# What is the Filter Design Problem?†

**K. Steiglitz**

Dept. of Computer Science
Princeton University
Princeton, New Jersey 08544

**T. W. Parks**

Dept of ECE
Rice University
Houston, Texas 77251

## ABSTRACT

The usual way of designing a filter is to specify a filter order and a nominal response, and then to find a filter of that order which best approximates that response. In this paper we propose a different approach: specify the filter only in terms of upper and lower limits on the response, find the lowest order which allows these constraints to be met, and then find a filter of that order which is farthest from the upper and lower constraint boundaries in a mini-max sense.

Previous papers have described methods for using an exchange algorithm for finding a feasible linear-phase FIR filter of a given order if one exists, given upper and lower bounds on its magnitude response. The resulting filters touch the constraint boundaries at many points, however, and are not good final designs because they do not make best use of the degrees of freedom in the coefficients. We will use the simplex algorithm for linear programming to find a best linear-phase FIR filter of least order, as well as to find the least feasible order itself. The simplex algorithm, while much slower than exchange algorithms, also allows us to incorporate more general kinds of constraints, such as convexity constraints (which can be used to achieve very flat magnitude characteristics).

We will give examples that illustrate how the proposed and the usual approaches differ, and how the new approach can be used to control peaks in transition bands of multi-band filters.

## 1. Introduction

There are two fundamentally different approaches to the FIR linear phase filter design problem, the approximation approach and the limit approach. In the approximation approach, the length of the filter and a desired frequency response are specified. The filter coefficients are determined to minimize the maximum weighted error between the desired and actual responses over the frequency bands of interest. In the limit approach, a set of upper and lower limits are specified for the frequency response. The values of filter coefficients for which the frequency response remains within the prescribed limits are then determined.

In 1970, Herrmann published an article describing the equations which must be solved to obtain a filter with the maximum possible number of equal ripples [1] (later called extra-ripple [2] or maximal-ripple [3] filters). This maximal ripple design is neither an approximation approach nor a limit approach. Rather, it is a hybrid approach where the filter length and ripple size (equivalent to limits on the frequency response) are specified and the band edges are determined by the algorithm. Schuessler, in 1970, presented the work he and Herrmann had been doing on the design of maximal-ripple filters at the Arden House Workshop [4]. Hofstetter developed an efficient algorithm for solving the equations proposed by Herrmann and Schuessler and presented papers with Oppenheim and Siegel at the 1971 Princeton conference [5] and the 1971 Allerton House conference [6] describing the algorithm and relating it to the Remes exchange algorithm.

Several papers on the Chebyshev approximation approach to filter design appeared at about the same time. Helms, in 1971 [7], described techniques, including linear programming, to solve the Chebyshev approximation problem for filter design. Parks and McClellan used the Remes exchange algorithm [8,9] to solve the Chebyshev approximation problem.

Hersey, Tufts, and Lewis described, at about the same time, an interactive method for designing filters with upper and lower constraints on the magnitude of the frequency response [10]. The limit approach was also used by McCallig and Leon in 1978 [11] and by Grenez in 1983 [12]

When a lowpass filter is designed using the Chebyshev approximation approach, the 5 interrelated parameters are the filter length, the passband edge Fp, the stopband edge Fs, the passband error delp, and the stopband error dels. Relations among these parameters have been determined numerically for the Chebyshev approximation problem and design formulas have been published [13]. With the help of these design formulas it is possible to fix any 4 of these parameters and optimize the remaining parameter. Since these design formulas are not exact, several iterations of the design process are usually necessary. For example, when the band edges and deviations are given, an estimate of the necessary filter length can be calculated using the design formulas. Usually the filter with this estimated length will not be exactly the minimum length required to meet the specifications and the the filter will be designed again with a slightly different length until the

† This work was supported in part by NSF Grants ECS-8414674 and ECS 83-14006 and U.S. Army Research-Durham Contract DAAG29-85-K-0191.

minimum length filter is obtained.

The use of transition bands will give good lowpass designs but may cause problems for multiband bandpass filters [14]. The frequency response is not controlled in the transition band and may make large, unexpected, excursions which make the design useless. The design formulas can be used to modify the stopband specifications to eliminate the unwanted excursions in most cases, but the choice of stopband edges and appropriate error weighting functions is more of an art than a science. The limit approach offers a way to avoid unwanted excursions in multiband filter design. Upper and lower limits are imposed on the response for all frequencies. The limits imposed on the bands which otherwise would be unrestricted transition bands eliminate the possibility of large peaks in the magnitude of the frequency response, but do not impose any particular shape on the response in these bands.

In this paper we describe a very flexible design program which combines most of the useful characteristics of the approximation approach and the limit approach to FIR filter design. We use the simplex algorithm for linear programming to find the linear phase filter of least order which meets prescribed limits on the frequency response and then maximize the distance from the constraints. For a fixed order filter, the bandedges can be adjusted to maximize or minimize the width of a frequency band while still meeting prescribed limits on the frequency response. Additional constraints, such as convexity of the response to give flat magnitude characteristics, can be imposed in appropriate frequency bands. First, we describe the algorithm and the Pascal program and then we give examples to show how this new approach can be used to control excursions in the transition band of a bandpass filter.

## 2. The Algorithm

For the purposes of this discussion we will assume that the filter model is the following sum of cosines, corresponding to an odd-order symmetric impulse response, although any linear combination of known functions can be used.

$$H(k) = \sum_{i=0}^{m-1} a_i \cos(i\,\omega_k)$$

$H(k)$ is the real-valued frequency response of the filter at frequency $\omega_k$, and the frequency points at which specifications are made, $\omega_k$, $k = 1, 2, 3, \ldots$, need not be equally spaced.

An upper-limit constraint at $\omega_k$ has the form

$$H(k) \le U(k) .$$

We will introduce a parameter $y$ which represents the distance between the frequency response and the upper bound, so that some of the constraints will look like

$$H(k) + y \le U(k) .$$

Since we will be maximizing $y$, we will call those constraints which have $y$ in them *optimized* constraints, and those that do not, *hugged* constraints. Similarly, lower bounds on the frequency response will result in constraints of the form

$$H(k) \ge L(k)$$

or

$$H(k) - y \ge L(k) ,$$

depending on whether the constraint is optimized or hugged.

Putting constraints on the second derivative of the frequency response has been shown to be an effective way to get filters that are very flat [1]. The second derivative is a linear function of the coefficients, namely,

$$H''(k) = -\sum_{i=1}^{m-1} i^2 a_i \cos(i\,\omega_k) ,$$

so that convexity constraints can be written as linear inequalities of the form

$$H''(k) \le 0 \quad \text{or} \quad H''(k) \ge 0 .$$

When all the constraints are written down, we get the linear programming problem

$$\max y \qquad \text{(PRIMAL)}$$

subject to

$$A^T a + h\, y \le b$$

where the vector $h$ has a 1 wherever a constraint is optimized, and a 0 wherever it is hugged. The variables $a$ and $y$ are unconstrained in sign. We will call this the *PRIMAL* problem. The dual of this linear program is in standard form, the most convenient for numerical solution:

$$\min b^T x \qquad \text{(DUAL)}$$

subject to

$$Ax = 0 , \quad h^T x = 1 , \text{ and } \quad x \ge 0 .$$

We will solve DUAL using the standard two-phase simplex algorithm [2]. Phase I searches for a feasible solution to DUAL, starting from an artificial basis, and phase II searches for an optimal solution.

It is a fundamental fact of linear programming theory that the cost function of the DUAL always satisfies $b^T x \ge y$, the cost function of the PRIMAL, with equality if and only if $x$ and $y$ are both optimal in their respective programs. Therefore, if the DUAL cost $b^T x$ ever falls below zero during pivoting, the optimal PRIMAL cost must be negative. This means that the original filter approximation problem is infeasible, and we stop the simplex algorithm whenever this condition is obtained. Application of the simplex algorithm to the DUAL problem therefore terminates in one of the following conditions:

a) Negative cost reached, implying that the original design problem is infeasible;

b) Optimality is reached in DUAL with non-negative cost, in which case the original design problem has a feasible solution;

c) DUAL is unbounded, which implies that PRIMAL (and the original design problem) is infeasible;

d) DUAL is infeasible, which implies that PRIMAL (and the original design problem) is either infeasible or unbounded.

A comment is in order as to why the variable $y$ is introduced in those situations when we are interested only in whether there is a feasible solution to lower- and upper-bound constraints. Computational experience has shown that with a trivial cost function in the primal, the simplex method applied to the dual sometimes cycles in realistic filter-design problems, because of degeneracy. A non-trivial cost function seems to provide enough direction to the simplex algorithm to avoid such stagnation. Rather than take special precautions to avoid cycling, we chose always to maximize the distance $y$ from the response to the constraint boundaries. (As we saw above, it is not always necessary to complete the optimization when the original problem is infeasible.) This has the additional advantage of being useful for the final design when the order is known, and also does not interfere with the resolution of ties based on size of the pivot elements, which is important for numerical stability (see [3]).

A special case arises unavoidably, however, when there are no constraints designated as *hugged*. In that case, $h = 0$ and DUAL is always infeasible. However, the constraint matrix of DUAL in this case is not of full rank, having a zero row, and phase I ends with an artificial basis element remaining in the basis. The redundant row is disregarded in phase II, and the optimization finds a solution to the original problem (if any exist) with zero cost, corresponding to a response that is allowed to touch any of the constraint boundaries. Thus, the algorithm functions in a useful way, even if a zero row is present in the DUAL constraint matrix.

The optimal value of the dual variable $x$ has a well-known and interesting interpretation. Suppose the constraint values $b$ are changed a small amount to $b + db$. This changes the cost function in the dual a small amount, but will not in general change the optimal solution $x$ to the dual. The new value of the optimal cost function becomes $y = b^T x + db^T x$. Thus, $x$ is the partial derivative of the optimal value of $y$ with respect to the constraint values $b$. Simplex finds an optimal value for $x$ that has at most $m$+1 positive entries, and, by complementary slackness, each of these corresponds to an extremum of the distance between the frequency response and constraints (a "ripple") in the case of an upper or lower bound, or to a point where the second derivative is zero in the case of a convexity constraint.

The simplex algorithm is used in the following three modes, depending on what design task is desired:

a) Given $m_1 < m_2$, find the minimum order $m$ between them such that the original design problem is feasible (that is, such that DUAL has a non-negative optimal solution), and optimize $y$ for that minimum order;

b) Solve the original optimization problem for fixed order $m$ ;

c) Given a particular right (left) bandedge and a set of constraints in which it occurs, find the largest (smallest) value for that bandedge for which the original design problem is feasible, and optimize $y$ for that bandedge value. (The optimum value of $y$ will in general be positive because the bandedge value is rounded to the nearest gridpoint.)

What is the best search strategy to use in finding the minimum order in a)? We might expect, because the cost of testing feasibility increases with $m$, that the strategy with least expected cost (assuming uniformly distributed answers) probes to the left of the midpoint between the current left and right boundaries. However, computation of the optimal strategies for probe-cost functions that grow as a low-order polynomial in $m$ shows that binary search is surprisingly near optimal. More work on this problem is in progress, but binary search appears adequate for this application. Mode b) allows us to do things like find the best stopband rejection, while keeping passband ripple within limits. Mode c) allows us to do things like extend the end of a stopband as far as possible, while keeping the other constraints fixed. Binary search is also used in c).

### 3. The Program

The algorithm described above was implemented in Pascal, and the current version is available from the authors. The authors' intent is that the program be read and modified by users, rather than used as a static package, and Pascal seems well suited to this purpose: it is widely available, cleanly designed, allows careful structuring, and hopefully, good readability. For example, rather than code options for odd- and even-order symmetric and anti-symmetric filter models, the user need only change trigonometric expressions at three points in the program.

As might be expected, the critical parts of the program involve the treatment of tests which theoretically determine whether quantities are positive, negative, or zero. These tests determine when each of the various termination conditions is reached, and roundoff error requires us to decide on how small a positive number is considered zero, how small a negative number is considered negative, and so on. Experience has shown that a single parameter *eps* can be used for these tests at several different places in the program, and that *eps* can be fixed at $10^{-8}$ for the range of problems used as examples in this paper.

The only cases observed so far where serious accumulation of roundoff error occurs is when a wide band of frequencies is unconstrained, and the frequency response is allowed to grow very large in those bands — say as large as $10^6$. The problem is manifested by the cost in phase I reaching relatively large negative numbers before detecting optimality, even though the cost in phase I is theoretically non-negative. Of course, these designs are impractical, and the accuracy problem irrelevant, but the program continues to function in these cases.

Trading off space for running time is a serious issue in the program design. At one extreme, we can pre-compute and store the tableau entries, which avoids re-computation, but uses a great deal of storage. At the other extreme, we can generate the tableau entries on the fly, using the least space, but the most time. As a compromise between the two, we can pre-compute and store tables of the trigonometric functions used for the tableau entries. We chose the first alternative because it appears that execution time is a more serious limitation than storage for the kinds of design problems likely to be solved. If storage is a serious problem, references to the tableau entries must be replaced by procedure calls that compute the required values.

## 4. Bandpass Example

We illustrate some of the features of the new algorithm with a bandpass example. The specifications for a bandpass filter are given below:

| | |
|---|---|
| Frequencies for Stopband 1 | 0.00 to 0.08 |
| Frequencies for Passband | 0.25 to 0.37 |
| Frequencies for Stopband 2 | 0.40 to 0.50 |
| Maximum Passband Deviation | 0.1 |
| Minimum Stopband Attenuation | 20 dB |

The problem is to find the minimum length linear phase filter which meets these specifications and has well-behaved transition bands. First, the Chebyshev approximation approach is shown to produce a large peak in the first transition band. Then then the new algorithm is used with limits on the response in the first transition band to eliminate the transition band peak, and finally the first stopband is widened, thus reducing the transition width and eliminating the peak in the transition band.

The first step in the approximation approach is to estimate the necessary length using the simplified design formula [3]

$$N = \frac{dB - 13}{14.6\,\delta F} + 1 \ .$$

where $\delta F$ is the the transition band width and dB is the stopband attenuation in this special case where the error weight is the same for all bands. While this formula wasn't intended for bandpass filters, it sometimes gives reasonable estimates when the width of the smallest transition band is used for $\delta F$. For this example, the estimated value of N from this formula is 17. In fact, after a few iterations, it was found that a length of 25 was necessary to meet these specifications. The frequency response for this design is shown in Figure 1. On the scale of Figure 1, there appears to be a problem in the transition band, but in the bands where the Chebyshev error was minimized, the response looks good. In Figure 2, the frequency response is shown on a different scale to show the extent of the peak in the first transition band.

To illustrate how the new algorithm can give about the same response as the approximation approach, the filter was designed requiring the response to be within limits of +/- 0.1 in the stopbands and within the upper and lower limits of 1.1 and 0.9 in the passband. The minimum filter length was found to be 25. In the passband the distance from the constraints was maximized, using up any slack resulting from the fact that the filter length must be an integer. The response, as shown in Figure 3, still has the undesirable transition peak.

To eliminate the transition band peak, additional limits were placed on the response in the first transition band. The new algorithm found that the filter length must be increased to N = 27 in order to meet these new, stricter, limits. The response of this length 27 filter is shown in Figure 4 along with the imposed limit on the transition band. As in Figure 3, the distance from the constraints is maximized in the passband.

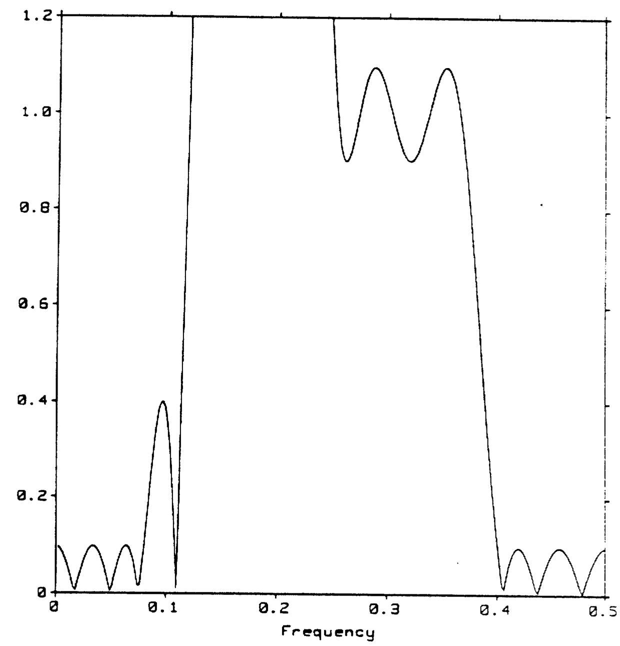


Fig. 1. Frequency Response: Chebyshev Approximation N=25.

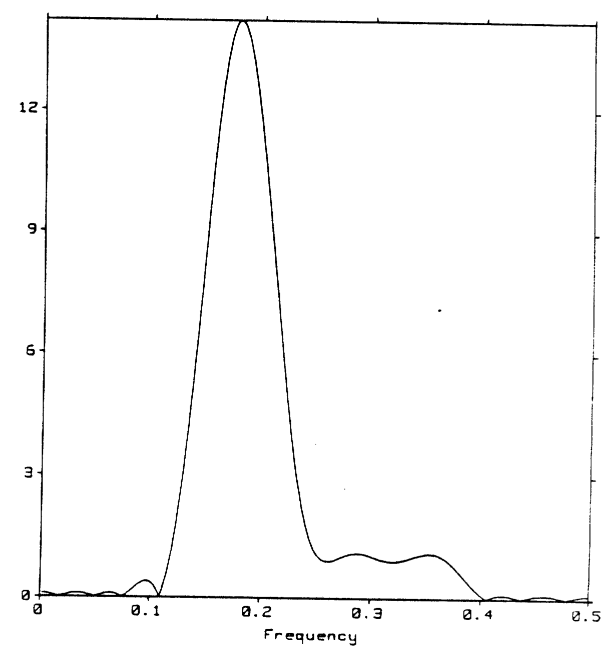


Fig. 2. Transistion band peak of Figure 1.

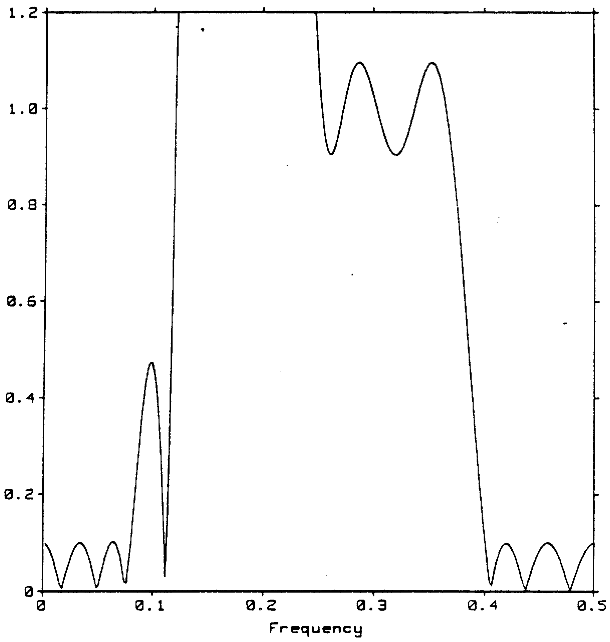


Fig. 3. Frequency Response for Limit Design, N = 25.

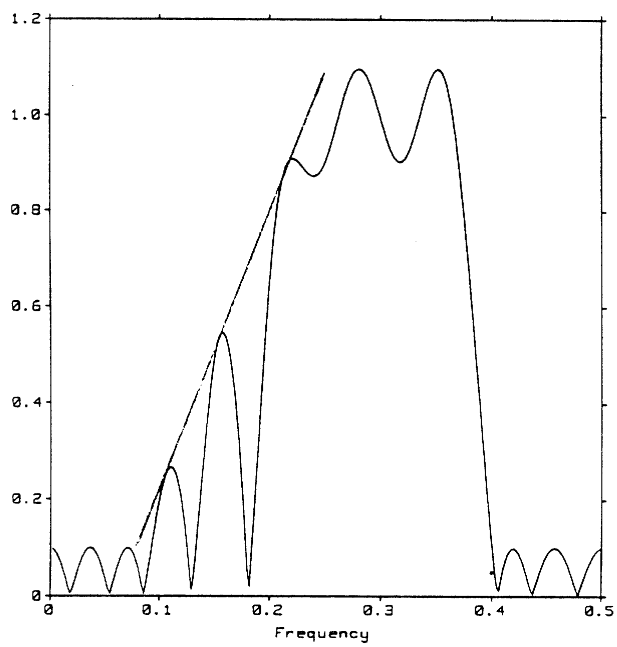


Fig. 4. Frequency Response with Transition Limit, N = 27.

Another alternative for eliminating the transition band peak is to fix the length at N = 27 and push the upper edge of the lower stopband to the right, maximizing the width of the first stopband, thus reducing the width of the first transition band and eliminating the transition band peak. The file of input parameters for this case is shown in Figure 5 and the resulting frequency response is shown in Figure 6.

## 5. Conclusion

Several approaches to the FIR linear-phase frequency-domain filter design problem were considered. A new algorithm, using the simplex method of linear programming, was proposed which is very general and can incorporate a wide variety of constraints on the frequency response of the filter. An application of this new algorithm to the problem of eliminating transition peaks in bandpass filters was given. There are many other types of constraints which may be incorporated depending on the answer to the question, "What is the filter design problem?"

```
14 14         min and max order: the same, so order is fixed.
r             bandedge is pushed to the right
2             number of constraints to be pushed
1 2           which constraints to push
100           no. of grid intervals
limit         type of constraint
+             upper limit
arithmetic    interpolation between left and right endpoints
hug           means that distance to this constraint is not optimized
0 .08         left and right band edges
.1 .1         upper limits at left and right edges
limit         type of constraint
-             lower limit
arithmetic    interpolation between left and right endpoints
hug           means that distance to this constraint is not optimized
0 .08         left and right band edges
-.1 -.1       lower limits at left and right edges
limit         type of constraint
+             upper limit
arithmetic    interpolation between left and right endpoints
optimize      distance to this constraint is optimized
.25 .37       left and right band edges
1.1 1.1       upper limit at left and right edges
limit
-
arithmetic
optimize          repeat for fourth constraint
.25 .37
.9 .9
limit
+
arithmetic
hug               repeat for fifth constraint
.4 .5
.1 .1
limit
-
arithmetic
hug               repeat for sixth constraint
.4 .5
-.1 -.1
end
```

Fig. 5. Parameters for Pushed Bandedge, N=27.

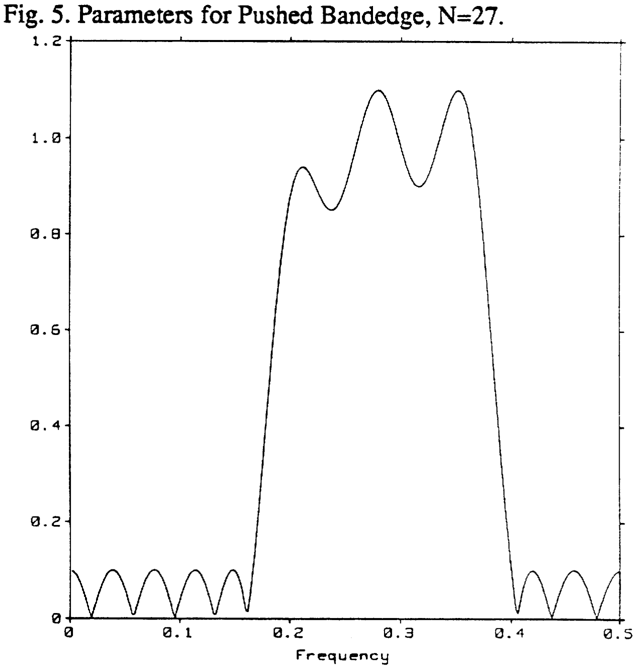


Fig. 6. Frequency Response for Pushed Bandedge, N=27.

## 6. References


[1] O. Herrmann, "Design of Nonrecursive Digital Filters With Linear Phase," *Electronics Letters*, Vol. 6, pp.328-329, May 28, 1970.

[2] T.W. Parks, L.R. Rabiner, and J.H. McClellan, "On the Transition Width of Finite Impulse-response Digital Filters," *IEEE Trans. Audio and Electroacoustics*, Vol. AU-21, pp.1-4, February, 1973.

[3] L.R. Rabiner, J.H. McClellan, and T.W. Parks, "FIR Digital Filter Design Techniques using Weighted Chebyshev Approximation," *Proceedings of the IEEE*, Vol. 63, no. 4, pp.595-610, April 1975.

[4] O. Herrmann and H.W. Schuessler, "On the Design of Selective Nonrecursive Digital Filters" presented at the IEEE Arden House Workshop on Digital Filtering, January 12, 1970.

[5] E.M. Hofstetter, A.V. Oppenheim, and J. Siegel, "A New Technique For the Design of Non-recursive Digital Filters" Proc. Fifth Annual Princeton Conference on Information Sciences and Systems, pp.64-72, March 1971.

[6] E.M. Hofstetter, A.V. Oppenheim, and J. Siegel, "On Optimum Nonrecursive Digital Filters" Proc. Ninth Annual Allerton Conference on Circuit and System Theory, pp.789-798, October 1971.

[7] H.D. Helms, "Digital Filters with Equiripple or Minimax Responses," *IEEE Trans. Audio and Electroacoustics*, Vol. AU-19, pp.87-94, March 1971.

[8] T.W. Parks, "Extensions of Chebyshev Approximation for Finite Impulse Response Filters," presented at the IEEE Arden House Workshop on Digital Filtering, January 10, 1972.

[9] 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, March 1972

[10] H.S. Hersey, D.W. Tufts, and J.T. Lewis, "Interactive Minimax Design of Linear Phase Nonrecursive Digital Filters Subject to Upper and Lower Function Constraints," *IEEE Trans. Audio and Electroacoustics*, Vol. AU-20, pp.171-173, June 1972.

[11] M.T. McCallig and B.J. Leon, "Constrained Ripple Design of FIR Digital Filters," *IEEE Trans. Circuits and Systems*, Vol. CAS-25, pp.893-902, November, 1978.

[12] F. Grenez, "Constrained Chebyshev Approximation for FIR Filters," ICASSP-83, Boston, MA, pp 194-196.

[13] 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, pp. 769-799, July-Aug.1973.

[14] L.R. Rabiner, J.F.Kaiser, and R.W. Schafer, "Some Considerations in the Design of Multiband Finite-Impulse-Response Digital Filters," *IEEE Trans. Audio and Electroacoustics*, Vol. ASSP-22, pp.462,472, December 1974.

[15] J.F. Kaiser, K. Steiglitz, "Design of FIR Filters with Flatness Constraints," *Proc. 1983 IEEE Int. on Acoustics, Speech, and Signal Processing*, Boston, Mass., pp.197-200, April 14-16, 1983.

[16] C.H. Papadimitriou, K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity* Prentice-Hall, Englewood Cliffs, New Jersey, 1982.

[17] K. Steiglitz, "Optimal Design of FIR Digital Filters with Monotone Passband Response," *IEEE Trans. on Acoustics, Speech, and Signal Processing*, Vol. ASSP-27, no. 6, pp.643-649, December 1979.