1-1-1974

Recursive Digital Filter Synthesis in the Time Domain

Francis Brophy

Andres C. Salazar

Recursive Digital Filter Synthesis in the Time Domain

FRANCIS BROPHY and ANDRES C. SALAZAR, Member, IEEE

Abstract—The nonlinear minimization problem that results from recursive digital filter design with phase constraints is simplified somewhat by working in the time domain. This paper describes techniques that utilize the time samples of the desired response as target values for an iterative minimization. Initial values for the α and β (feedforward and feedback) coefficients can be obtained by one of several reliable methods and fed into iterative routines that lead to a locally optimal solution for the coefficients. The initial guess procedures, stemming from regressionlike equations, only require the solution of a set of linear equations. In addition, the iteration procedures described in this paper lead to recursive filter designs requiring little computer time. Examples are presented to illustrate a range of applications.

I. Introduction

The design of recursive digital filters is made difficult because of the nonlinear programming problem involved and the need for good initial values of α (feedforward) and β (feedback) coefficients. If the design is formed in the frequency domain, then sampled amplitude and phase spectra of the ideal and approximating filters are compared after every adjustment of the α and β vectors. The computation of the spectra for this comparison requires many exponential and inverse trigonometric function evaluations that take up a substantial amount of time in the design program. If the number of α and β parameters is moderately large (e.g., 10 each) the number of iterations required to reach a local minimum is large and the computation time becomes prohibitively expensive. Also, there does not seem to be an easy method of obtaining good initial values for the α and β vectors when working in the frequency domain.

An alternative filter design procedure is presented here that utilizes time domain techniques to find a realizable filter whose response approximates an ideal time sequence. The closer the designed filter's impulse response is to the ideal, the "better" (e.g., L_2 or L_∞ metric) the approximation is in frequency with regard to both amplitude and phase. We recall that the impulse response of a digital filter is the sequence of Fourier coefficients of its frequency response. It can be shown that a frequency domain approximation is directly related to a search for a realizable filter whose impulse response matches a large number of the values of an ideal time sequence closely (see Appendix). Initial values for the α and β vectors can be easily obtained from this sequence by solving nothing more than a set of linear equations. Iterative programs, which then search for a set of α and β coefficients yielding an impulse response closest (e.g., least squares) to the desired one, require gradient (and sometimes Hessian) evaluations that are easily performed in the time domain with no need for transcendental function computations. As the degrees of freedom for the design of a recursive digital filter increase, namely, the number of α and β coefficients, computation savings become increasingly important. This aspect, as well as the ease with which good initial α and β values can be obtained, support the contention that the time domain may be the more natural domain for recursive filter design.

Time domain designs, however, can only deal with an overall frequency specification. That is, the approximation problem has been transferred from minimum stopband loss requirements and maximum passband deviation in the frequency domain to one involving maximum response deviations from an ideal time sequence. However, since initial guesses for α and β coefficients are difficult to obtain for frequency domain designs, it still may be practical to first implement a time domain procedure similar to one that we will treat in the next few sections, and then feed the locally optimal time domain solution to a frequency domain design procedure that will iterate towards a locally optimal frequency domain solution.

We emphasize that the need for a recursive digital filter design with specified phase does not arise from an academic viewpoint. Implementation considerations encourage minimization of the number of delays in a filter and this can be done easily with recursive filters with no sacrifice in the number of degrees of freedom in design. (By design degrees of freedom we mean the number of filter variables, namely, the feedback or feedforward tap settings.) For example, a common implementation is that of the cascaded second-order section that is multiplexed many times between incoming signal samples. Thus, in order to use the full capability of this implementation a recursive filter design is required. Also, in minimizing the number of delays in a filter design an important benefit is derived if coefficient length is a limitation. In general, the higher the order of the filter the greater will be the range of coefficients.

II. Time Domain Recursive Filter Synthesis

The impulse response of a causal recursive digital filter $\{g_n\}_0^\infty$ is a sequence of Fourier coefficients of some complex frequency function $G(e^{-j\omega T_0})$ on

Manuscript received May 17, 1973.
The authors are with Bell Laboratories, Holmdel, N.J. 07733.

$B \equiv \{\omega : |\omega| < (\pi/T_0)\}$ where T_0 is the sampling period. From classical analysis of Fourier series we understand that an approximation

$$H_K(\omega) = \sum_{n=-K_1}^{K_2} \tilde{h}_n e^{jn\omega T_0}$$

to a desired continuous spectrum $H(\omega)$, can be arbitrarily improved in a given norm (such as L_2 or L_∞) on B by making K_1 and K_2 sufficiently large. We introduce a constant delay $K_1 T_0$ and so we can now reindex $\{\tilde{h}_n\}_{K_1}^{K_2}$ to $\{h_n\}_0^T$ where $T = K_1 + K_2$.

We choose to measure the goodness of the approximation in the time domain by a function $\mathcal{E}$:

$$\mathcal{E} = \sum_{n=0}^{T} (h_n - g_n)^2. \tag{1}$$

It is evident that even if $\mathcal{E}$ is reduced to zero by adjustment of the α and β coefficients the spectrum approximation error has not been reduced to zero. This is apparent from the requirement that a realizable digital filter have at most a semi-infinite response. We refer to $\{h_n\}_0^T$ as the target sequence.

We recognize that the approximation error is composed of two parts, the first consisting of $\{h_n\}_{-\infty}^{-K_1-1}$ and the second is that of $\{h_n\}_{K_2+1}^{\infty}$. If $K_2 >> K_1$, the second part becomes insignificant compared to the first for most approximation problems of interest. Hence, in forming $\{h_n\}_0^T$ as the target sequence, we have sacrificed most of the approximation error to a realizability consideration.

The approximation of $H(\omega)$ on B by a harmonic series with respect to some norm is a subject which has been well studied (see Appendix). We need only mention that there exist various methods for obtaining the functions

$$H_K(e^{j\omega T_0}) = \sum_{n=-K_1}^{K_2} \tilde{h}_n e^{j\omega T_0}$$

such that $H_K \to H$ in B with respect to a given norm. For example, if the $L_2(B)$ norm is used then h_n are nothing more than the Fourier coefficients of $H(\omega)$ relative to the complex exponentials $\{e^{jn\omega T_0}\}_{-\infty}^{\infty}$. A least squares norm with a weighting function can also be used in which case the $\tilde{h}_n$ are the Fourier coefficients of $H(\omega)$ with respect to the set of functions orthonormalized with respect to that weighting function (e.g., Chebyshev polynomials). A brief review of methods for obtaining target sequence $\{\tilde{h}_n\}_{-\infty}^{\infty}$ which approximate the desired spectrum $H(\omega)$ with respect to the complex exponentials is given in the Appendix.

Let a realizable digital filter have the transfer function[1]

$$G(z^{-1}) = \frac{1 + \alpha_1 z^{-1} + \alpha_2 z^{-2} + \cdots + \alpha_M z^{-M}}{1 - \beta_1 z^{-1} - \beta_2 z^{-2} - \cdots - \beta_N z^{-N}} \qquad M \leq N. \tag{2}$$

By writing the impulse response recursion equations for $G(z^{-1})$:

$$g_n = \sum_{k=1}^{N} \beta_k g_{n-k} + \alpha_n \qquad 1 \leq n \leq M \tag{3}$$

$$g_n = \sum_{k=1}^{N} \beta_k g_{n-k} \qquad n > M \tag{4}$$

it then becomes clear that g_n is a nonlinear function of the $\{\beta_k\}_{k=1}^N$ and $\{\alpha_k\}_{k=1}^M$. Thus, the minimization of $\mathcal{E}$ is a difficult nonlinear programming problem. As the number of degrees of freedom increase, viz., $M + N$, the region over which minimization is sought becomes awesomely large. It therefore becomes imperative that the initial guess for the α and β vectors be of high quality to avoid long iteration times common for such nonlinear minimization problems.

A method [1] of reducing the number of variates in the minimization[2] of $\mathcal{E}$ consists of relating the α parameters to the iterating β vector by using the α's to force $g_n = h_n$ for $1 \leq n \leq M$ in (3). For example, the degrees of freedom would be cut in half if $M = N$.

III. Initial Guesses for the α and β Parameters

Some advantages of a good initial guess for α and β values for iterating towards a local minimum of $\mathcal{E}$ have been discussed in the previous section. We present here time domain techniques for obtaining initial guesses of the α and β parameters. In addition to being simple to implement, these techniques yield digital filter designs of excellent quality.

The advantage of having several ways of obtaining initial values for the α and β parameters becomes apparent when a particular method is used and an unstable β solution results. In such instances, another method offers an opportunity to find an initial β vector corresponding to a stable filter.

Since the α vector has a primary effect on the time response $\{g_n\}$ of (3) for the first M samples and has only a secondary effect on the time samples of (4) we can justify the attention given to solving for an optimal β vector separately in (4). Hence, we minimize (5) first and then

$$\mathcal{E}' = \sum_{n=M+1}^{\infty} \left(h_n - \sum_{k=1}^{N} \beta_k g_{n-k} \right)^2 \tag{5}$$

[1] $G(z^{-1})$ has been normalized at $z^{-1} = 0$ for notational convenience and reduced to the canonical form where $M \leq N$.

[2] The function $\mathcal{E}$ of (1) represents only one way of measuring the difference between the ideal and the synthesizable sequences. One can propose other functions for measuring this difference but $\mathcal{E}$ of (1) will yield a simple gradient for iteration purposes as we shall see.

return with a given set of $\{\beta_k\}_1^N$ to solve for $\{\alpha_k\}_1^M$ such that $g_n = h_n$, $n = 1, 2, \cdots, M$. Thus the minimum of $\mathcal{E}'$ is identical to the minimum of $\mathcal{E}$ (1) subject to this constraint on the α's. Keeping this idea in mind let us examine several methods of estimating a β vector.

A. Modified Least Squares

The actual response $\{g_n\}$ of a realizable filter can be decomposed into the ideal response $\{h_n\}$ and an error sequence. Thus, we write

$$g_n = h_n + \epsilon_n \tag{6}$$

where $\{\epsilon_n\}$ is an error sequence to be minimized:

$$\mathcal{E}' = \sum_{M+1}^{T} (h_n - g_n)^2 = \sum_{M+1}^{T} \epsilon_n^2 \tag{7}$$

where $T >> M + N$ (e.g., 10 times $M + N$). Rewriting (5) in terms of (6) we have

$$\mathcal{E}' = \sum_{M+1}^{T} \left(h_n - \sum_{k=1}^{N} \beta_k h_{n-k}\right)^2 - 2\sum_{M+1}^{T} \left(h_n - \sum_{k=1}^{N} \beta_k h_{n-k}\right) \cdot \left(\sum_{k=1}^{N} \beta_k \epsilon_{n-k}\right) + \sum_{M+1}^{T} \left(\sum_{k=1}^{N} \beta_k \epsilon_{n-k}\right)^2. \tag{8}$$

Instead of seeking to minimize simultaneously the three terms (absolute magnitude) on the right-hand side of (8), we minimize the first term only:

$$\mathcal{E}'' = \sum_{M+1}^{T} \left(h_n - \sum_{k=1}^{N} \beta_k h_{n-k}\right)^2. \tag{9}$$

This consideration is not without justification. For we note that minimizing $\mathcal{E}''$ has the additional effect of minimizing the second term of (8) because of the factor

$$\left(h_n - \sum_{k=1}^{N} \beta_k h_{n-k}\right).$$

Further, the sequence $\{\epsilon_n\}$ is minimized if β's are found so that

$$h_n \approx \sum_{k=1}^{N} \beta_k h_{n-k}$$

for every n.

The set of β's minimizing $\mathcal{E}''$ can be found easily from the normal equations[3]

$$\sum_{k=1}^{N} \beta_k \left(\sum_{M+1}^{T} h_{n-k} h_{n-l}\right) = \sum_{M+1}^{T} h_n h_{n-l} \qquad l = 1, 2, 3, \cdots, N. \tag{10}$$

The question of stability in obtaining this initial guess has never been fully explored. Experimental evidence suggests that for sufficiently large T and a continuous spectrum corresponding to $\{h_n\}$ the β vector that results from this initial guess offers a stable filter. Of course, $\{h_n\}_{M+1}^T$ is intended to contain a large portion of the "tail" of the time impulse response.

B. Padé Approximants[4]

It is possible to solve for the α's and β's from (3) and (4) so that $g_n = h_n$, $n = 0, 1, 2, \cdots, M + N$. This amounts to equating the first $M + N + 1$ coefficients of the power series expansion of $G(z^{-1})$ of (2) with

$$H(z^{-1}) = \sum_{n=0}^{\infty} h_n z^{-n}$$

representing the ideal transfer function. Of course, $G(z^{-1})$ does not have a sufficient number of degrees of freedom so that $g_n = h_n$, all n, but a sufficient number of the Fourier coefficients $\{h_n\}$ will be matched exactly for a reasonable spectrum approximation.

Some care should be exercised in forming the Padé approximant since an unstable filter may result. Experience has shown that two factors that could lead to instability are the discontinuity of the spectrum specification and a choice of $M + N$ that is too low. A simple test can be performed on the samples $\{h_n\}_0^{M+N}$ in order to determine whether stability will result from the Padé synthesis technique [2].

C. Generalizations to Padé or Modified Least Squares Techniques

Simplicity is an important advantage of any initial guess procedure and often outweighs any other distinction an initializing procedure may have. We offer several generalizations of the two previous methods discussed that still entail only the solution to a set of linear equations.

1) An obvious generalization to the modified least squares method involves weighting the target sequence approximation:

$$\mathcal{E} = \sum_{n=M+1}^{T} \left(h_n - \sum_{k=1}^{N} \beta_k h_{n-k}\right)^2 w_n$$

where $w_n \geq 0$, $n = M + 1, M + 2, \cdots, T$.

Again we assume that the α's will be used to match $\{h_n\}_0^M$ exactly. We can see easily that $w_n = 1$, $M + 1 \leq n \leq M + N$, $\omega_n = 0$, otherwise yields the Padé approximant technique, while $w_n = 1$, all n, forms the modified least squares method. By choosing an appropriate set of weights $\{w_n\}_{M+1}^T$, we argue that improvement over the Padé initial guess could result if we more closely matched the first $N + K$ samples (past the zeros' influence) at the expense of not matching the first N exactly. In the case of modified least squares we could de-emphasize the smaller

[3] Shanks [1] in a related study encountered the same equations.

[4] See [2] for a fuller discussion of the Padé technique.

samples in the pulse tail and weight more heavily the large samples near the pulse peak (e.g., in relation to their contribution to the overall energy).

2) Instead of solving (3) and (4) for the α's and β's that yield $g_n = h_n$, $n = 0, 1, 2, \cdots, M + N$, we could choose to solve these equations for any index set I of cardinality $M + N$ (by assuming $g_n = h_n$ over all pertinent index n). The problem is linear and easily solved. The inherent assumption (viz., $g_n = h_n$), for some n, induces errors but at the same time points further from the pulse peak will be used to influence the solution. This technique can best be applied when the time response is smooth and matching the first $M + N + 1$ samples seems redundant.

3) We could replace the entries to the Padé approximant matrix equation by values which represent averaged values about that entry, e.g., replace h_n by

$$\frac{1}{5}\sum_{n-2}^{n+2} h_k \quad \text{for all } n.$$

The strategy is obvious. We hope to sacrifice the accuracy of any one sample to gain the influence of a few more points.

IV. Iterative Routines

The nonlinear nature of $\mathcal{E}'$ in (5) necessitates the use of an iterative routine for its minimization. Since $\mathcal{E}'$, as a function of the β's, can have its gradient and Hessian easily computed, a facile implementation of one of several minimization methods leads to a locally optimal solution.

Most unconstrained minimization routines are related through the matrix A_n found in the iterative equation

$$\vec{x}_{n+1} = \vec{x}_n - \Delta_n A_n \vec{\nabla}\mathcal{E}'(\vec{x}_n) \tag{11}$$

where $\vec{x}_n$ is the estimate of the location of the minimum of $\mathcal{E}'$ at the nth step, Δ_n is the nth step size, and $\vec{\nabla}\mathcal{E}'$ is the gradient of the object function $\mathcal{E}'$ evaluated at $\vec{x}_n$. Convergence to a local minimum of $\mathcal{E}'$ requires that A_n be a positive definite matrix at every step. For the steepest descent algorithm $A_n \equiv I$ while a Newton-Raphson technique uses $A_n = H^{-1}(\vec{x}_n)$, the inverse of the Hessian of $\mathcal{E}'$ evaluated at $\vec{x}_n$. The latter method works best when $\mathcal{E}'$ is a convex function in the neighborhood of every $\vec{x}_n$. Finally, the Fletcher-Powell (F-P) minimization algorithm [3] changes A_n monotonically from $A_0 = I$ to the inverse Hessian at the final estimate of $\mathcal{E}'$'s minimum.

We recall that $\mathcal{E}'$ can be written

$$\mathcal{E}' = \sum_{n=M+1}^{T}\left(h_n - \sum_{k=1}^{N}\beta_k g_{n-k}\right)^2 \tag{12}$$

where constraints on the α's allow $g_n = h_n$, $n = 1, 2, \cdots M$ in (3).

The gradient vector of $\mathcal{E}'$ denoted by $\vec{\nabla}\mathcal{E}' = (e_1, e_2, \cdots, e_N)$ has component (noting g_n is also a function of β)

$$e_k = \frac{\partial \mathcal{E}'}{\partial \beta_k} = -\sum_{n=M+1}^{T} 2(h_n - g_n)\left(g_{n-k} + \sum_{l=1}^{N}\beta_l a_{n-l}(k)\right)$$
$$k = 1, 2, \cdots, N. \tag{13}$$

The array $a_m(k)$ is computed from the formula

$$a_m(k) = \frac{\partial g_m}{\partial \beta_k} = g_{m-k} + \sum_{l=1}^{N}\beta_l a_{m-l}(k) \qquad m \geqslant k$$
$$k = 1, 2, \cdots, N \tag{14}$$

with $a_m(m) = 1$ for $m = 1, 2, \cdots N$, and $a_m(n) = 0$ for $m < n$.

For minimization algorithms requiring Hessian evaluations, the array $a_m(k)$ again comes into play:

$$V_{jk} = \frac{\partial^2 \mathcal{E}'}{\partial\beta_j \partial\beta_k} = 2\sum_{n=M+1}^{T}\left\{a_n(j)\left[g_{n-k} + \sum_{l=1}^{N}\beta_l a_{n-l}(k)\right]\right.$$
$$\left. - (h_n - g_n)\left[2a_{n-k}(j) + \sum_{l=1}^{N}\beta_l b_{n-l}(j,k)\right]\right\}$$

where V_{jk} is the j, k entry of the Hessian of $\mathcal{E}'$ and $b_m(j, k)$ is an array that is also computed iteratively:

$$b_m(j,k) = \frac{\partial g_m}{\partial\beta_j\partial\beta_k} = a_{m-k}(j) + a_{m-j}(k) + \sum_{l=1}^{N}\beta_l b_{m-l}(j,k)$$

where $m = 1, 2, \cdots, T$ and $1 \leqslant j, k \leqslant N$.

At first glance, it would seem that calculation of $a_m(k)$ and $b_n(j, k)$ require a great deal of computation. However, it is easy to show that

$$a_m(k) = a_{m+1}(k+1) \qquad k = 1, 2, \cdots N - 1.$$

Thus, only $a_m(1)$, $m = 1, 2, \cdots, T$ need be computed. Similarly, since $b_m(j, k) = b_m(k, j)$ and $b_m(j, k) = b_{m+1}(j, k + 1)$ and $b_m(j, k) = b_{m+2}(j + 1, k + 1)$ where $b_m(j, k) = 0$ if $j + k > m$ and $b_{j+k}(j, k) = 2$, we need only compute $b_m(1, 1)$, $m = 1, 2, \cdots, T$. Hence, the computation of V_{jk} is greatly simplified.

Success in iterating to a locally optimal solution for the β's depends on a judicious choice for Δ_n. The sensitivity of the location of the poles with respect to the β's can be seen easily from the theory of equations. Hence, it is possible that the filter becomes unstable for only small changes in the coefficients of the transfer function's denominator polynomial. Any iterative routine would then have to proceed cautiously in sequentially choosing its adjustment step size Δ_n in order to prevent an unbounded time response $\{g_n\}$. We have found that standard unconstrained minimization programs such as that of the

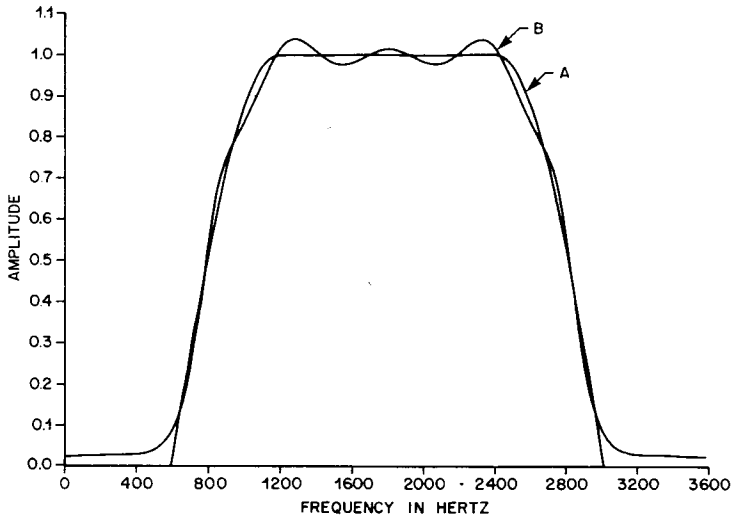

Fig. 1. Padé approximant as initial guess-amplitude comparisons.

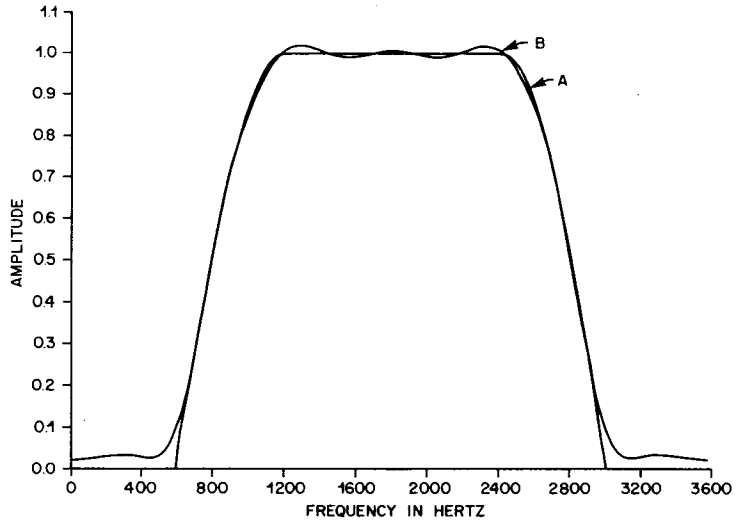

Fig. 2. Steepest descent minimization-amplitude comparisons.

IBM scientific subroutines implementation of the F-P algorithm were not suitable for our usage because of its lack of control on step sizes. However, we have been able to program a version of the F-P algorithm suitable for the filter synthesis problem that we have outlined.

V. Examples

We illustrate the synthesis procedures discussed in the previous sections with a few practical design examples typically found in data transmission system applications.

The first example is one of a spectrum shaping bandpass filter, the square root of a baseband signaling filter. In Fig. 1 we show the desired shape to approximate (A) along with the initial guess to its approximation (B). The initial guess is obtained using the Padé technique (see Section III-B), and as can be seen, this serves as a fine initial approximation. Starting from this guess, a steepest descent algorithm was applied to further improve the approximation to that of form B in Fig. 2. Two hundred thirty-three gradient evaluations were required to converge to this tenth order approximation. Also, as shown in Fig. 3, the associated phase approximates linear phase in the passband quite well. In fact, after removing the linear phase introduced by the approximation the phase deviates by no more than 0.65° in the passband. We finally note that the final approximation has a square error term [$\mathcal{E}'$ of (12)] of 0.165 times the initial guess squared error term, bringing the largest absolute error for the time samples to 0.011 where $\max_n |h_n| = 1$. In the frequency domain the maximum deviation between the specified curve A and the approximating spectrum B was 0.02 (or 0.17 dB) in the passband (900-2700 Hz).

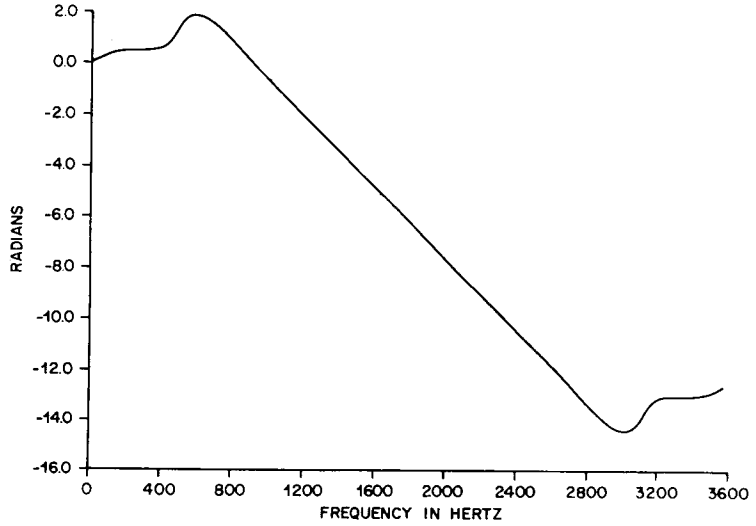

Fig. 3. Steepest descent minimization-resultant phase.

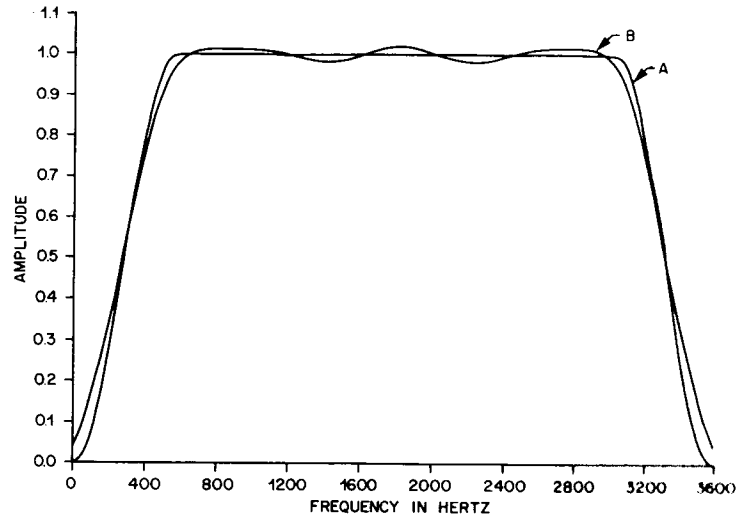

Fig. 4. Steepest descent minimization-amplitude comparisons.

As a second example we show a tenth order recursive digital filter design approximating a Hilbert transformer. For our purposes, we gave the filter a smooth rolloff (raised cosine) to assist the convergence of the time sequence. Again, the Padé approximant technique was applied to obtain the initial guess. Starting from there 23 iterations of the steepest descent algorithm brought the approximation to that of form B in Fig. 4. The resultant phase is $90° \pm 2.8°$ (having removed the linear phase which was introduced by the approximation). Again we note that the squared error term was reduced by a factor of 0.9 bringing the maximum absolute error time sample to 0.013 with $\max_n |h_n| = 1$. We also note a maximum magnitude error of 0.027 (or 0.2 dB) in the frequency specification over the passband.

The third example is a typical low-pass filter. We again gave the filter a raised cosine rolloff to assist the convergence of the time sequence. The desired shape (A) and its final approximation (B, eleventh order) can be seen in Fig. 5. The method used for obtaining the initial guess for this approximation was referred to in Section III-C, and consisted of matching every other sample of the first $2N$. We sacrificed accuracy, in that all samples were not matched exactly, but hopefully gained in the approximation, in that more samples affected the approximation and the general nature of the time response was faithfully reproduced.

We recorded that a Fletcher-Powell minimization converged to the final approximation in nine steps. The associated items of interest are: 1) a 3.7° deviation in phase in the passband (linear term removed), 2) a squared error reduction by 0.117 from the initial guess to the final approximation, and 3) a maximum absolute error of 0.012 for the time samples with $\max_n |h_n| = 1$, while the maximum deviation in the frequency specification was 0.021 (or 0.18 dB). Perhaps, at this point, two issues should be cleared. First, we cannot hope to claim that these optima are

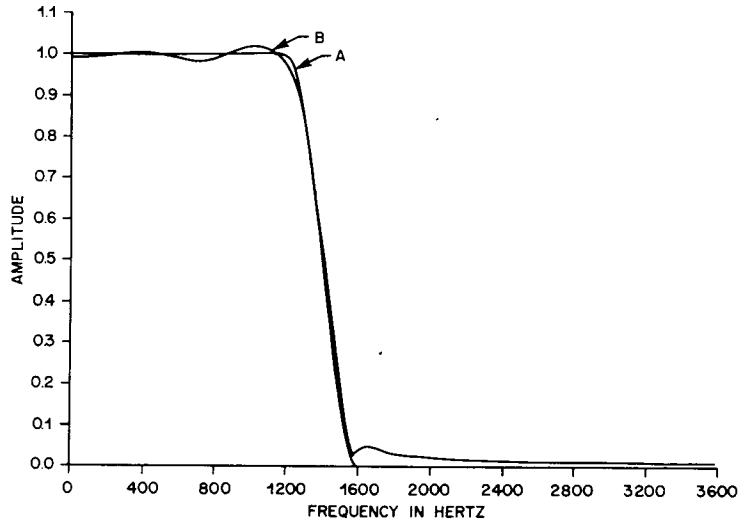

Fig. 5. Fletcher-Powell minimization-amplitude comparisons.

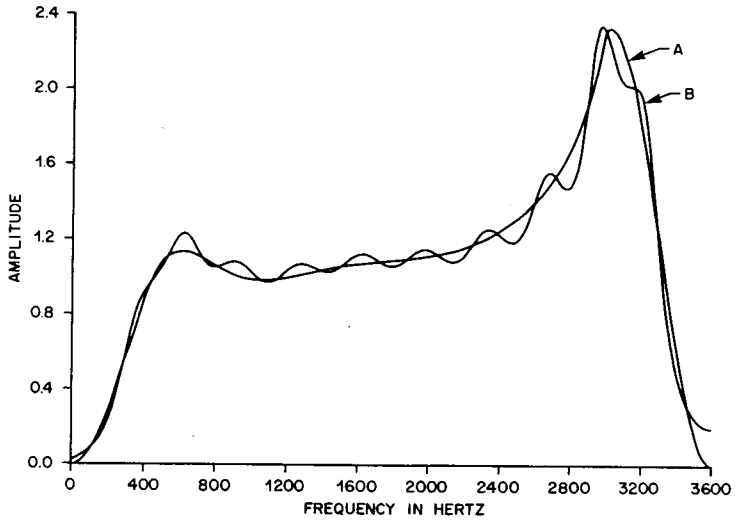

Fig. 6. Modified least squares as initial guess-amplitude comparisons.

anything more than local. In fact, intuitively, we feel that the chance of finding a good local optimum depends on the technique of determining an initial guess. And this notion is precisely what motivated the development of the work on initial guesses reported here. Second, we have found that the generally powerful Fletcher-Powell minimization technique does not significantly speed up convergence to a local optimum for the time domain filter design techniques we have treated here. When the number of iterations (namely, gradient evaluations) is relatively small the steepest descent algorithm serves adequately enough, especially as it is simpler to implement. In this same light, implementing any modified Newton-Raphson method, which requires Hessian evaluations does not appear to be useful.

As a last example, we present what may appear to be a difficult design problem, a compromise equalizer. We identify the magnitude specification as form A in Fig. 6; this should typify a moderately complicated shape (Fig. 8, form A gives the associated phase). We chose a twentieth order approximation. The initial guess (form B, Fig. 6) was obtained using the modified least squares technique (see Section III-A) and after eight iterations of a Fletcher-Powell minimization the final approximation (magnitude) can be found in Fig. 7. The phase approximation can be appreciated by viewing Fig. 8. Here we record a maximum phase error of 3.5° in the passband and a maximum absolute error over the time samples of 0.099 with $\max_n |h_n| = 1$, reducing the overall error by a factor of 0.6 in the minimization from the initial guess. The maximum magnitude deviation was 0.06. We capsule the results of all the examples in Table I.

It is interesting to note that the size of the approximation is not necessarily the limiting variable using the described time domain techniques. Typically, as the number of variables increases in a nonlinear mini-

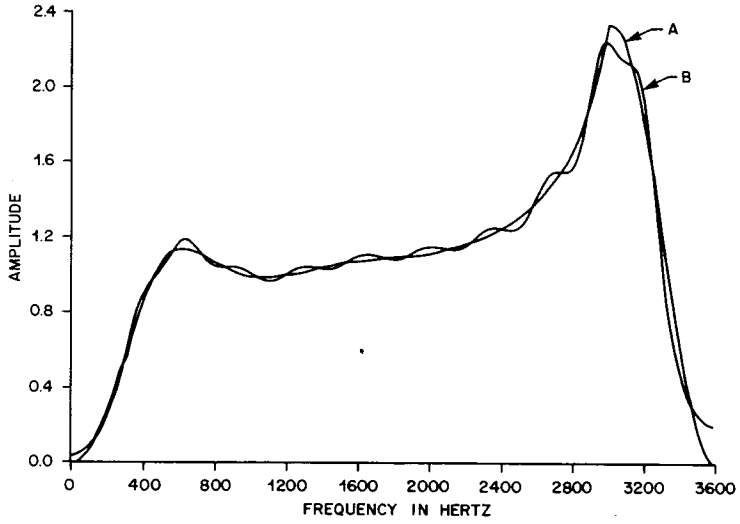

Fig. 7. Fletcher-Powell minimization-amplitude comparisons.

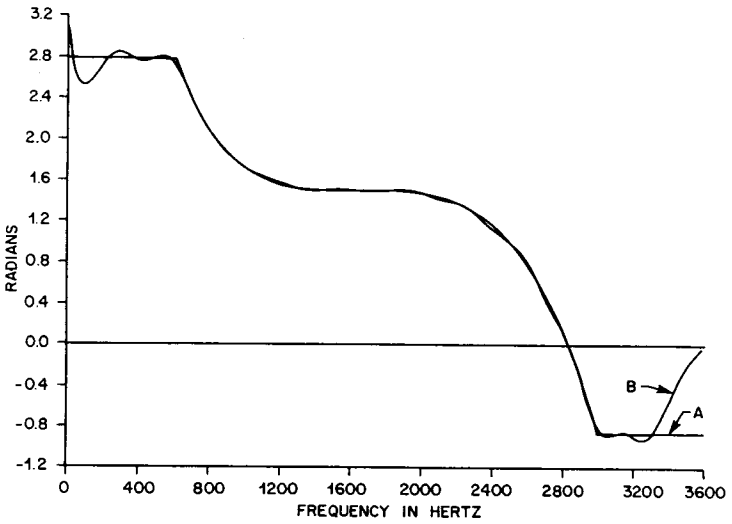

Fig. 8. Fletcher-Powell minimization phase-linear term removed.

mization problem, iteration times become longer. However, due to the nature of the initial guess, this phenomenon does not appear here. In fact the initial guesses also improve with the degree of the filter. And so, one could possibly envision cases where the iteration times actually decrease with increasing order of the approximation (of course, the time to obtain an initial guess will increase slightly).

We can perhaps now summarize an interesting phenomenon associated with the time domain synthesis of recursive digital filters, namely, this technique is more ideally suited for filters of higher degree (≥ 8). There is nothing conceptually wrong with designing filters of low order using this technique; however, from a practical viewpoint the methods of obtaining stable initial guesses are not as standardized. (For example, the Padé approximant cannot be guaranteed to be stable, especially for low order filters.) Of course, there is nothing to prevent the designer from obtaining his own initial guess and then iterating from there.

Finally, we remind the reader that in the time domain the amplitude and phase cannot be separated. That is to say that the quality of the approximation cannot be varied, at will, between amplitude and phase. Likewise, in the time domain the designer has no simple way to guarantee stability (such as root reflection when the approximation is done in the frequency domain) without seriously altering the desired samples to be approximated. (Once an initial guess is stable then further iterations remain stable.) We refer the reader to [4], [13] for a discussion of the frequency domain design of recursive digital filters with specified magnitude and phase characteristics.

VI. Conclusions

Recursive filter designs are often required in digital circuits in which the number of delays is to be minimized and coefficient accuracy is limited. The main obstacle that arises for such designs is that the determination of the α's and β's that will yield a desired

TABLE I
Design Characteristics of Figs. 1-8

Filter Name	Bandpass Shaping Filter	Phase Shifter	Low Pass	Compromise Equalizer
Figs.	1-3	4	5	6-8
Order of filter	10	10	11	20
Type of initial guess	Padé	Padé	modified Padé	modified least squares
Type of minimization	steepest descent	steepest descent	Fletcher-Powell	Fletcher-Powell
Number of iterations	233	28	9	8
Percent reduction	83.6	29.8	10	30.2
Maximum relative time sample error	0.011	0.099	0.013	0.012
Maximum relative magnitude error	0.0201	0.06	0.027	0.0207
Maximum phase error (linear term removed)	0.646°	3.52°	2.81°	3.21°
Average phase error (linear term removed)	0.005°	0.77°	0.847°	0.407°

spectral shape is a nonlinear programming problem. A simplication occurs if the desired spectrum constraints are carried into the time domain by forming a target time sequence (see Appendix for several methods of obtaining this sequence). The approximating recursive filter is then forced via one of several techniques to have its impulse response follow (in the least squares sense) this target sequence. Iterative routines can then be employed to adjust the α's and β's further until a locally optimal solution is reached.

An important feature of this time domain approach is that it yields simple initial guesses to α and β values for arbitrary spectral specifications. Further, iteration in the time domain is void of complex arithmetic since both magnitude and phase requirements have been transformed into a real time sequence. Examples have been provided to illustrate the flexibility of this approach.

A computer program has been written incorporating the filter synthesis techniques outlined in this paper. The program consists of five sections: 1) the generation of the desired frequency response, 2) an application of the fast Fourier transform to obtain the associated time response, 3) the calculation of initial α and β coefficients corresponding to a particular initial guess procedure, 4) the continuation of the minimization of $\mathcal{E}'$ through either the steepest descent or the Fletcher-Powell algorithms, and 5) output, including the printing and plotting of the final frequency and time approximations. In all the examples, the minimization (i.e., 2-4) time was found to be less than 1 min on an IBM 370. This maximum time is especially significant when one notes that the times required to design even low-order recursive filters (typically of degree 6 or so) using frequency domain techniques take well over a minute [4], [12], [13].

Appendix—Review of Approximation Theory for Digital Filter Synthesis

We reduce the digital filter synthesis problem to a problem of approximation by a certain class of analytic functions on the unit disk. To see this, we start with a familiar setting.

A. Approximation by a Finite Trigonometric Polynomial

Let $H(f)$ denote a piecewise continuous, real and even function defined on $[-\frac{1}{2}, \frac{1}{2}]$. We can formally write the truncated Fourier series for $H(f)$ by

$$S_N(f) = \sum_{n=-N}^{N} a_n e^{in2\pi f} \qquad |f| \leqslant \tfrac{1}{2} \qquad \text{(A1)}$$

where a_n is the Fourier coefficient of $H(f)$ relative to $e^{in2\pi f}$. Of course, the sequence of functions $S_N(f)$ need not converge pointwise everywhere although $L_2[-\frac{1}{2}, \frac{1}{2}]$ and a.e. convergence is guaranteed. The function $H(f)$ here represents the spectrum requirement of the filter. Generally, this function is specified by a sequence of straight lines or other smooth curves. In any case, a finite number of discontinuities (of the first kind) of $H(f)$ may result. It is at these points of discontinuity that convergence of (A1) becomes a problem. This is nothing more than a conclusion of Riemann's principle of localization [5, vol. I, p. 103]. In the subintervals where $H(f)$ is defined by C_p, $p \geqslant 1$ (p-differentiable) functions, we have uniform convergence of (A1). However, in the neighborhood of a point of discontinuity we experience Gibbs' phenomenon with the associated slow convergence of $S_N(f)$ to $H(f)$ at points of continuity of $H(f)$.

Mitigation of this convergence problem is accomplished by adjusting the coefficient sequence $\{a_n\}_{-N}^{N}$ as we shall see.

We find that S_N can be written

$$S_N(f) = \int_{-1/2}^{1/2} H(u)\, D_N(u - f)\, du$$

where $D_N(u)$ is the Dirichlet kernel:

$$D_N(u) = \frac{\sin\left(N + \frac{1}{2}\right) 2\pi u}{2 \sin \pi u} \qquad |u| < \tfrac{1}{2}, \qquad \text{(A2)}$$

or reproducing kernel of T_N, the set of trigonometric polynomials up to degree N, is taken as a subspace of $L_2[-\frac{1}{2}, \frac{1}{2}]$. We note that the Dirichlet kernel corresponds to no adjustment of the Fourier coefficients $\{a_n\}_{-N}^{N}$. If we transform $\{a_n\}_{-N}^{N}$ to

$\{\lambda_n^{(N)} a_n\}_{-N}^{N}$ where $\lambda_n^{(N)} = 1 - (n/N)$ we then obtain the integral equation

$$\tilde{S}_N(f) = \int_{-1/2}^{1/2} H(u)\, K_N(u - f)\, du \qquad \text{(A3)}$$

where

$$K_N(u) = \frac{1}{N}\left(\frac{\sin N\pi u}{\sin \pi u}\right)^2, \qquad \text{(A4)}$$

the Fejér kernel.

The benefit derived from the preceding adjustment is that on any closed interval of continuity of $H(f)$, $\tilde{S}_N(f)$ converges uniformly to $H(f)$. Further [5, vol. I, p. 135] at a point of discontinuity (of the first kind) f_0 of H, $S_N(f)$ converges to

$$\frac{H(f_0 + 0) + H(f_0 - 0)}{2}.$$

Most importantly, perhaps, is that

$$\lim_{\epsilon \to 0} \lim_{N \to \infty} \tilde{S}_N(f_0 \pm \epsilon) = H(f_0 \pm 0),$$

which was not possible with the Dirichlet kernel.

In general,[5] any adjustment of $\{a_k\}_{-N}^{N}$ to $\{\lambda_k^{(N)} a_k\}_{-N}^{N}$ where $\lambda_k^{(N)}$ is real and

$$\lambda_k^{(N)} = \lambda_{-k}^{(N)} \qquad k = 1, 2, \cdots N$$

and

$$\lim_{N \to \infty} \lambda_k^{(N)} = 1 \qquad \text{(A5)}$$

and for all N,

$$\int_{-1/2}^{1/2} 2 \sum_{k=-N}^{N} \lambda_k^{(N)} e^{ik2\pi u}\, du < \text{constant} \qquad \text{(A6)}$$

results in an integral equation of the type

$$\tilde{S}_N(f) = \int_{-1/2}^{1/2} H(n)\, A_N(u - f)\, du \qquad \text{(A7)}$$

where

$$A_N(u) = \lambda_0^{(N)} + 2 \sum_{k=1}^{N} \lambda_k^{(N)} \cos 2\pi u, \qquad |u| < \tfrac{1}{2}.$$

If the kernel $A_N(u)$ is formed this way then $\tilde{S}_N(f)$ of (A7) enjoys the same convergence properties as $\tilde{S}_N(f)$ of (A3) associated with the Fejér kernel. There exist many convergence factors $\{\lambda_n^{(N)}\}_{-N}^{N}$ that are known to improve the convergence of $S_N(f)$ near a point of discontinuity of $H(f)$ [6, p. 220] or [7, p. 200]. In passing we note that the factors $\{\lambda_n^{(N)}\}_{-N}^{N}$ have not been tabulated which transform the approximation $S_N(f)$ to one consisting of the projection of $H(f)$ onto the Chebyshev polynomials on $[-\frac{1}{2}, \frac{1}{2}]$. Also it is not known whether there exist factors that transform $S_N(f)$ into the trigonometric polynomial of best approximation to an $H(f)$ continuous on $[-\frac{1}{2}, \frac{1}{2}]$ (or a compact subset thereof), although it is known that any such map must satisfy a Lipschitz condition for every continuous $H(f)$ [11, p. 27].

The difference in approximation between that given by Chebyshev polynomials and the polynomial of best approximation is very slight if $H(f)$ is continuous [8, p. 127]. Remez [9] algorithms or derivatives thereof are usually implemented to obtain the polynomial of best approximation to a continuous $H(f)$. Since such algorithms are iterative, the Chebyshev expansion appears to be the simpler one to implement in many cases.

When $H(f)$ is a complex function but Hermitian, (real even part and odd imaginary part) the truncated Fourier series of (A1) is still available as an approximation. Except for interpolation procedures in the complex plane [10, pp. 243–252] or [11, pp. 76–80] this seems to be the only viable method of approximation for obtaining a sequence $\{a_n\}_{-\infty}^{\infty}$ such that

$$S_N(f) = \sum_{-N}^{N} a_n e^{in2\pi f} \approx H(f).$$

B. Approximation by Analytic Functions on Unit Disk

We have seen in Section A of this Appendix that a finite sequence $\{h_n\}_{-N}^{N}$ can be formed in many ways so that

$$\sum_{n=-N}^{N} h_n e^{jn2\pi f}$$

provides an excellent approximation to $H(f)$ on $[-\frac{1}{2}, \frac{1}{2}]$. Let us now focus attention on the case where $\{h_n\}_0^{\infty}$ represents the ideal response of a recursive digital filter.

The response function of a realizable digital filter is

$$G(z^{-1}) = \sum_{n=0}^{\infty} g_n z^{-n} \qquad \text{(A8)}$$

which, for reasons of stability, is an analytic function in the closed unit disc $|z^{-1}| \leqslant 1$. However, if $H(f)$ is real then the approximation to the desired spectral function is in the form

$$H(z^{-1}) = \sum_{n=-\infty}^{\infty} h_n z^{-n} \qquad \text{(A9)}$$

where, by substituting $z = e^{i2\pi f}$, we can obtain the more familiar form discussed in Section A. If $\Sigma |h_n| < \infty$ then (A9) represents the real part of an analytic function of the type in (A8) since $h_n = h_{-n}$. Since linear phase is not a detriment to our approximation, the problem becomes one of finding g_n's such that $G(Z^{-1})$ approximates a shifted version of $H(z^{-1})$.

$$G(z^{-1}) \sim z^{-M} H(z^{-1}) \qquad \text{for a finite integer } M. \qquad \text{(A10)}$$

In terms of the Hardy spaces $\mathcal{H}^p$, $p \geqslant 1$ on the unit

[5] See [5, vol. II, p. 3] or [6, p. 220].

disk this approximation problem takes on a simple form if $p = 2$ is considered. For a fixed M we can find a function $G(z^{-1}) \in \mathcal{H}^2$ such that

$$\int_{-1/2}^{1/2} |G(e^{-i2\pi u}) - e^{i2\pi Mu} H(e^{-i2\pi u})|^2 \, du \quad \text{(A11)}$$

is minimum (assuming $H(e^{-i2\pi u}) \in L_2(V)$ where V is the unit circle). Since $\mathcal{H}^2$ is a Hilbert space on V [in particular a closed subspace of $L_2(V)$] the function $G(z^{-1})$ consists of nothing more than the projection of $z^{-M} H(z^{-1})$ onto $\mathcal{H}^2$:

$$G(z^{-1}) = \sum_{n=0}^{\infty} h_{n+M} z^{-n} \quad \text{(A12)}$$

or simply the truncated expansion of $z^{-M} H(z^{-1})$. The reproducing kernel (for $\mathcal{H}^2$ as a subspace of L_2) in this case is the Szegö kernel

$$K_u(u, v) = \left(\frac{1}{1 - u\bar{v}}\right).$$

For $\mathcal{H}^p$ approximation G to an L_p function H (again on the unit circle) for $1 < p < \infty$, it is known [11, p. 60] that the best $\mathcal{H}^p$ approximation G^* has the property that for a.e., $|u| < \frac{1}{2}$

$$e^{-i2\pi u} F(e^{-i2\pi u}) (G^*(e^{-i2\pi u}) - H(e^{-i2\pi u})) \geq 0$$

and

$$|F(e^{-i2\pi u})| = |G^*(e^{-i2\pi u}) - H(e^{-i2\pi u})|^{p-1}$$

for some $F(e^{-i2\pi u}) \in \mathcal{H}^q$ where $q = p/(p-1)$. Further, the L_p norm of the smallest difference $G^* - H$ in approximation is

$$\left\{\frac{1}{2\pi i} \int_{|z^{-1}|=1} [F(z^{-1}) (G^*(z^{-1}) - H(z^{-1}))] \, dz^{-1}\right\}^{1/p}.$$

An explicit solution for $G^*(e^{-i2\pi u})$ approximating an $H(e^{-i2\pi u})$ of $L_p(V)$ is unknown except for some simple cases. In this paper the solution of (A12) will be used explicitly.

References

[1] J. L. Shanks, "Recursion filters for digital processing," *Geophysics*, vol. 32, pp. 33-51, Dec. 1967.

[2] F. Brophy and A. C. Salazar, "Considerations of the Padé approximant technique in the synthesis of recursive digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-21, pp. 500-505, Dec. 1973.

[3] R. Fletcher and M. J. D. Powell, "A rapidly convergent descent method for minimization," *Comput. J.*, vol. 6, no. 2, pp. 163-168, 1963.

[4] R. E. King and G. W. Condon, "Frequency domain of a class of optimum recursive digital filters," *Int. J. Contr.*, vol. 17, no. 3, pp. 497-509, 1973.

[5] N. K. Bary, *A Treatise on Trigonometric Series*, vols. I, II. New York: Pergamon, 1964.

[6] I. P. Natanson, *Constructive Function Theory*, vol. I. New York: Ungar, 1964.

[7] J. R. Rice, *Approximation of Functions*, vol. I. Reading, Mass.: Addison-Wesley, 1964.

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

[9] E. Y. Remez, *General Computational Methods of Chebyshev Approximation*, Book 1, U.S. AEC-tr-4491, also, *The Problems with Linear Real Parameters*, Book 2, U.S. AEC-tr-4491.

[10] E. L. Stiefel, *Introduction to Numerical Mathematics*. New York: Academic, 1963.

[11] H. S. Shapiro, *Topics in Approximation Theory: Lecture Notes in Mathematics*, no. 187. New York: Springer, 1971.

[12] K. Steiglitz, "Computer aided design of recursive digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-18, pp. 123-129, June 1970.

[13] A. G. Deczky, "Synthesis of recursive digital filters using the minimum p-error criterion," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 257-263, Oct. 1972.

Reprinted by permission from IEEE TRANSACTIONS ON ACOUSTICS, SPEECH, AND SIGNAL PROCESSING
Vol. ASSP-22, No. 1, February 1974, pp. 45-55