pp. 551–558, Mar. 1992.

[26] E. W. Cheney, *Introduction to Approximation Theory*. New York: McGraw-Hill, 1966.

[27] M. J. D. Powell, *Approximation Theory and Methods*. Cambridge, U.K.: Cambridge Univ. Press, 1981.

[28] A. J. Sieradski, *An Introduction to Topology and Homotopy*. Boston, MA: PWS-Kent, 1992.

[29] V. L. Stonick and S. T. Alexander, "A relationship between the recursive least squares update and homotopy continuation methods," 1991, preprint.

[30] G. Strang, *Introduction to Applied Mathematics*. Wellesley, MA: Wellesley-Cambridge Press, 1986.

[31] R. E. Stone and C. A. Tovey, "The simplex and projective scaling algorithms as iteratively reweighted least squares methods," *SIAM Rev.*, vol. 33, no. 2, pp. 220–237, June 1991.

[32] J. D. Fisher, "Design of finite impulse response digital filters," Ph.D. dissertation, Rice Univ., Houston, TX, 1973.

[33] N. N. Chit and J. S. Mason, "Complex Chebyshev approximation for FIR digital filters," *IEEE Trans. Signal Processing*, vol. 39, pp. 49–51, Jan. 1991.

[34] D. Távora de Queiroz Cobra, "FIR digital filter design by the method of successive projections," Master's thesis, Mass. Inst. Technol., Jan. 1986.

[35] C.-Y. Tseng, "A numerical algorithm for complex Chebyshev FIR filter design," in *Proc. IEEE Int. Symp. Circ. Syst.* (San Diego, CA), May 1992, pp. 549–552.

[36] C.-Y. Chi and S.-L. Chiou, "A new self-initiated WLS approximation method for the design of two-dimensional equiripple FIR digital filters," in *Proc. IEEE Int. Symp. Circuits, Syst.* (San Diego, CA), May 1992, pp. 1436–1439.

[37] C.-Y. Tseng and L. J. Griffiths, "Are equiripple digital FIR filters always optimal with minimax error criterion?" *IEEE Signal Processing Letts.*, vol. 1, nol 1 pp. 5–8, Jan. 1994.

[38] ——, "A simple algorithm to achieve desired patterns for arbitrary arrays," *IEEE Trans. Signal Processing*, vol. 40, pp. 2737–2746, Nov. 1992.

[39] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Englewood Cliffs, NJ: Prentice-Hall, 1974.

[40] P. E. Gill, W. Murray, and M. H. Wright, *Practical Optimization.* London: Academic, 1981.
[41] C. Moler, J. Little, and S. Bangert, *Matlab User's Guide.* South Natick, MA: The Math Works, Inc., 1989.
[42] D. G. Luenberger, *Optimization by Vector Space Methods.* New York: Wiley, 1969.
[43] R. W. Schafer, R. M. Mersereau, and M. A. Richards, "Constrained iterative restoration algorithms," *Proc. IEEE*, vol. 69, pp. 432–450, 1981.
[44] C. B. Garcia and W. I. Zangwill, *Pathways to Solutions, Fixed Points, and Equilibria.* Englewood Cliffs, NJ: Prentice-Hall, 1981.
[45] J. M. Ortega and W. C. Rheinboldt, *Iterative Solutions of Nonlinear Equations in Several Variables.* New York: Academic, 1970.
[46] J. J. Dongarra, J. R. Bunch, C. B. Moler, and G. W. Stewart, *LINPACK User's Guide.* Philadelphia, PA: SIAM, 1979.
[47] J. A. Barreto, "L_p approximation by the iteratively reweighted least squares method and the design of digital FIR filters in one dimension," Master's thesis, Rice Univ., Houston, TX, July 1992.
[48] M. Lang and J. Bamberger, "Nonlinear phase FIR filter design with minimum LS error and additional constraints," in *Proc. Int. Conf. Acoust., Speech, Signal Processing* (Minneapolis, MN), Apr. 1993, pp. III-57–60.
[49] ——, "Nonlinear phase FIR filter design according to the L_2 norm with constraints for the complex error," *Signal Processing*, vol. 36, 1994.
[50] 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.
[51] M. Schulist, "Improvements of a complex FIR filter design algorithm," *Signal Processing*, vol. 20, pp. 81–90, 1990.
[52] ——, "Complex approximation with additional constraints," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Apr. 1992, pp. 101–104.
[53] K. Preuss, "On the design of FIR filters by complex approximation," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, pp. 702–712, May 1989.
[54] ——, "A novel approach for complex Chebyshev approximation with FIR filters using the Remez exchange algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* (Dallas, TX), Apr. 1987, pp. 872–875.
[55] J. A. Barreto and C. S. Burrus, "L_p complex approximation using iterative reweighted least squares for FIR digital filters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing* (Adelaide, Australia), Apr. 1994, pp. 19–22.
[56] J. H. Lodge and M. Fahmy, "An efficient L_p optimization technique for the design of two-dimensional linear-phase digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-28, 1980.
[57] D. B. Harris and R. M. Merserau, "A comparison of algorithms for minimax design of two-dimensional linear-phase FIR digital filters," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, 1977.
[58] V. R. Algazi, M. Suk, and C. S. Rim, "Design of almost minimax FIR filters in one and two dimensions by WLS techniques," *IEEE Trans. Circuits, Syst.*, vol. CAS-33, pp. 590–596, June 1986.
[59] J. A. Barreto and C. S. Burrus, "Iterative reweighted least squares and the design of two-dimensional FIR digital filters," in *Proc. IEEE Int. Conf. Image Processing* (Austin, TX), Nov. 1994, pp. 13–16.

C. Sidney Burrus (S'55–M'61–SM'75–F'81) received the B.A., B.S.E.E., and M.S. degrees from Rice University, Houston, TX, in 1957, 1958, and 1960, respectively, and the Ph.D. degree from Stanford University, Stanford, CA, in 1965.

From 1960 to 1962, he taught at the Navy Nuclear Power School in New London, CT, and during the summers of 1964 and 1965, he was a Lecturer in Electrical Engineering at Stanford University. In 1965, he joined the faculty at Rice University, where he is now Professor of Electrical and Computer Engineering and Director of the Computer and Information Technology Institute. From 1972 to 1978, he was Master of Lovett College at Rice University, and from 1984 to 1992, he was chairman of the ECE Department at Rice. In 1975–1976 and again in 1979–1980, he was a Guest Professor at the Universitaet Erlangen-Nuernberg, Germany. During the summer of 1984, he was a Visiting Fellow at Trinity College, Cambridge University, Cambridge, England, and during the academic year 1989–1990, he was a Visiting Professor at Massachusetts Institute of Technology, Cambridge. He has coauthored four books on digital signal processing.

Dr. Burrus is a member of Tau Beta Pi and Sigma Xi. He received teaching awards at Rice in 1969, 1974, 1975, 1976, 1980, and 1989, an IEEE S-ASSP Senior Award in 1974, a Senior Alexander von Humboldt Award in 1975, a Senior Fulbright Fellowship in 1985, and was a Distinguished Lecturer for the Signal Processing Society and for the Circuits and Systems Society from 1989 through 1992.

J. A. Barreto received the B.S. degree in electronic engineering from the Universidad Simon Bolivar, Venezuela, in 1983, and the M.S. degree in electrical and computer engineering from Rice University, Houston, TX, in 1992.

From April 1983 until October 1986, he was the Technical Manager for Teleservicios, which is a paging company in Venezuela. From November 1986 until June 1990, he worked as a Telecommunication Systems Engineer for Cerveceria Polar, which is a brewery company in Caracas, Venezuela. From August 1992 until May 1993, he worked as a research assistant for the Department of Electrical and Computer Engineering at Rice University. Currently, he is working as a Research Associate for the Institute for Neuroscience of the University of Texas at Austin. His research interests include digital filter design, biomedical image processing, multidimensional signal processing, and telecommunications.

Ivan W. Selesnick received the B.S. and M.E.E. degrees in electrical engineering in 1990 and 1991, respectively, from Rice University, Houston, TX. He is currently working toward the Ph.D. degree at Rice University.

He has been employed by McDonnell Douglas and IBM, working on neural networks and expert systems. His current research interests are in the area of signal processing, in particular, fast algorithms for DSP and digital filter design.