Approximate Design Relationships for Low-Pass FIR Digital Filters

LAWRENCE R. RABINER

Abstract—In this paper, a set of simple, approximate relationships between FIR, linear phase, low-pass filter parameters is presented. Based on these relationships, it is shown how an existing, readily available, filter design program can be used to efficiently design low-pass filters that meet or exceed arbitrary input specifications.

Introduction

Although a great deal has been learned about how to design linear phase FIR low-pass digital filters [1]-[5] the relationships between filter parameters is not yet fully understood quantitatively. Fig. 1 shows the frequency response of a low-pass filter with passband cutoff frequency F_p; stopband cutoff frequency F_s; passband deviation δ_1; and stopband deviation δ_2. Along with the preceding four parameters, an FIR low-pass filter is specified by N, the filter impulse response duration in samples. Additional useful low-pass filter parameters are transition width ΔF and deviation ratio K, defined as

$$\Delta F = F_s - F_p \tag{1}$$

$$K = \delta_1/\delta_2. \tag{2}$$

An optimal low-pass filter (in the Chebyshev sense) is one for which N, F_p, F_s, and K are specified, and δ_1 and δ_2 are minimized. Parks and McClellan [4] have given the necessary and sufficient conditions for such an optimal filter to exist and be unique. Furthermore, they have presented in great detail a design program [6] which on input accepts values of N, F_p, F_s, and K, and then designs an optimum filter with the minimum values of δ_1 and δ_2. There is little flexibility in the algorithm in that the user cannot arbitrarily choose *any* four of the five filter parameters, but must instead choose the ones listed previously. This constraint on the design program led Herrmann *et al.* [7] to investigate the relationships between filter parameters so as to be able to predict the required value of N to meet given values of F_p, F_s, δ_1, and δ_2. In this paper the design relationships are extended so as to interface with the Parks-McClellan algorithm in an iterative, but straight forward manner so as to allow the filter designer the option of choosing any four of the five low-pass parameters and having the resulting filter meet or exceed the given specifications.

Manuscript received April 26, 1973; revised June 4, 1973.
The author is with Bell Laboratories, Murray Hill, N.J. 07974.

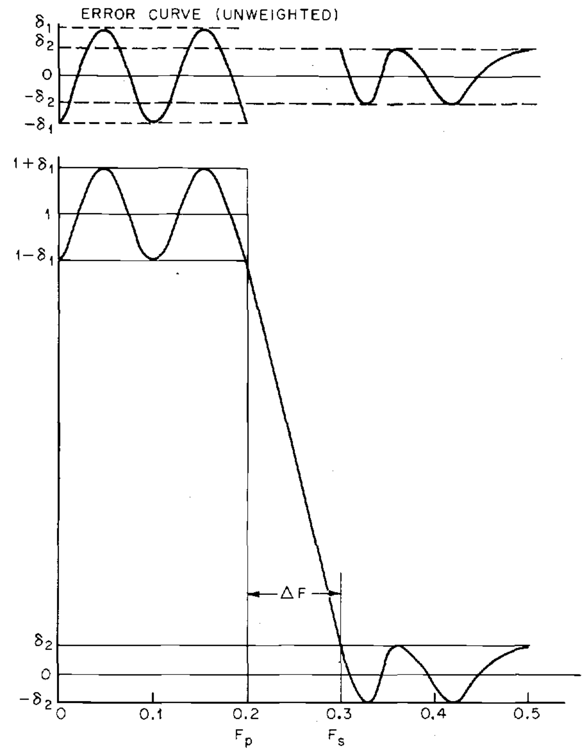

Fig. 1. The frequency response and error curve of an FIR low-pass filter defining, filter parameters F_p, F_s, ΔF, δ_1, and δ_2.

FIR Design Relationships

Based on measurements on an extensive set of optimal, linear phase, low-pass filters, Herrmann *et al.* empirically determined the relationship

$$D = D_\infty(\delta_1, \delta_2) - f(\delta_1, \delta_2)(\Delta F)^2 \tag{3}$$

where

$$D = (N - 1)\Delta F \tag{4}$$

and

$$D_\infty(\delta_1, \delta_2) = [a_1(\log_{10}\delta_1)^2 + a_2\log_{10}\delta_1 + a_3] \cdot \log_{10}\delta_2 + [a_4(\log_{10}\delta_1)^2 + a_5\log_{10}\delta_1 + a_6] \tag{5}$$

with

$$a_1 = 5.309 \times 10^{-3}$$
$$a_2 = 7.114 \times 10^{-2}$$
$$a_3 = -4.761 \times 10^{-1}$$
$$a_4 = -2.66 \times 10^{-3}$$
$$a_5 = -5.941 \times 10^{-1}$$
$$a_6 = -4.278 \times 10^{-1}$$

and

$$f(\delta_1, \delta_2) = b_1 + b_2\log_{10}\delta_1 - b_2\log_{10}\delta_2 \tag{6}$$

with

$$b_1 = 11.01217$$

$$b_2 = 0.51244.$$

The coefficients in (5) and (6) were determined by a minimum mean-square error fitting procedure to the data, whereas the forms for (5) and (6) were suggested by some simple data fitting procedures.

We now consider the original problem—that is, given any four of the five parameters $N, F_p, F_s, \delta_1, \delta_2$, how can the unspecified parameter be estimated, and then adjusted using feedback from the Parks-McClellan algorithm, until specifications on all parameters are met or exceeded. We now consider the five possibilities for the unspecified parameter.

Case 1: $F_p, F_s, \delta_1, \delta_2$ specified—N unspecified. In this case (3) and (4) may be used to give $\hat{N}$, the estimate of N, as

$$\hat{N} = \frac{D_\infty(\delta_1, \delta_2)}{\Delta F} - f(\delta_1, \delta_2)(\Delta F)^2 + 1. \tag{7}$$

Fig. 2 shows the logic required to obtain the actual value of N that is required. After estimating $\hat{N}$ from (7), the direction parameter JD is initialized to 0. The parameters $\hat{N}$, F_p, F_s, and $K = \delta_1/\delta_2$ are used as input to the optimal design algorithm that returns the value $\hat{\delta}_2$ as the actual deviation in the stopband. This value is compared with δ_2 and if they are equal (to within some tolerance) the algorithm is done. If $\hat{\delta}_2 > \delta_2$ then $\hat{N}$ is incremented by 2 (i.e., one filter order) and a check is made to see if the direction parameter JD was -1, indicating that $\hat{N}$ had previously been decreasing. If so, the new value of $\hat{N}$ is the smallest N that meets or exceeds specifications and the algorithm is done. If not, the value of JD is set to 1 and the updated value of $\hat{N}$ is used as input to the optimal design algorithm. A similar path is taken if $\hat{\delta}_2 < \delta_2$ whereby if JD was 1, the current value of $\hat{N}$ is the minimum value of N. Otherwise N is decreased by 2 and JD set to -1 and the algorithm repeats.

It has been found that two to three iterations suffice for a large range of filter parameters.

Case 2: N, F_s, δ_1, and δ_2 specified—F_p unspecified. In this case (3) and (4) are used to give $\hat{F}_p$, the estimate of F_p, as

$$\Delta\hat{F} = F_s - \hat{F}_p = \frac{(N-1)}{2f(\delta_1, \delta_2)}, \left(\sqrt{1 + \frac{4f(\delta_1,\delta_2)D_\infty(\delta_1,\delta_2)}{(N-1)^2}} - 1\right) \tag{8}$$

or

$$\hat{F}_p = F_s - \Delta\hat{F}. \tag{9}$$

Fig. 3 shows a flow chart of the logic required to obtain the correct value of F_p. First $\hat{F}_p$ is calculated

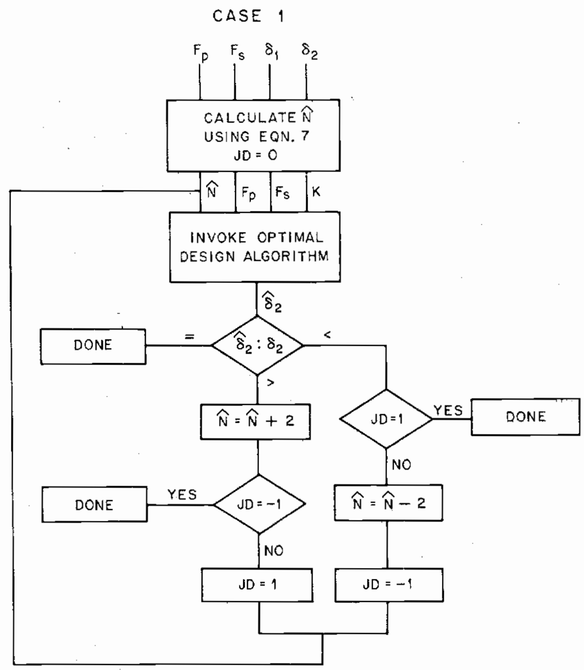

Fig. 2. Algorithm for choosing smallest N for FIR low-pass filter to meet specifications on F_p, F_s, δ_1, and δ_2.

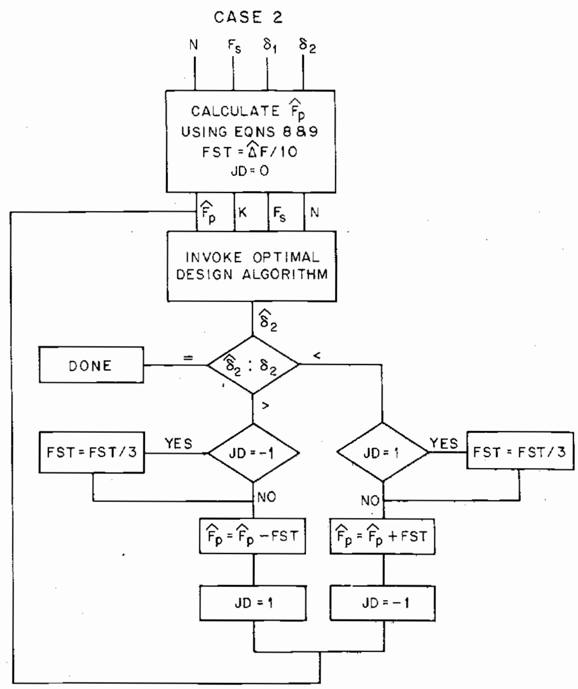

Fig. 3. Algorithm for choosing value of F_p for FIR low-pass filter to meet specifications on N, F_s, δ_1, and δ_2.

and the frequency step size (FST) is set to $\Delta\hat{F}/10$, and the direction parameter JD is set to 0. The initial value $\hat{F}_p$ along with N, F_s, and K are the inputs to the design algorithm that returns the value $\hat{\delta}_2$ as the actual stopband deviation. If $\hat{\delta}_2 = \delta_2$ to within some specified tolerance (typically 1 percent or 0.1 percent), the algorithm is done. If $\hat{\delta}_2 > \delta_2$, and the direction parameter JD is -1, the step size is divided by 3 to take into account the fact that the search has changed direction, and hence significantly narrowed down the interval under investigation. The value of $\hat{F}_p$ is decremented by the step size FST (giving a larger transition width, hence a smaller value of $\hat{\delta}_2$), and the direction parameter is set to 1. If $\hat{\delta}_2 < \delta_2$, a similar path is executed whereby $\hat{F}_p$ is incremented by the current step size.

Case 3: N, F_p, δ_1, and δ_2 specified—F_s unspecified. This case is almost identical to Case 2. Equation (8) is used to estimate ΔF, and $\hat{F}_s$ is obtained as $F_p + \Delta \hat{F}$. The flow chart of Fig. 3 can be used for this case with two simple modifications. First $\hat{F}_p$ and $\hat{F}_s$ are interchanged, and second, for the box where $\hat{F}_p$ was decremented, $\hat{F}_s$ is incremented, whereas for the box where $\hat{F}_p$ was incremented, $\hat{F}_s$ is decremented. The similarity between Cases 2 and 3 should be clear.

Case 4: N, F_p, F_s, δ_1 specified—δ_2 unspecified. A formula for estimating δ_2 may be obtained by rewriting (5) and (6) in the form

$$D_\infty(\delta_1, \delta_2) = c_1 \log_{10} \delta_2 + c_2 \tag{10}$$

$$f(\delta_1, \delta_2) = d_1 - b_2 \log_{10} \delta_2 \tag{11}$$

where

$$c_1 = a_1 (\log_{10} \delta_1)^2 + a_2 \log_{10} \delta_1 + a_3 \tag{12}$$

$$c_2 = a_4 (\log_{10} \delta_1)^2 + a_5 \log_{10} \delta_1 + a_6 \tag{13}$$

$$d_1 = b_1 + b_2 \log_{10} \delta_1 \tag{14}$$

and then solving for $\log_{10} \hat{\delta}_2$, the estimate of $\log_{10} \delta_2$ from (3) as

$$\log_{10} \hat{\delta}_2 = \frac{(N-1)\Delta F + d_1 (\Delta F)^2 - c_2}{c_1 + b_2 (\Delta F)^2} = \alpha \tag{15}$$

or

$$\hat{\delta}_2 = 10^\alpha. \tag{16}$$

The initial estimate $\hat{\delta}_2$ serves to give an initial estimate $\hat{K}$ since δ_2 is not explicitly used as an input parameter. The parameter $\hat{K}$ is then varied until the value $\hat{\delta}_1$ returned by the design algorithm is within a specified tolerance of δ_1, the input specification. Fig. 4 shows a flow chart of how $\hat{K}$ is varied to achieve the desired specifications on δ_1.

First $\hat{\delta}_2$ and $\hat{K} = \delta_1/\hat{\delta}_2$ are computed using (15) and (16); the step size multiplier on K (KM) is set to 2.0; and the direction indicator is initialized to 0. The initial parameters are used in the design algorithm that returns the value $\hat{\delta}_1 = \hat{K}\hat{\delta}_2$ as the actual passband deviation. If $\hat{\delta}_1 = \delta_1$ to within the prescribed tolerance, the algorithm terminates. If $\hat{\delta}_1 > \delta_1$, the estimate K is decreased by the factor $(1 + KM)$ where KM is the current step size multiplier. If a change in direction was detected, the step size multiplier is halved.

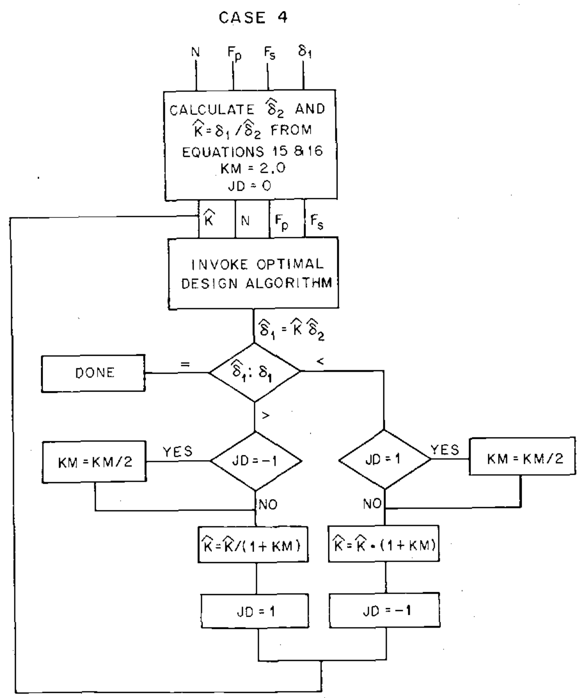

Fig. 4. Algorithm for choosing value of K for FIR low-pass filter to meet specifications on N, F_p, F_s, and δ_1.

The direction indicator is then set to 1. If $\hat{\delta}_1 < \delta_1$, a similar path is executed with $\hat{K}$ being increased by the factor $(1 + KM)$. The new estimate $\hat{K}$ is used as updated information for the design algorithm.

Case 5: N, F_p, F_s, δ_2 specified—δ_1 unspecified. A formula for estimating δ_1 may be obtained by rewriting (5) and (6) as

$$D_\infty(\delta_1, \delta_2) = e_1 (\log_{10} \delta_1)^2 + e_2 (\log_{10} \delta_1) + e_3 \tag{17}$$

$$f(\delta_1, \delta_2) = g_1 + b_2 \log_{10} \delta_1 \tag{18}$$

where

$$e_1 = a_1 \log_{10} \delta_2 + a_4 \tag{19}$$

$$e_2 = a_2 \log_{10} \delta_2 + a_5 \tag{20}$$

$$e_3 = a_3 \log_{10} \delta_2 + a_6 \tag{21}$$

$$g_1 = b_1 - b_2 \log_{10} \delta_2. \tag{22}$$

Equations (3) and (4) can be used to give a quadratic equation in $\log_{10} \delta_1$, i.e.,

$$(\log_{10} \delta_1)^2 + g_2 \log_{10} \delta_1 + g_3 = 0 \tag{23}$$

where g_2 and g_3 are defined as

$$g_2 = \left(\frac{e_2 - b_2 (\Delta F)^2}{e_1}\right) \tag{24}$$

$$g_3 = \frac{e_3 - g_1 (\Delta F)^2 - (N-1)\Delta F}{e_1}. \tag{25}$$

Solving for $\log_{10} \hat{\delta}_1$, the estimate of $\log_{10} \delta_1$, we get

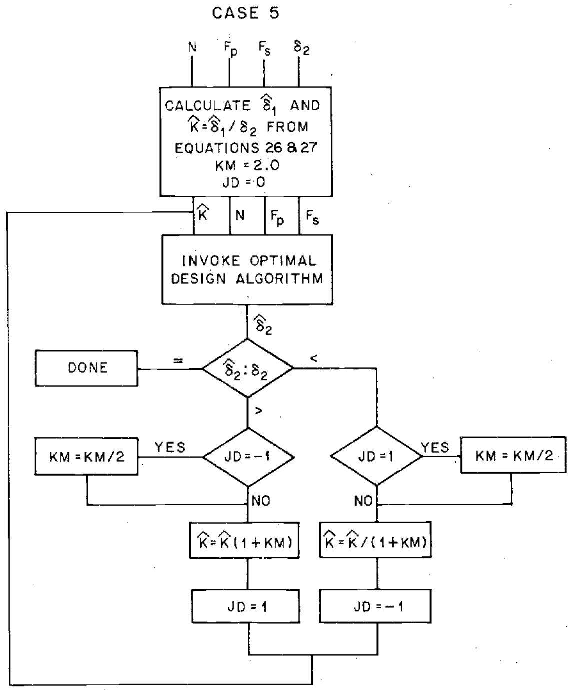

Fig. 5. Algorithm for choosing value of K for FIR low-pass filter to meet specifications on N, F_p, F_s, and δ_2.

$$\log_{10} \hat{\delta}_1 = \frac{-g_2}{2} + \sqrt{\frac{g_2^2}{4} - g_3} = \beta \tag{26}$$

or

$$\hat{\delta}_1 = 10^{\beta}. \tag{27}$$

(We have used the positive sign for the square root because the negative value leads to unrealistically small values of $\hat{\delta}_1$.)

From the estimate $\hat{\delta}_1$, an estimate $\hat{K} = \hat{\delta}_1/\delta_2$ is obtained. The flow diagram of Fig. 5 shows how $\hat{K}$ is adjusted until the filter specifications are achieved. Since this flow chart is so similar to the one of Fig. 4, we will not discuss the details.

Example of Use of Design Formulas

To illustrate how the above formulas are used, the parameters $N = 19$, $F_p = 0.14$, $F_s = 0.3182422$, $\delta_1 = 0.01$, and $\delta_2 = 0.0001$ were used as input to the preceding design rules with one of the parameters left unspecified. (These parameters are those for an optimal filter which had previously been designed.)

For Case 1, N was unspecified. The initial estimate $\hat{N}$ was 17. Table I shows values for $\hat{N}$, δ_2, and JD for the two iterations that were required. In this case, for a 1.0 percent accuracy criterion the value $N = 19$ was selected. The total run time on a Honeywell 6000 computer was 0.20 s.

For Case 2, F_p was unspecified. The initial estimate $\hat{F}_p$ was 0.1605246. Table II shows values for $\hat{F}_p$, $\hat{\delta}_2$, FST, and JD for the five iterations that were required to get $\hat{\delta}_2$ to within 1.0 percent tolerance. The total run time was 0.47 s. The final value of F_p was 0.1394956 as opposed to the exact value of 0.140 for which $\delta_2 = 0.0001$.

For Case 3, F_s was unspecified. The initial estimate $\hat{F}_s$ was 0.2977176. In this case ten iterations were required to get $\hat{\delta}_2$ to within 1.0 percent tolerance. The run time here was 0.95 s. The final value of F_s was 0.3181625 for which δ_2 was 0.000099.

For Case 4, δ_2 was unspecified. The initial estimate $\hat{\delta}_2$ was 0.0000166 or $\hat{K}$ was 601. Seven iterations were required to get δ_1 to within 1.0 percent of δ_1. Table III shows values of $\hat{\delta}_1$, $\hat{K}$, KM, and JD for the seven iterations. The run time was approximately 0.63 s.

TABLE I

Iteration	$\hat{N}$	$\hat{\delta}_2$	JD
1	17	0.0001931	0
2	19	0.0001001	1

TABLE II

Iteration	$\hat{F}_p$	δ_2	FST	JD
1	0.1605246	0.000151	0.0157718	0
2	0.1447529	0.000110	0.0157718	1
3	0.1289811	0.000078	0.0157718	1
4	0.1342384	0.000088	0.0052573	−1
5	0.1394956	0.000099	0.0052573	−1

TABLE III

Iteration	$\hat{K}$	$\hat{\delta}_1$	KM	JD
1	601.0	0.025003	2.0	0
2	200.4	0.015567	2.0	1
3	66.8	0.007426	2.0	1
4	133.6	0.012181	1.0	-1
5	89.0	0.009207	0.5	1
6	111.3	0.010786	0.25	-1
7	98.9	0.009933	0.125	1

Finally, for Case 5, δ_1 was unspecified. The initial estimate $\hat{\delta}_1$ was 0.002655 or $\hat{K} = 26.55$. It required ten iterations for $\hat{\delta}_2$ to be within 1 percent of δ_2. The run time for this case was about 1.0 s.

Summary

An optimal FIR low-pass filter can now be designed where any four of the five parameters N, F_p, F_s, δ_1, and δ_2 are specified, and the remaining parameter is chosen so as to meet or exceed specifications on all given parameters. A set of simple, approximate formulas was given for obtaining initial estimates of the unspecified parameter. Finally, simple iterative rules were given for varying the unspecified parameter from its initial estimate so as to meet input specifications to within a given tolerance.

References

[1] J. F. Kaiser, "Design methods for sampled data filters," in *Proc. 1st Allerton Conf. Circuit and System Theory*, 1963, pp. 221-236.
[2] L. R. Rabiner, "Techniques for designing finite-duration impulse-response digital filters," *IEEE Trans. Commun. Technol.*, vol. COM-19, pp. 188-195, Apr. 1971.
[3] O. Herrmann, "Design of nonrecursive digital filters with linear phase," *Electron. Lett.*, vol. 6, no. 11, pp. 328-329, 1970.
[4] T. W. Parks and J. H. McClellan, "Chebyshev approximation for nonrecursive digital filters with linear phase," *IEEE Trans. Circuit Theory*, vol. CT-19, pp. 189-194, Mar. 1972.
[5] L. R. Rabiner, "The design of finite impulse response digital filters using linear programming techniques," *Bell Syst. Tech. J.*, vol. 51, pp. 1177-1198, July-Aug. 1972.
[6] T. W. Parks and J. H. McClellan, "A program for the design of linear phase finite impulse response digital filters," *IEEE Trans. Audio Electroacoust.*, vol. AU-20, pp. 195-199, Aug. 1972.
[7] O. Herrmann, L. R. Rabiner, and D. S. K. Chan, "Practical design rules for optimum finite impulse response low-pass digital filters," *Bell Syst. Tech. J.*, vol. 52, no. 6, pp. 769-799, July-Aug. 1973.

Least *p*th Optimization of Recursive Digital Filters

JOHN W. BANDLER and BERJ L. BARDAKJIAN

Abstract—The application of the Bandler-Charalambous method using extremely large values of *p*, typically 10 000 to the problem of choosing the coefficients of a recursive digital filter to meet arbitrary specifications on the magnitude characteristics, is described. The Fletcher (1970) method is used in conjunction with least *p*th optimization and is compared with the well-known Fletcher-Powell method. Some relevant design ideas, such as local optimality checking by perturbation, increasing the order complexity of the filter through growing filter sections, and meeting the stability requirements by using a pole inversion technique, have been implemented. A general description of a computer program package that uses these ideas, along with some illustrative examples are given.

Manuscript received November 10, 1972; revised May 7, 1973. This work was supported by the National Research Council of Canada under Grants A7239 and C154. This paper was presented at the 1973 International Symposium on Circuit Theory, Toronto, Canada, April 9-11, 1973.
The authors are with the Department of Electrical Engineering, McMaster University, Hamilton, Ont., Canada.

I. Introduction

Two main approaches have been taken to approximation problems in digital filter design. The first of these is an analytical approach through classical approximation theories [1]-[3]. The second is an iterative approach that is particularly appropriate for use on a digital computer [4]-[6]. Sablatash [7] discussed many contributions to both approaches.

Haykin [3] presented a unified treatment of recursive digital filtering by using the convolution integral to derive an integro-difference equation for defining the input-output relation of a linear time invariant filter. Then he used that equation to obtain various analog-to-digital filter transformations for the digitization of a continuous transfer function, with each transformation corresponding to a specific way of approximating the continuous time excitation.

An iterative method for designing recursive digital filters with arbitrary prescribed magnitude characteristics was described by Steiglitz [4]. The method uses the Fletcher-Powell algorithm [8] to minimize a square-error criterion in the frequency domain. A strategy was described whereby stability and minimum phase constraints were observed, while still using the unconstrained optimization algorithm.

Helms [5] reviewed and occasionally extended tech-