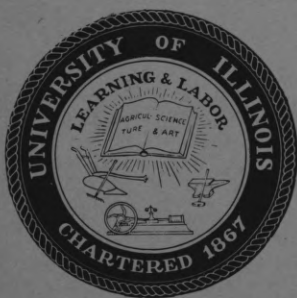




Coordinated
Science
Laboratory



UNIVERSITY OF ILLINOIS - URBANA, ILLINOIS

**A NUMERICAL APPROXIMATION
TECHNIQUE FOR FILTER FUNCTIONS**

William Frederick McGee

REPORT R-151

AUGUST, 1962

**COORDINATED SCIENCE LABORATORY
UNIVERSITY OF ILLINOIS
URBANA, ILLINOIS**

**Contract DA-36-039-SC-85122
D/A Sub-Task 3-99-01-002**

The research reported in this document was made possible by support extended to the University of Illinois, Coordinated Science Laboratory, jointly by the Department of the Army (Signal Corps), Department of the Navy (Office of Naval Research), and the Department of the Air Force (Office of Scientific Research) under Signal Corps Contract DA-36-039-SC-85122.

A NUMERICAL APPROXIMATION TECHNIQUE FOR FILTER FUNCTIONS

William Frederick McGee, Ph.D.

Department of Electrical Engineering

University of Illinois, 1962.

The author proposes a new method of generating amplitude-frequency filter transfer functions of the usual type. The method consists of two algorithms based on modifications of the second method of Remez. Using this method it is shown how to generate the following types of filter functions -- the low-pass equi-ripple polynomial filter function with zeros at the origin; the low pass equi-ripple filter function with a variable number of zeros in the pass-band and a variable number of poles in the stop-band; the asymmetrical equi-ripple band-pass filter function; and the double band-pass equi-ripple filter function.

The parameters that must be specified are discussed. Modifications to the basic methods that make them more useful to the filter designer are also presented.

A discussion is given of the convergence of the proposed algorithms. Examples are given.

It is expected that these methods will be useful in filter design.

ACKNOWLEDGEMENT

The author is indebted to his adviser, Professor S. Seshu for advice given throughout the research reported here. The author also wishes to acknowledge the help received in conversations with Dr. Jean Descloux, Mr. G.R. Lang, and Mr. M.L. Blostein.

The author wishes to express his gratitude to the Coordinated Sciences Laboratory of the University of Illinois for the use of the 1604 computer, to the Radio Corporation of America for a fellowship for the period covering this study, and to the University of Illinois Library for bibliographic assistance.

TABLE OF CONTENTS

	<u>Page</u>
CHAPTER 1	
Filter Design and the Approximation Problem	1
CHAPTER 2	
Some Earlier Work	8
CHAPTER 3	
Algorithm I	15
CHAPTER 4	
Algorithm II	25
CHAPTER 5	
Examples and Results	32
CHAPTER 6	
Advantages and Disadvantages of These Methods	59
CHAPTER 7	
Conclusion	60
Bibliography	61

CHAPTER 1

FILTER DESIGN AND THE APPROXIMATION PROBLEM

In certain branches of Electrical Engineering it is useful to think of an electrical signal as a sum of sine wave signals. An electrical 'filter' is a network that modifies a signal by attenuating (filtering out) certain frequencies. This modification is often represented by a transfer function $H(w) = U(w) + j V(w)$, which is, typically, the output signal transform divided by the input signal transform. The engineer synthesizing a filter must prescribe a transfer function that is realizable with the elements at his disposal, and that meets a set of specifications describing the filtering action desired. The determination of the transfer function $H(w)$ is the problem before us; it is called the 'filter approximation problem'. Specifications which are often met are described in this chapter; the determination of some H 's is described in the later chapters.

The engineer usually incorporates into his specification suitable phase response, impulse response, group delay, amplitude-frequency response, and others. Only the amplitude-frequency response $|H(w)|^2$ will concern us.

A filter composed of a finite number of linear elements has an amplitude response $|H|^2$ that is a rational function of the frequency variable w . We call a zero or pole of a rational function a 'critical point'. Attention is focused on amplitude-frequency functions of the form

$$|H(\omega)|^2 = \frac{1}{1 + \epsilon^2 f^2(\omega)}$$

We call f a 'filter function'. The parameter ϵ is called the 'ripple factor'. The introduction of the ripple factor permits normalization of the filter function. The function f must be even or odd⁽⁷⁾. Transfer functions of this type are always realizable as doubly-terminated lossless networks using the Darlington procedure⁽²⁷⁾. The reasons for the choice of this particular form of the transfer function are given by Papoulis⁽²⁰⁾. Essentially, if the function $|H|^2$ is not of this form, then one that is, with the same degree, will provide better filtering action.

The filter function f is to be near zero in the band of frequencies the filter transmits (the 'pass-band'), and $1/f$ is to be near zero in the band of frequencies the filter attenuates (the 'stop-band'). It is this dual nature of the filter function that proves challenging because, while multiplying f by a constant k ($0 < k < 1$) results in a function kf that is nearer zero in the pass-band, $1/kf$ is further from zero in the stop band than $1/f$.

Because the function f is to be a filter function its behavior is restricted. Papoulis explains that for 'optimum' filtering action the function f should have all its zeros real and in the pass-band. Similarly, all the poles of f should be real and in the stop-band. A typical filter function f is shown in Figure 1.

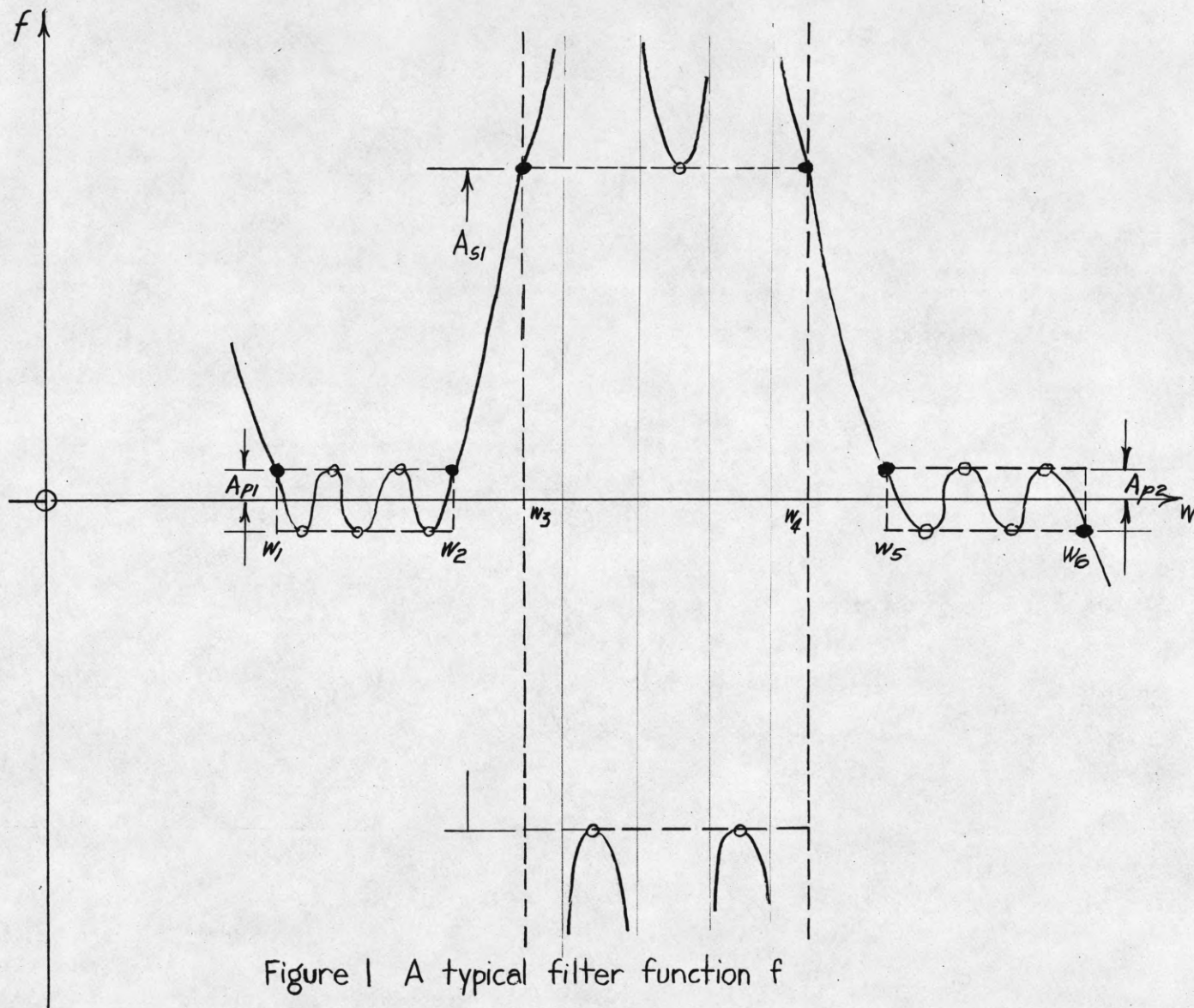


Figure 1 A typical filter function f

The derivable extrema of f have magnitudes that are specified. These magnitudes are called the 'extremum ordinates'. The frequency at which an extremum ordinate occurs is not known before the approximation procedure has been completed. The frequencies w_1, w_2, \dots , the 'band-edge frequencies' are also usually specified, as well as the magnitude the function f is to have at these frequencies. The pair $(f(w_i), w_i)$ is called a 'band-edge coordinate'.

If the band-edge coordinates and the extremum ordinates are specified, it is found that the number of critical points in each band is determined. However, the filter approximation problem is more tractable if the number of critical points in each band is specified. In this case the approximator needs all the extremum ordinates and, if the entire frequency axis is divided into stop-and pass-bands, a band-edge coordinate at every common band-edge. This band-edge coordinate is thought of as shared between the two bands adjacent to the band-edge. The critical points are simple and in the proper bands.

The number of components necessary to realize a filter function f is roughly proportional to the total number of finite poles and zeros of the function. The cost of the filter depends on the number of components. The design procedure is to determine, using the methods presented here, a function f with a small number of poles and zeros and see if its performance is satisfactory. If its performance is not satisfactory the number of zeros or poles is increased until

the desired response is achieved. The greater the number of zeros and poles the better the filter's performance and, sadly, the greater is the filter's cost.

When all the extremum ordinates in a band are equal the filter function is said to exhibit 'equi-ripple' behavior in that band. Equi-ripple behavior is usually desired in filter functions for the following reason. In a given band the magnitude of an extremum represents a deviation from the ideal. If one peak has a certain magnitude then the filter designer, in effect, says that he is prepared to allow this deviation from the ideal. If the other peak magnitudes are also this value, the filter will be 'better' than if some of these extremum magnitudes have a lower value. In a sense, the filter designer says that since he is prepared to accept some deviation from the ideal at one peak, he is prepared to accept this deviation at all the peaks. Our methods do not depend on the equi-ripple behavior of functions, but most of the applications are made for functions with this behavior.

In diagrams the following notation is used. An assigned extremum ordinate is indicated with a o . This symbol may be imagined a ring on a rod; the symbol is to indicate that the magnitude of the function is fixed, but not the extremum frequency. If a function is to assume a definite magnitude at a definite frequency (usually a band-edge coordinate) this magnitude and frequency is represented with a filled-in o , a \bullet . This symbol is to indicate that horizontal and vertical movement of the function is prevented at this

point. Both these symbols are used in Figure 2. This figure shows typical specifications for a particular type of filter function f .

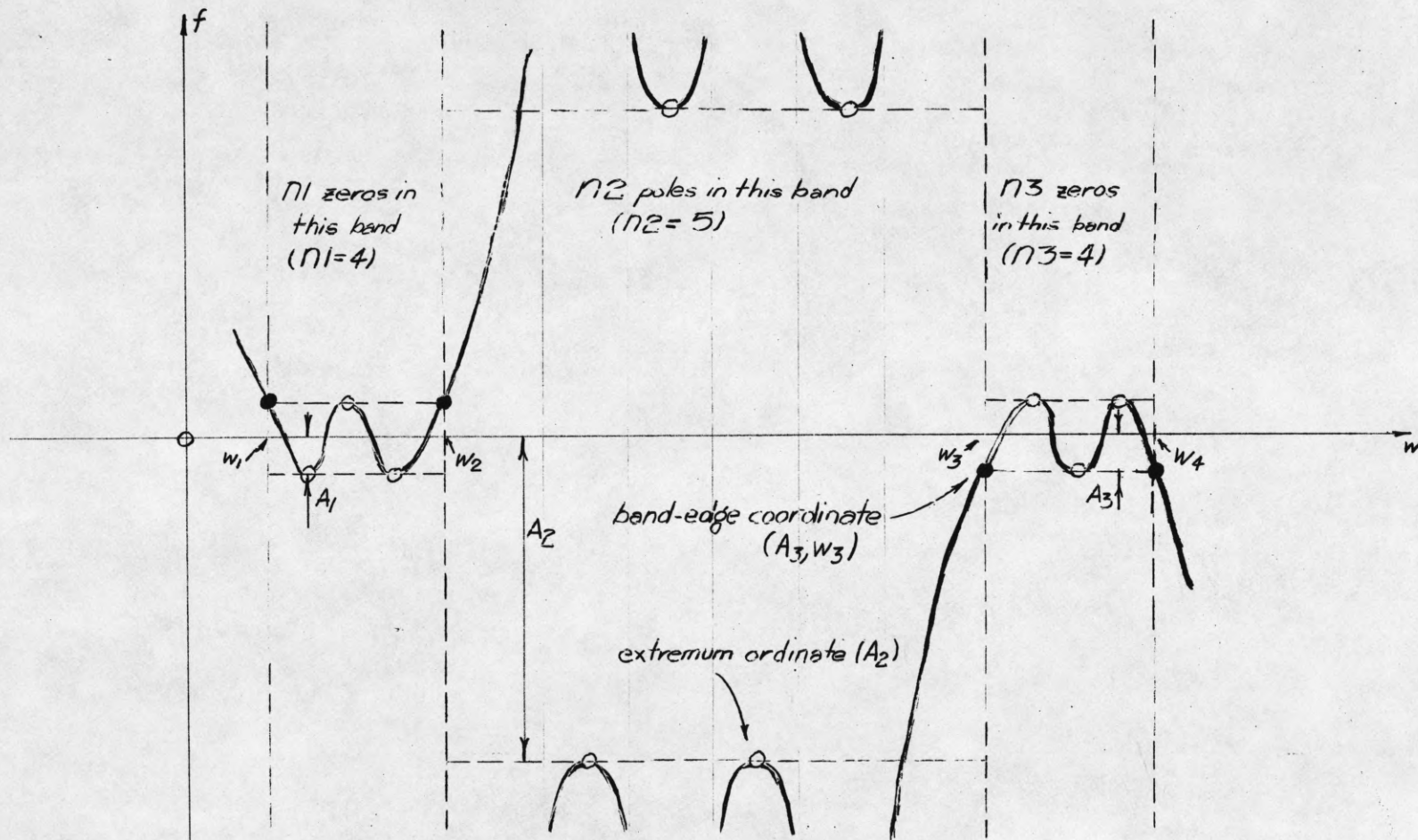


Figure 2. A typical filter function with the usual specifications indicated.

CHAPTER 2

SOME EARLIER WORK

Too much has been written about filter approximation to give a complete summary of the field here⁽²⁹⁾. A review is given of some results that are useful for our purposes.

The analytical approach has been used by Cauer, Guillemin, and Piloty, among many others^(3,4,7,8,11,12,15,18,21,29). This approach uses results of the mathematicians Chebyshev, Bernstein, and Zolotarev^(1,5,9,14,25,26,30). It leads typically to two types of filters -- the 'low-pass Chebyshev' and the 'low-pass elliptic' or 'Cauer' filters.

The low-pass Chebyshev filter function is

$$f = T_n(w) = \cos n \arccos w ,$$

the n^{th} order Chebyshev polynomial^(10,27). This polynomial has n real simple zeros in $(-1,1)$. It exhibits unity-bounded equi-ripple behavior in the interval (the pass-band in our nomenclature), and the polynomial has unity magnitude at the band edges. Of all n^{th} order polynomials that are unity bounded in $(-1,1)$, the Chebyshev polynomial $T_n(w)$ increases in magnitude the most rapidly outside the pass-band. This high rapidity of magnitude increase is a characteristic property of all equi-ripple functions which makes them of particular interest as filter functions.

The low-pass elliptic filter function has the form

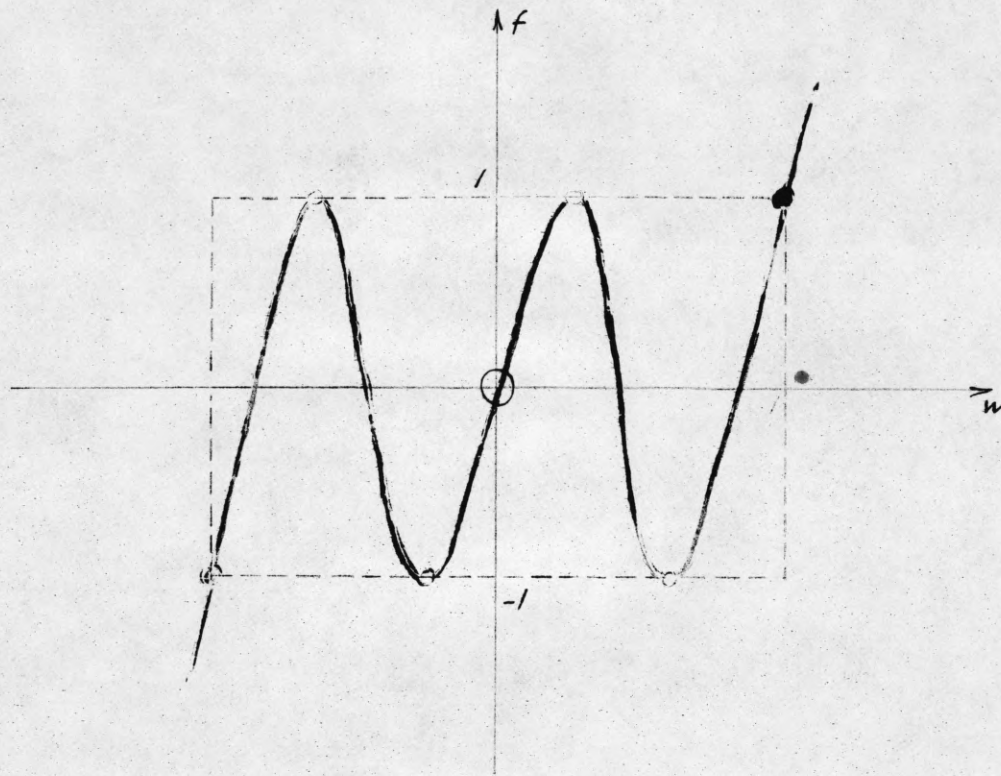


Figure 3. A Chebyshev Polynomial.

$$f(w) = \frac{w'(w'^2 - w_1^2)(w'^2 - w_2^2) \dots (w'^2 - w_n^2)}{C (w_1^2 w'^2 - 1)(w_2^2 w'^2 - 1) \dots (w_n^2 w'^2 - 1)}$$

or

$$f(w) = \frac{(w'^2 - w_1^2)(w'^2 - w_2^2) \dots (w'^2 - w_n^2)}{C (w'^2 w_1^2 - 1)(w'^2 w_2^2 - 1) \dots (w'^2 w_n^2 - 1)},$$

where $w' = sw$. Only the first form will be discussed. The numbers C , w_i and s are chosen according to well-known rules (3,8,12). A low-pass elliptic filter function is shown in Figure 4.

The low-pass filter function of Figure 4, the elliptic filter function, has $2n + 1$ real zeros in the pass-band $(-1,1)$, and $2n$ finite real poles in the stop-band. It has a zero at the origin and a pole at infinity. The parameter A is the minimum of the function's magnitude between two poles. That frequency nearest $w = 1$ for which $f(w) = A$ is denoted by w_s . The parameter A determines the closeness of w_s to 1. The larger A , the further w_s is from 1. This is a general property of filter functions. This property is sometimes stated as follows -- the faster the rate of cutoff (closeness of w_s to 1), the less the minimum attenuation in the stop-band (A). A and w_s cannot be prescribed independently, since this would involve the specification of a band-edge coordinate in addition to the band-edge coordinate already specified, that is $(1,1)$. The calculation of this filter function ordinarily requires a table of elliptic functions, hence the name.

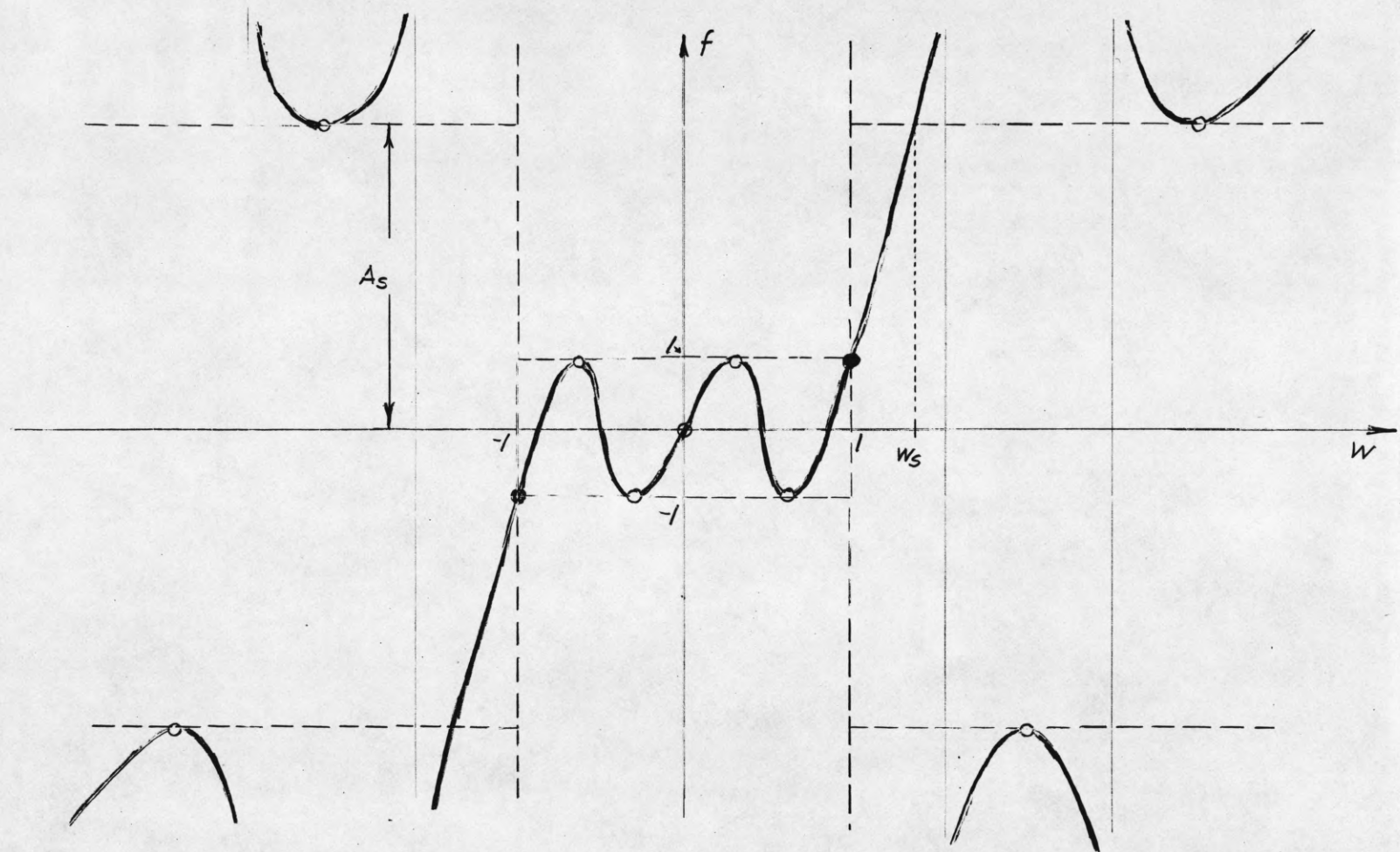


Figure 4. A low-pass elliptic filter function

Frequency transformations are used to derive band-pass and high-pass filter functions from these low-pass filter functions (7,12,27).

The potential analogy is useful in obtaining filter functions (6,16,17,24). An iterative method given by Bennett⁽¹²⁾ is useful in obtaining the low-pass elliptic filter function.

The second half of this chapter is devoted to a discussion of the mathematics of polynomial approximation (1,14,19,23,25,26,30). The filter approximation problem is related to the problem of polynomial approximation, but the two problems are distinct.

Let $[a,b]$ be a closed, finite interval. Let $g(x)$ be a given continuous function and let $W(x)$ (the 'weight function') be a given function that is continuous and non-zero and non-infinite in (a,b) . Let

$$Q = q_0 x^n + q_1 x^{n-1} + \dots + q_n, \quad (1)$$

where n is given.

The problem of Chebyshev approximation is to find those Q for which the deviation of Q from g ,

$$E_Q = \max_{a \leq x \leq b} |W(x) (Q(x) - g(x))|$$

is minimum for all functions Q of the form (1).

The existence theorem states that among all functions of the form (1) there is one, f , for which E_Q has a minimum.

The fundamental theorem of polynomial approximation is Chebyshev's theorem. This theorem states that the function f deviating least from g is uniquely determined and characterized by the following property -- the number of consecutive points of $[a, b]$ at which the difference-product $W(x) [f(x) - g(x)]$ (with alternate sign changes) takes the value $\pm E_Q$, is not less than $n + 2$.

There are only a few methods for finding polynomial approximations numerically^(14,26). One of the most popular is the second method of Remez⁽²³⁾. This method will now be described.

It is always possible to find a Q of the form (1) such that at $n + 2$ points $x_0 < x_1 < \dots < x_{n+1}$ in $[a, b]$ the function $E(x) = Q(x) - g(x)$ assumes values that alternate in sign. Let $W(x) = 1$. In every interval between two consecutive x_i , x_j and x_{j+1} there is at least one point at which $E(x) = 0$. Let this point be z_{j+1} . Let $z_0 = a$ and $z_{n+2} = b$. Suppose $E(x_j) > 0$. Then choose a new x_j such that $E(x_j) \gg E(x)$ for $z_j < x < z_{j+1}$. If $E(x_j) < 0$, then choose the new x_j so that $E(x_j) \leq E(x)$ for $z_j < x < z_{j+1}$. Repeat this process for all j from 0 to $n + 1$.

At the $n + 2$ points x_j the following equation is to hold

$$E(x_j) - C(x_j) = (-1)^j \rho \quad (2)$$

$C(x)$ is a polynomial of degree no greater than n , and is a correction to G . The equations (2) are $n + 2$ linear equations in $n + 2$ unknowns, the $n + 1$ coefficients of $C(x)$ and the

number ρ . Remez shows that

$$\pm \rho = \frac{\sum d_i |E(x_i)|}{\sum d_i}$$

where

$$d_i = \frac{1}{(x_i - x_0) \dots (x_i - x_{i-1})(x_{i+1} - x_i) \dots (x_{n+1} - x_i)} > 0$$

This, ρ is a mean of the deviations $E(x_i)$. Once ρ is determined the polynomial $C(x)$ may be determined using a Lagrangian interpolation at $n + 1$ of the $n + 2$ points x_i . The basic operations of Remez's method are extremum location, calculation of ρ , and Lagrangian interpolation. The convergence of Remez's second method is demonstrated by Novodroskii and Pinsker⁽¹⁹⁾. Remez's second method is sometimes called the method of 'equalization at extrema'.

Algorithm I of the next chapter may be viewed as a modification of Remez's second method. The function $g(x)$ is always x^{n+1} . Equation (2) is also changed to

$$E(x_i) - C(x_i) = (-1)^i k_i \rho$$

where k_i is a fixed positive number. The introduction of the number k_i permits the application of the basic ideas of Remez's method to the filter approximation problem. The method of finding a minimum deviation approximation is artificial when applied to the filter approximation problem and masks several of the points of interest, and so a more direct method is employed in the next chapter.

CHAPTER 3

ALGORITHM I

In this chapter an iterative procedure is proposed to find the polynomial with the following properties. The polynomial has all its zeros real, simple, and in the interval under consideration. This polynomial is to assume given values at its extrema and at the interval endpoints. The assigned values alternate in sign. That is, the extremum ordinates and the band-edge coordinates are specified.

All discussions relate to a finite interval $[a, b]$ called the 'band'. A rational function $W(x)$ called the 'weight function', with real zeros and poles not in the interval, is specified. $W(x)$ changes 'slowly' over the band. Assume $W(x) > 0$ for convenience, in the band.

A polynomial $p(x)$ is desired. This polynomial has n simple zeros in the band, and is of degree n . If the $n + 1$ extremum locations of $p(x)$ are $a = x_0 < x_1 < \dots < x_{n-1} < x_n = b$, then at each of these extremum locations the following equation is to hold.

$$W(x_i) p(x_i) = (-1)^i v_i \quad (i = 0, 1, \dots, n)$$

The numbers v_i are given numbers that are greater than zero. Between every consecutive pair of zeros of p , Wp has the same sign. In each of these inter-zero intervals, that x_i is selected for which Wp has its maximum magnitude. Usually these x_i are determined by differentiation. That these derivable extremum

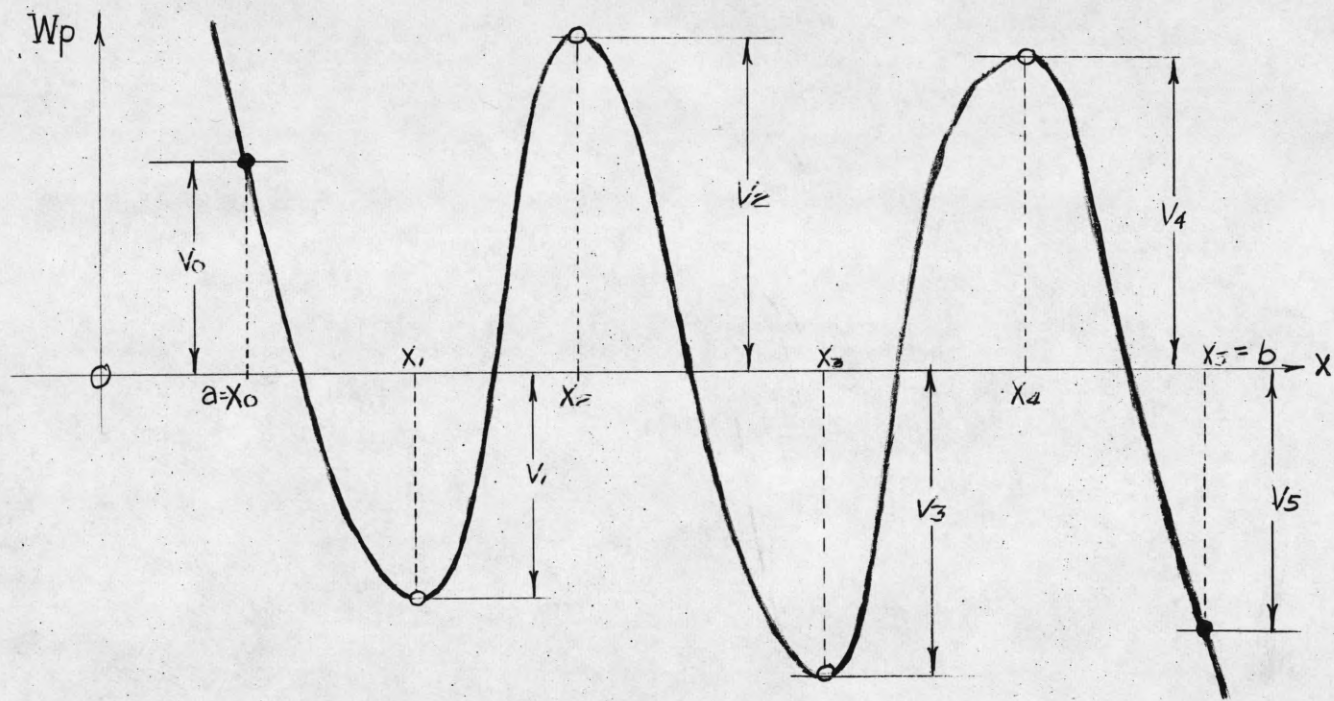


Figure 5. The polynomial p of Algorithm I.

locations can be so found should always be verified. It is assumed henceforth that W_p has $n-1$ derivable extrema in the band.

To find the polynomial p a set of iterates is constructed that converges to p . We first construct a polynomial called the 'initial polynomial'. The initial polynomial is of degree n and has n simple zeros in the band. Typical initial polynomials are

$$B^0(x) = \prod_{k=1}^n \left(x - a - \frac{b-a}{n+1} k \right)$$

or

$$B^0(x) = T_n \left(\frac{2x - a - b}{b - a} \right)$$

where $T_n(x)$ is the Chebyshev polynomial of order n . This second initial polynomial is the more usual.

We then calculate the zeros of the derivative of $B^0 W$ and have the set of $n + 1$ extremum locations $a = x_0 < x_1 < \dots < x_{n+1} < x_n = b$. We keep $x_0 = a$ and $x_n = b$ throughout the discussion. Next, we form the Lagrangian interpolator⁽¹³⁾ with weight function $W(x)$

$$\sum_{i=0}^n \frac{\prod_{j \neq i} (x - x_j)}{\prod_{j \neq i} (x_i - x_j)} \frac{(-1)^i v_i}{W(x_i)} \quad (3)$$

This completes the first iteration; the Lagrangian interpolator is the first iterate polynomial $B^1(x)$. It is of degree n and has n simple zeros in the band.

On the k^{th} iteration, we calculate the $n + 1$ extremum locations of $B^{k-1}W$ and call them $a = x_0 \ x_1 \ x_2 \ \dots \ x_{n-1} \ x_n = b$ and form the Lagrangian interpolator (3) which is the next iterate $B^k(x)$. Only the extremum locations change from iteration to iteration. The algorithm is continued until the change in the coefficients of the $B^k(x)$ becomes sufficiently small.

We note that $n + 1$ functional values and abscissa are needed for Lagrangian interpolation with a polynomial of degree n . It is interesting to compare Algorithm I with Lagrangian interpolation. In the algorithm some of the $n + 1$ functional value and abscissa data is changed for functional value and derivative nullity data. There are still $n + 1$ pairs of information -- the 2 band-edge coordinates and the $n - 1$ extremum ordinates.

If note is made that $B^k(x)$ is of degree n , as is $B^{k-1}(x)$, then B^k may be rewritten in the form

$$\begin{aligned} B^k(x) &= \sum_{i=0}^n \frac{(-1)^i v_i}{W(x_i)} \frac{\prod_{j \neq i} (x-x_j)}{\prod_{j \neq i} (x_i-x_j)} + B^{k-1} - B^{k-1} \\ &= B^{k-1}(x) - \sum_{i=0}^n \frac{B^{k-1}(x_i) W(x_i) - (-1)^i v_i}{W(x_i)} \frac{\prod_{j \neq i} (x-x_j)}{\prod_{j \neq i} (x_i-x_j)} \\ &= B^{k-1}(x) - C(x). \end{aligned}$$

$C(x)$ is called the correction polynomial. It assumes the value

$$\frac{B^{k-1}(x_i) W(x_i) - (-1)^i v_i}{W(x_i)}$$

at the extrema of $WB^{k-1}(x)$. Thus the correction polynomial pushes the value of $B^{k-1}W$ at any extremum down to v_i . Figure 6 illustrates this idea.

A proof that an analogous algorithm has a limit for the maximum deviation at an extremum is now presented.

We can assume that $W(x)$ is unity with no loss of generality. On the k th iteration let the deviation of the magnitude of the j th peak of B^{k-1} from v_j be $E(k,j)$. Let $M(k) = \max_j E(k,j)$. Note that $E(k,j)$ is non-negative since B must go through $\pm v_j$ near the peak. The correction polynomial is, as above, $C(x) = B^{k-1}(x) - B^k(x)$. $B^k(x)$ is determined by the Lagrange interpolation formula, but it is not the k th iterate, although B^{k-1} is the $k-1$ st iterate. $C(x)$ is of degree n or less.

If it is assumed that $E(k,j)$ is non-zero for all j then, since $C(x_0) = 0$ and $C(x_n) = 0$, $C(x)$ has at most one zero between every derivable extremum location of B^{k-1} .

If $E(k,j) = 0$ for some j then related arguments show that the zero between the peaks moves to the peak location itself, as illustrated in Figure 8.

Three types of behavior are possible at a peak of the iterate B^{k-1} . These are illustrated in Figure 9. The three types of behavior are called case 1, 2, and 3 behavior.

We are going to find a $\mu > 0$ such that $B^{k-1} - \mu C$ is

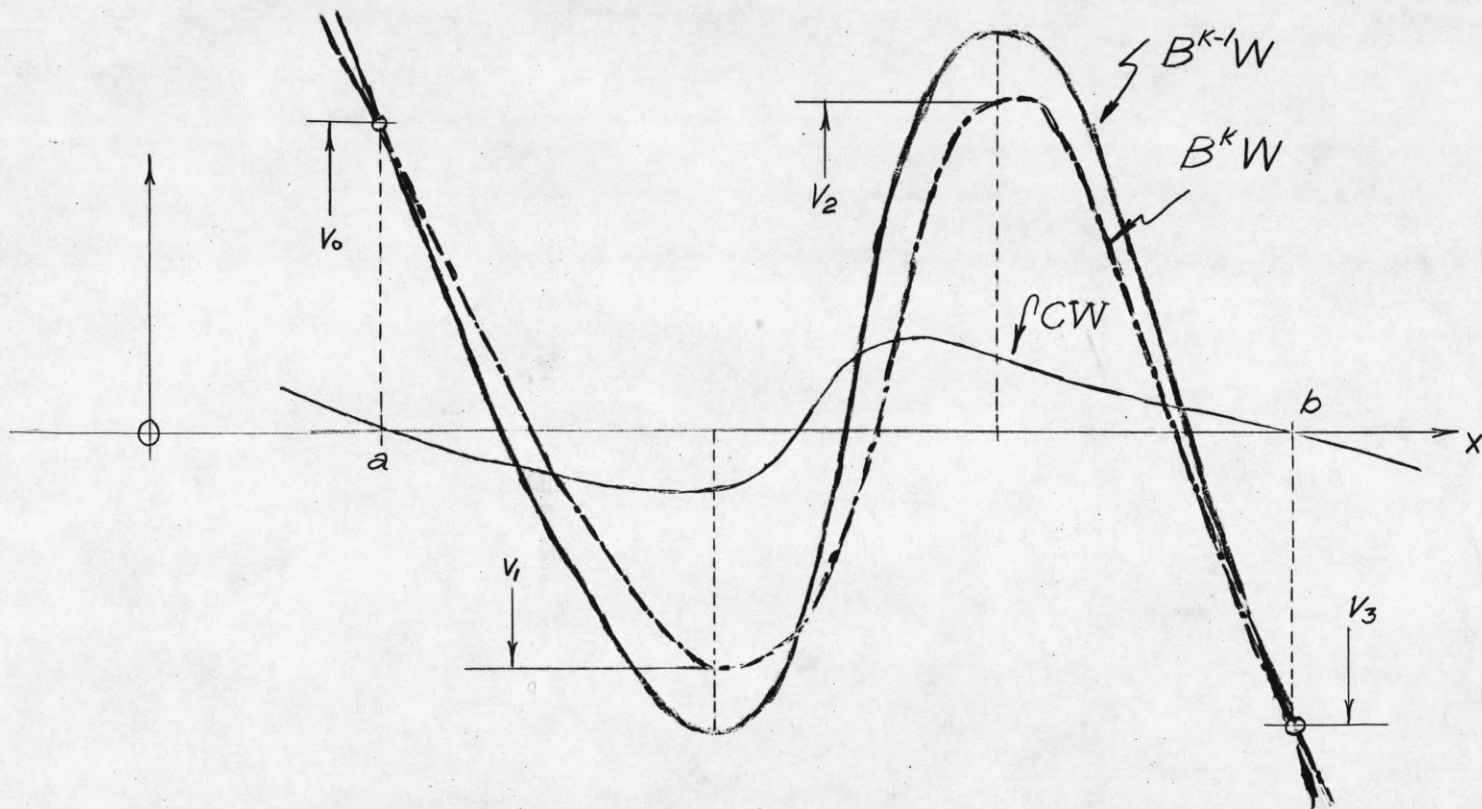


Figure 6. The iterative procedure

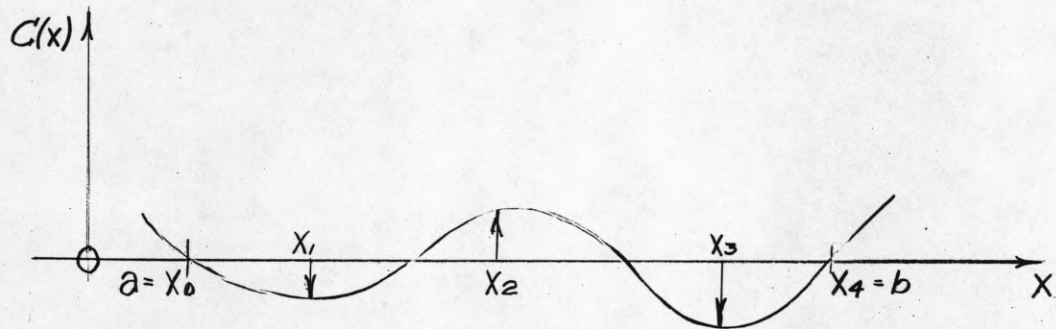


Figure 7. Behavior of the correction polynomial.

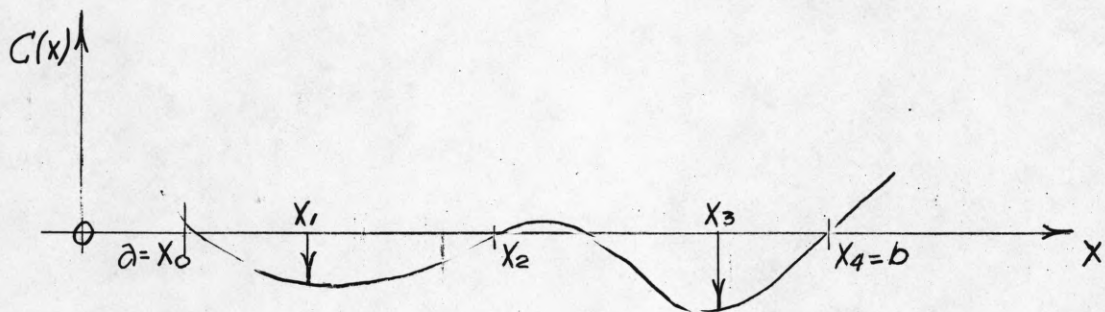


Figure 8. The correction polynomial.

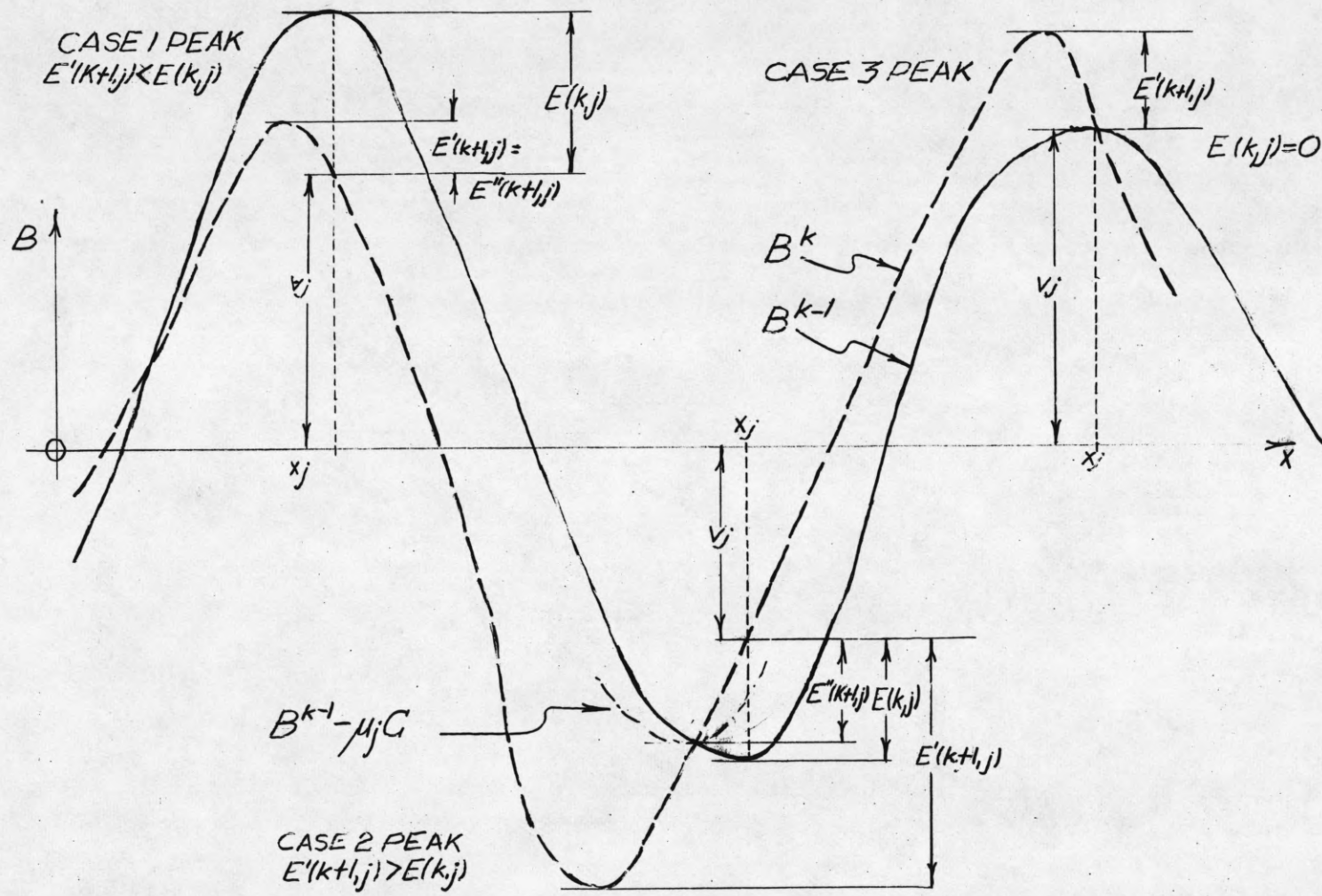


Figure 9. Behavior at peaks of $B^{k-1}(x)$.

the next iterate and $M(k+1) < M(k)$. The deviation of the next iterate will be $E(k+1, j)$ at the j th peak.

Let $E'(k+1, j)$ be the deviation at the j th peak of B^k . B^k is determined from the Lagrangian interpolation, and is not the next iterate (in general). Let $E''(k+1, j)$ be the deviation at the j th peak of $B^{k-1} - \mu_j C$, where μ_j is a number yet to be determined.

For case 1 peaks, let $\mu_j = 1$. Since at a case 1 peak $E'(k+1, j) < E(k, j)$, then if $0 < \mu \leq \mu_j = 1$, $E''(k+1, j) = E'(k+1, j) \leq E(k+1, j) < E(k, j) \leq M(k)$.

For case 2 peaks define μ_j such that $B^{k-1} - \mu_j C$ has an extremum at the x^* near the peak of B^{k-1} for which $C(x^*) = 0$. The magnitude of $B^{k-1}(x^*) < E(k, j) + v_j$ and so if $0 < \mu < \mu_j \leq 1$, $E''(k+1, j) \leq E(k+1, j) < E(k, j) \leq M(k)$.

Let $E' = \max E''(k+1, j)$, where the maximum is taken over all case 1 and case 2 peaks. E' is defined unless all peaks are case 3 -- but this occurrence signifies the termination of the algorithm.

For case 3 peaks define an μ_j such that the deviation $E''(k+1, j)$ is less than E' .

Now let $\mu = \min_j \mu_j$. Then, since $E(k+1, j) \leq E' < M(k)$, $M(k+1) < M(k)$. Thus $M(k)$ is a monotonically decreasing function of k .

We desire, of course, that $M(k)$ go to zero. To show that this is a reasonable desire, we show that, for a few examples, the difference in the deviation at a peak from iteration is proportional to the magnitude of the deviation itself.

Near a case 2 peak of B^{k-1} it is possible to represent B^{k-1} and B^k as parabolas. Thus

$$B^{k-1} = E(k,j) + v_j - L_j^2(x-x_j)^2$$

and

$$B^k = E'(k+1,j) + v_j - L_j'^2(x-x_j')^2 .$$

(Assume $B^{k-1}(x_j) > 0$ for the sake of argument.)

If μ_j is chosen as above, then if $E(k,j) = E'(k+1,j)$ and $L_j = L_j'$

$$\frac{E(k,j) - E''(k+1,j)}{E(k,j)} = \frac{1}{4}$$

If $L_j = L_j'$

$$\frac{E(k,j) - E''(k+1,j)}{E(k,j)} = \frac{1}{4} \frac{E(k,j)}{E'(k+1,j)}$$

If $E(k,j) = E'(k+1,j)$

$$\frac{E(k,j) - E''(k+1,j)}{E(k,j)} = \frac{1}{1 + L_j'^2/L_j^2}$$

These three examples show that we can expect the deviation at a case 2 peak to decrease proportionally to the deviation, that is, the deviation is expected to diminish exponentially. The case 1 peaks may be considered in the same way. If the maximum deviation does not have zero as its limit, then a better initial polynomial must be chosen.

In numerical calculations the author always uses the algorithm presented first in this chapter, that is, $\mu = 1$. A proof that this is wise would be comforting!

CHAPTER 4

ALGORITHM II

Algorithm II allows the approximator to obtain equi-ripple behavior in two or more bands simultaneously.

Algorithm II has as its basic idea, modification of the weight function of Algorithm I.

Suppose that three bands are required -- a stop-band, a pass-band, and a stop-band, in that order. The band-edges are given, the number of zeros and poles in each band is given, and the extremum ordinates and band-edge coordinates are given. Such a prescription may be represented as in Figure 10.

The algorithm proceeds as follows. We choose an initial rational function satisfying the zero and pole location criteria. This initial rational function is broken down into three factors. The factors are polynomials $p(1)$, $p(2)$, $p(3)$; $p(1)$ and $p(3)$ are for the two stop-bands, and $p(2)$ is for the pass-band. Thus

$$R = \frac{p(2)}{p(1) \cdot p(3)}$$

$1/R$ is considered first.

$$1/R = \frac{p(1) \cdot p(3)}{p(2)}$$

We rewrite this as

$$1/R = p(1) \frac{p(3)}{p(2)}$$

Now, we consider $p(3)/p(2)$ as a weight function and calculate a

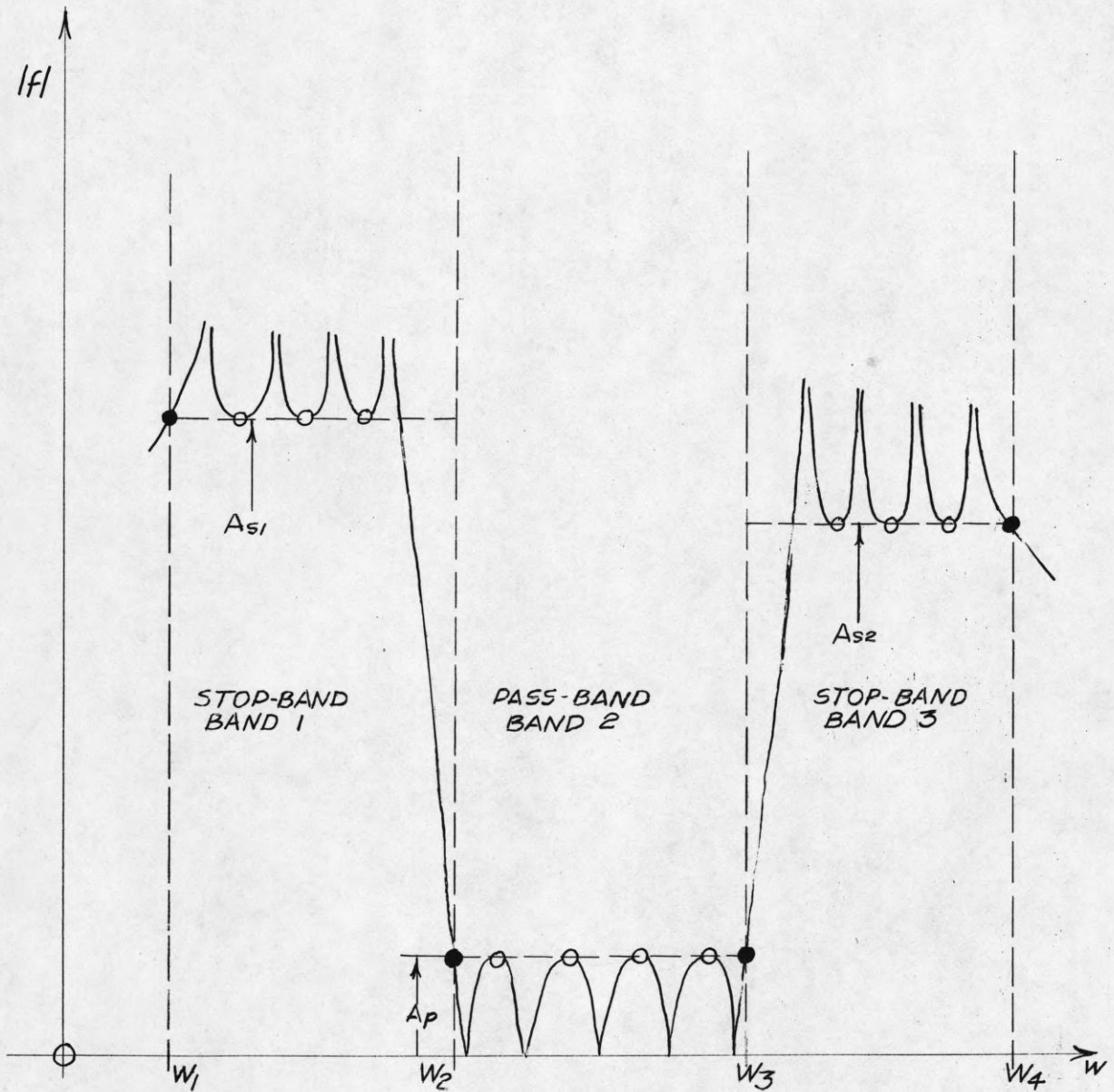


Figure 10. Typical prescription for Algorithm II.

new $p(1)$ using Algorithm I. Then the new $p(1)$ replaces the old $p(1)$. The next step is obvious -- the new R is considered as

$$p(2) = 1/p(1) p(3)$$

and a new $p(2)$ is calculated using Algorithm I.

The iterative procedure is now clear. We continue calculating new $p(1)$, $p(2)$, $p(3)$, $p(1)$, $p(2)$, $p(3)$, ... using Algorithm I.

There are a great many variations possible when using Algorithm II. One such procedure consists in calculating all the extremum locations of the rational function R and then calculating the new $p(i)$ using the Lagrangian interpolation in Algorithm I, but without determining the new extremum locations for each individual band. That is, we first calculate the zeros of the derivative of R on the real axis, and then we use the Lagrangian interpolation formula to find new $p(1)$, $p(2)$, $p(3)$, using the polynomials as they are generated as weight function factors. This constitutes one iteration. This is the procedure used by the author in all his calculations using this method; it seems to produce satisfactory results.

A convergence proof for Algorithm II has not yet been found. However, calculations have been carried out using it and no cases of non-convergence have yet been found. If the algorithm should refuse to converge, an initial rational function nearer the final rational function should, of course, be tried.

Algorithm II is almost as easy to use as Algorithm I.

The only difficulties are the differentiation procedure, which is discussed at the end of this chapter, and the choice of the band-edge coordinates, which is discussed now.

There must be only one band-edge coordinate for every adjacent pair of bands. Thus, Figure 11 indicates a proper specification of the band-edge coordinates. The same coordinate is used in the calculation of the polynomial in the lower band, $p(i)$, as for the next higher band, $p(i+1)$.

Thus the specifications for band 1, using the nomenclature of Algorithm I, read

$$\begin{array}{ll} x_0(1) = w_1 & v_0(1) = 1/A_s \text{ (band-edge coordinate)} \\ & v_1(1) = 1/A_s \text{ (extremum ordinate)} \\ \dots\dots\dots & \\ & v_{n-1}(1) = 1/A_s \text{ (extremum ordinate)} \\ x_n(1) = w_2 & v_n(1) = 1/A_p \text{ (band-edge coordinate)} \end{array}$$

and the specifications for band 2 read

$$\begin{array}{ll} x_0(2) = w_2 & v_0(2) = A_p \text{ (band-edge coordinate)} \\ & v_1(2) = A_p \text{ (extremum ordinate)} \\ \dots\dots\dots & \\ x_n(2) = w_3 & v_n(2) = A_p \text{ (band-edge coordinate)}. \end{array}$$

The two arrows indicate the connection between the v_i 's and the edge frequencies used in the calculations for band 1 and band 2.

If the preceding method for assigning the v_i and the band-edge frequencies is not followed, different results will be obtained. For example, if the specifications of Figure 12

The specified parameters are underlined

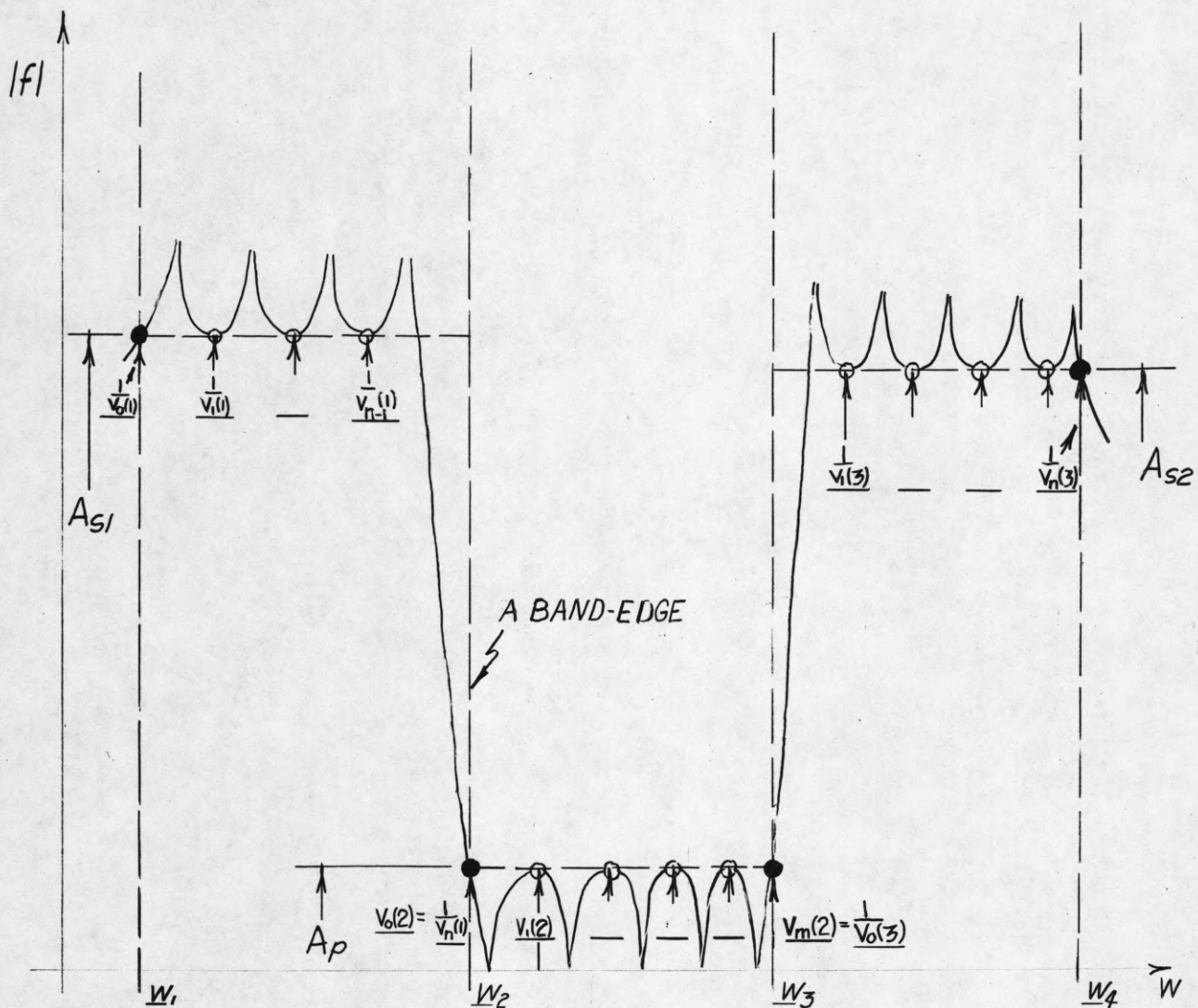


Figure II The parameters to be specified in Algorithm II.

are given, where there is no connection between the band-edge coordinates, then the function of Figure 13 may typically result. There is no control over the resultant extremum ordinates, even though the function is equi-rippled in each band. The function obtained by such an 'improper' specification may be desirable, of course; in that case, here is a method for obtaining that function.

In calculating the derivative zeros of the rational functions that appear in Algorithm II the following method is convenient. First we calculate the zeros of the derivative of each constituent polynomial. This is effectively done using Newton's method. Then we calculate the zeros of the derivative of the rational function using the earlier determined zeros as initial guesses. This calculation is also effectively done using Newton's method. Spurious derivative zeros are eliminated from the calculation in this way.

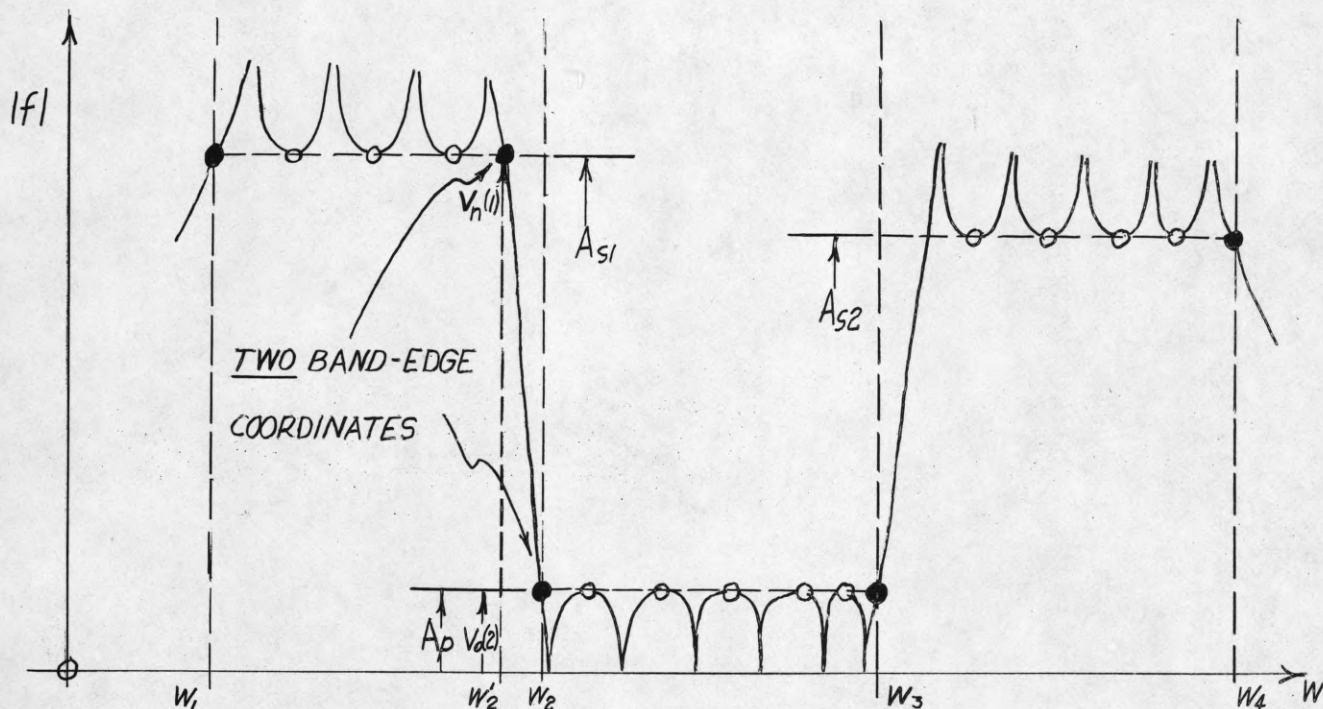


Figure 12. Improper specification for Algorithm II.

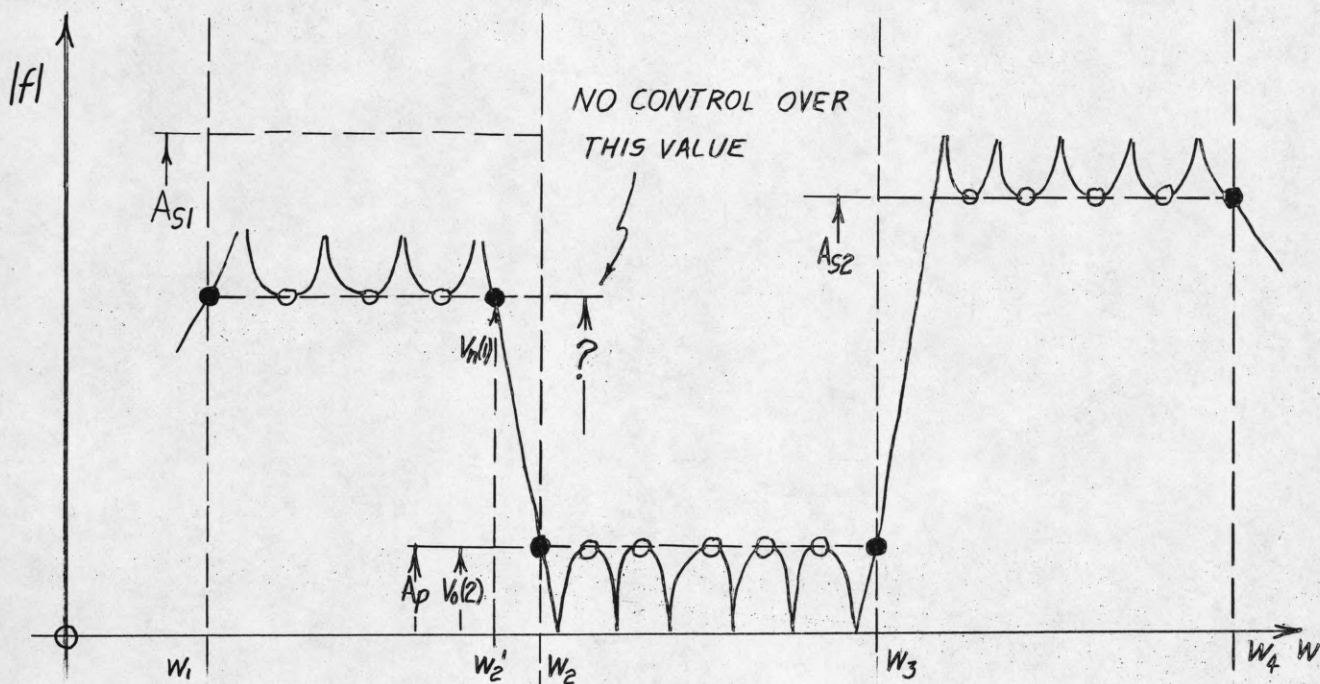


Figure 13. Resultant function for the improper specification of Figure 12 for Algorithm II.

CHAPTER 5

EXAMPLES AND RESULTS

This chapter is devoted to a series of problems with procedures and some examples and results. Some generalizations are scattered throughout. A preliminary note discusses a simplification that is possible because filter functions are considered.

As mentioned in Chapter 1 a filter function f is even or odd. We may split f up into two factors -- the first containing the origin critical point and the second factor all the other critical points. The first factor is always considered a part of any weight function. Consideration is centred on positive frequencies alone.

PROBLEM 1 (low-pass equi-ripple filter with zeros at the origin)

Construct a filter function f that has q ($q \neq 0$) zeros at the origin, n simple zeros in $(0,1)$, and whose extrema in $(0,1)$ (the pass-band) have magnitude one.

SOLUTION:

We use Algorithm I for this example. Using the nomenclature of Algorithm I, we let $W(w) = w^q$, $a = 0$, $b = 1$, and let the initial polynomial be the Chebyshev polynomial of order $2n$, $T_{2n}(w)$. All v_i are unity. There is no left endpoint extremum. However, this extremum may be considered pushed to the right by the weight function. In any case, there are $n + 1$ extrema in $(0,1)$, and this is the proper number for the

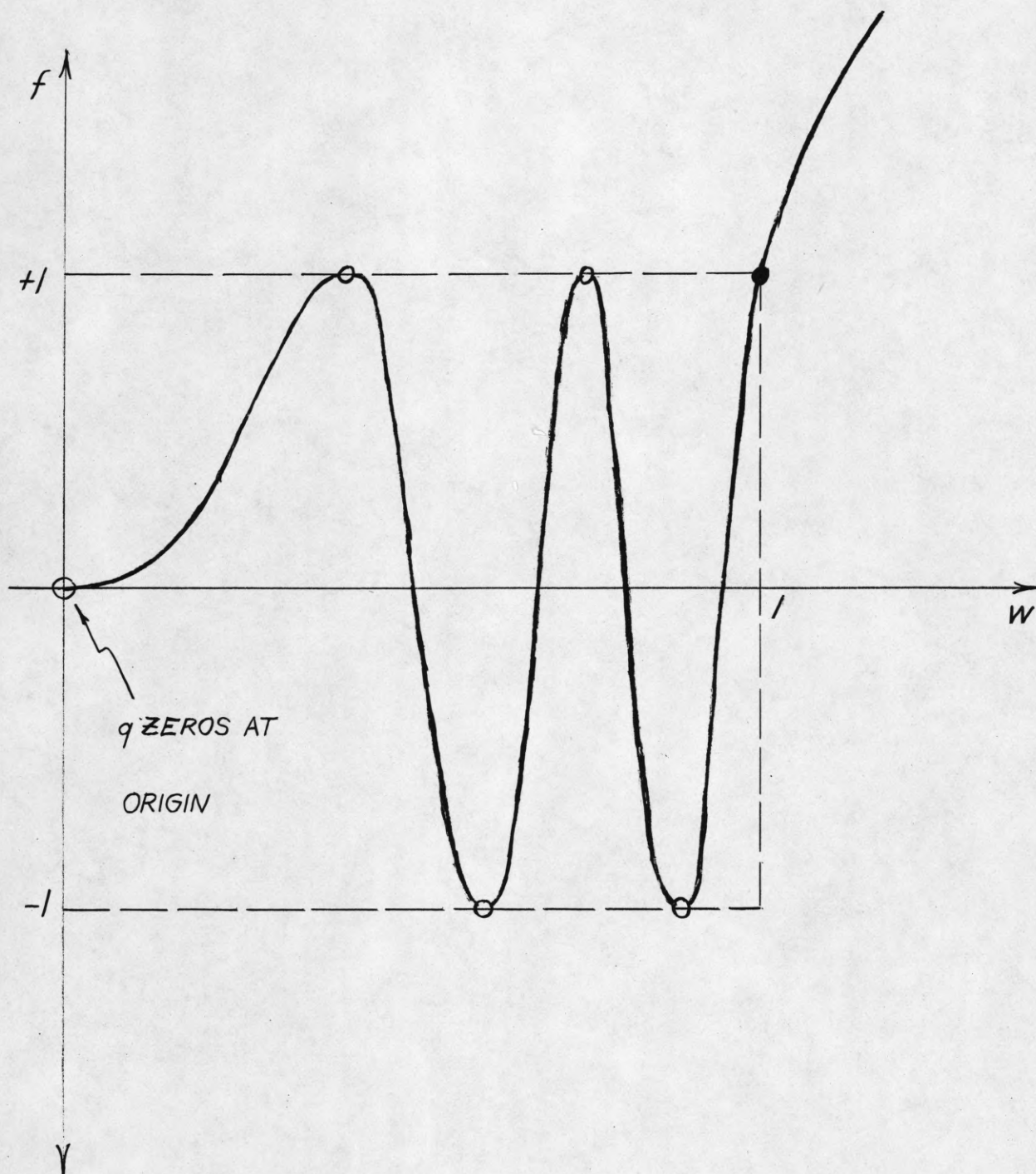


Figure 14. The polynomial of Problem 1.

Lagrangian interpolation of Algorithm I.

In the design of filters using the Darlington technique it is possible to take account of losses in elements by using the method of predistortion. The lossiness of components is measured by the quality factor, called 'Q'. The larger Q, the less dissipative is the element. Most filters are designed assuming an infinite Q. The filter's performance with real components is poorer than the performance predicted with the ideal lossless elements.

In general, for a specific filter function and a given ripple factor, there is, when using the predistortion technique of Darlington, a lower Q below which it is not possible to design for exact response. This Q we call Q'. It is found that $1/Q'$ is about one half the distance of the nearest zero of $1 + \epsilon^2 f^2(w)$ from the real w axis.

When a Chebyshev filter is built with filter elements that have a Q less than the Q' for this filter, the filter's performance is roughly that sketched in Figure 15.

On the other hand, a Butterworth filter ($f(w) = w^n$) does not have the rate of cutoff of the Chebyshev filter, but the lower limit on Q, Q', is much smaller than the Q' for the Chebyshev filter.

Thus the filter designer meets the following problem: he can design using a Chebyshev design and use lossy elements which will not give the design response, or he can design using the Butterworth filter function, predistort, and obtain the exact, but poor, response. If a cross between the Chebyshev

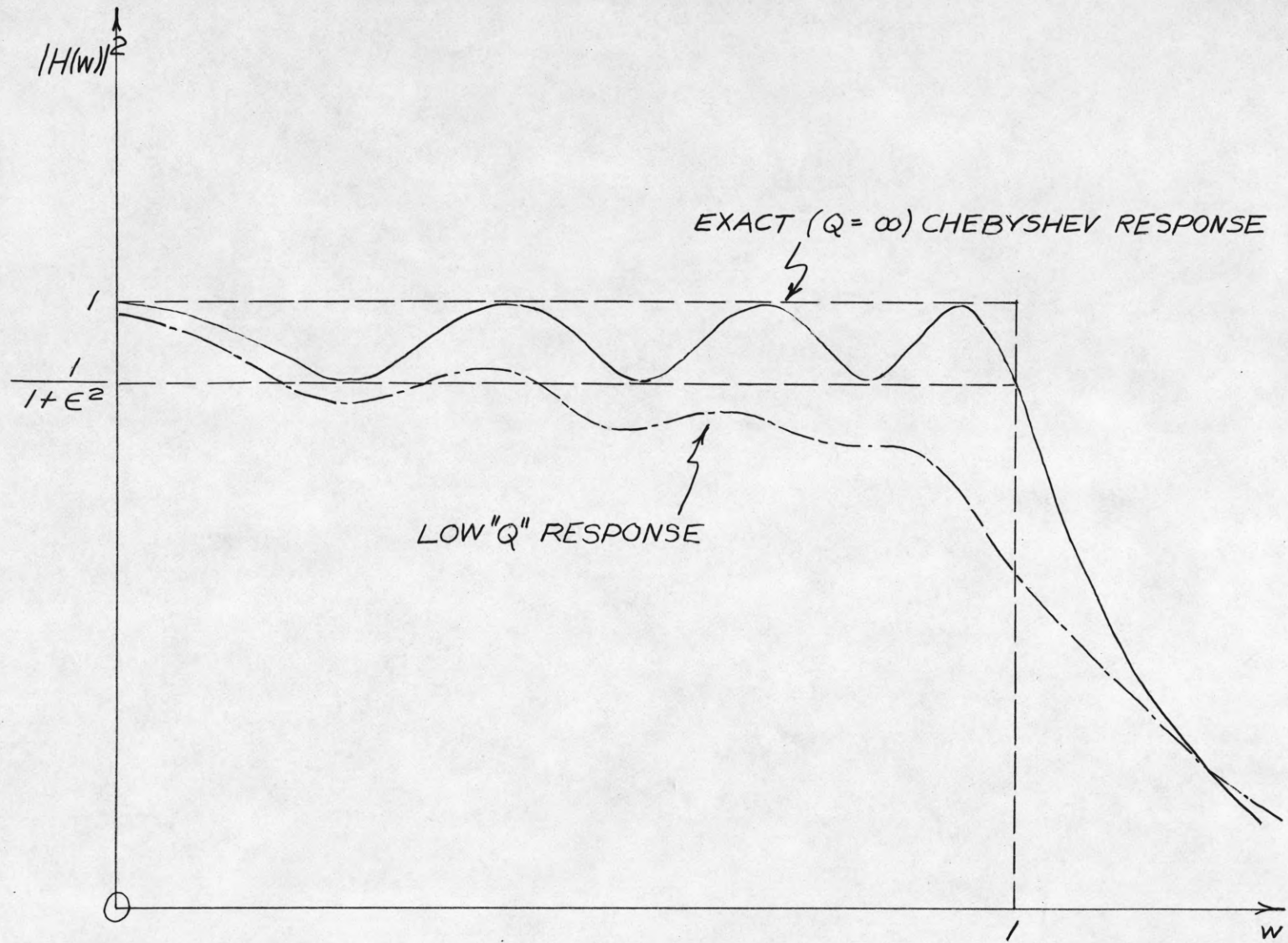


Figure 15. Effect of lossy elements on Chebyshev filter performance.

and the Butterworth is desired, the polynomials described in this problem give this class of polynomials.

For example, for the case $2n + q = 4$, the following functions are found -- $8w^4 - 8w^2 + 1$, $5.8284271w^4 - 4.8284271w^2$, and w^4 . The positions of the zeros of $1 + \epsilon^2 f^2(w)$, for the above functions f , are shown in Figure 16 for $\epsilon = 0.765$.

An examination of Figure 16 shows that the poles do not move very far away from the axis for the intermediate polynomial. The Butterworth is really much worse. The slight movement amounts to 50% of the distance from the axis of the Chebyshev transfer function poles, and this is a significant movement.

Table 1 gives a listing of some of the polynomials of this problem calculated by the author.

PROBLEM 2 (Cocci-Cauer-Darlington-elliptic filter)

Find the following filter function f . The function f has q zeros at the origin, n simple zeros in $(0,1)$ (the pass-band), m simple poles in the stop-band and at least one pole at infinity. The function f has extremum ordinates of A_p in the pass-band and A_s in the stop-band. The (one) band-edge coordinate is $(A_p, 1)$.

SOLUTION:

Algorithm II is to be used here, since there is equi-ripple behavior in more than one band. We let $W(w) = w^q$. The polynomial for the pass-band is called $p(1)$, and the initial

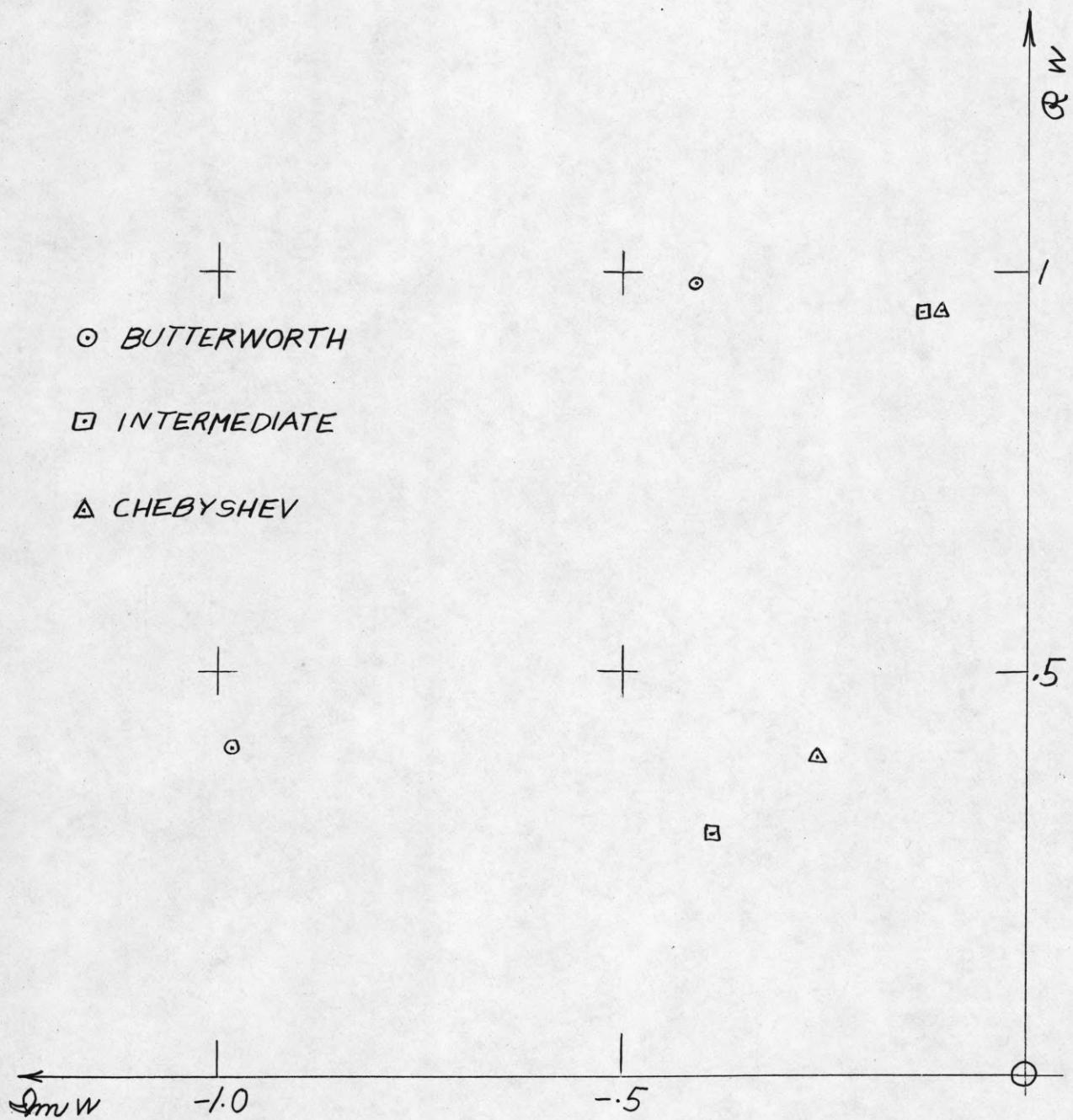


Figure 16. Poles in upper right quadrant of the transfer function $(1+\epsilon^2 f^2)^{-1}$ for text filter functions f .

(Poles exhibit quadrantal symmetry)

TABLE 1

Some low-pass equi-ripple polynomials

1.	(2.	$w^2-1.$)	
w	(4.	$w^2-3.$)	
w^2	$w^2(5.82842w^2-4.82842)$			
w^3	$w^3(7.63914w^2-6.63914)$			
w^4	$w^4(9.44354w^2-8.44354)$			
w^5	$w^5(11.2450w^2-10.2450)$			
	(8.	$w^4-8.$	$w^2+1.$)
w	(16.	$w^4-20.$	$w^2+5.$)
w^2	$w^2(25.9904w^4-36.1865w^2+11.1962)$			
w^3	$w^3(38.2555w^4-56.9512w^2+19.6957)$			
w^4	$w^4(52.8152w^4-82.3164w^2+30.5012)$			
w^5	$w^5(69.6747w^4-112.287w^2+43.6127)$			
	-32.	$w^6+48.$	$w^4-18.$	$w^2+1.$
w	(-64.	$w^6+112.$	$w^4-56.$	$w^2+7.$
w^2	$w^2(-109.59771w^6+210.52271w^4-122.03435w^2+20.10936)$			
w^3	$w^3(-172.24904w^6+353.11392w^4-225.01050w^2+43.14562)$			

$p(1)$ is $T_{2n}(w)$, the Chebyshev polynomial of order $2n$. The initial polynomial in the stop-band is $p(2) = w^{2m}T_{2m}(1/w)$.

When calculating the new $p(1)$'s, which we sometimes refer to as 'calculating in the pass-band', we use $v_0(1) = v_1(1) = \dots = v_n(1) = A_p$. When calculating in the stop-band we use $v_0(2) = 1/A_p$; $v_1(2) = v_2(2) = \dots = 1/A_s$. Note the same band-edge coordinate.

This filter function is sketched in Figure 17.

Sometimes it is useful to pick the band-edge coordinate as $(A_s, 1)$ or $((A_s A_p)^{1/2}, 1)$ instead of $(A_p, 1)$. The functional behavior of Figure 17 results.

Calculations have been made by the author for q from 1 to 10, m from 1 to 2, for those n for which $2n \geq 2m+q$, and with $A_p = 1$ and A_s varying from 100 to 10,000,000. for this filter function. The typical number of iterations is twenty, at about five seconds per iteration, on the Control Data 1604 computer. In Table 2 we reproduce some of the filter functions for q, n and m fixed, and with A_s varying from 10 to 10^7 . In Table 3 we reproduce some filter functions for low n, m , and q . The parameter A_s is fixed at 10^3 . The parameter w_s is the frequency nearest to $w = 1$ for which $|f(w)| = A_s$.

On examining Table 2 we see that the parameter w_s increases as A_s increases. Also we see that the numerator polynomial as $A_s \rightarrow \infty$ is the Chebyshev polynomial of the proper order, or the function of Problem 1 of the proper order.

An examination of Table 3 brings out two points mentioned in the second chapter. The greater the number of poles

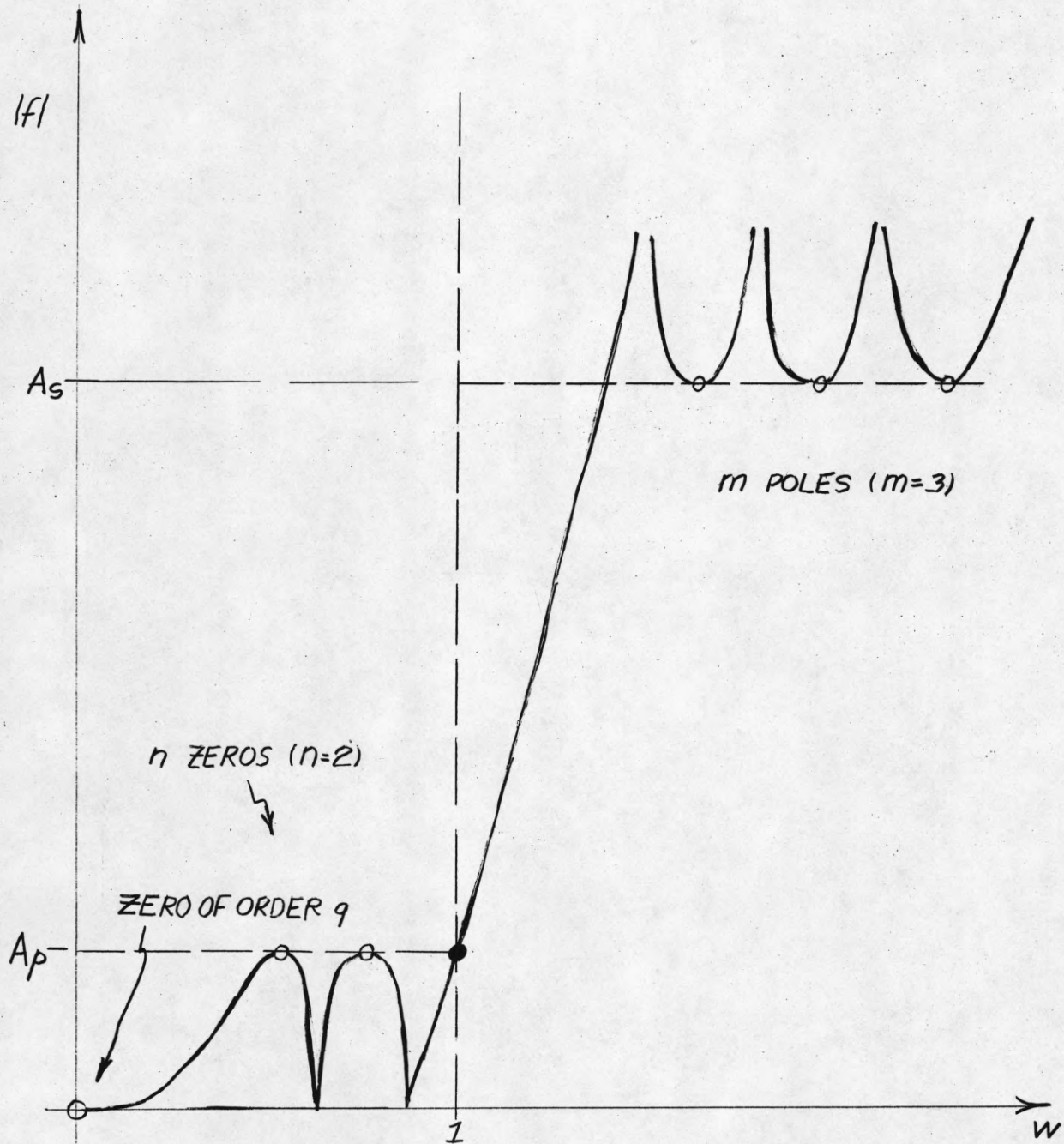


Figure 17 Behavior of low-pass "elliptic" filter function of Problem 2.

TABLE 2

Some filter functions calculated by the author

$\frac{w(7.1721483w^2 - 6.1721483)}{(1.8159749w^2 - 2.8159749)}$	$A_S = 10$ $w_S^2 = 1.155$
$\frac{w(5.1500693w^2 - 4.1500693)}{(.61131646w^2 - 1.6113165)}$	$A_S = 31.4$ $w_S^2 = 1.457$
$\frac{w(4.4812813w^2 - 3.4812813)}{(.24747388w^2 - 1.2474739)}$	$A_S = 100$ $w_S^2 = 1.978$
$\frac{w(4.2133388w^2 - 3.2133388)}{(.10805500w^2 - 1.1080550)}$	$A_S = 314$ $w_S^2 = 2.796$
$\frac{w(4.0969693w^2 - 3.0969693)}{(-.048774972w^2 - 1.0487750)}$	$A_S = 1000$ $w_S^2 = 4.031$
$\frac{w(4.0445772w^2 - 3.0445772)}{(.02235037w^2 - 1.0223504)}$	$A_S = 3140$ $w_S^2 = 5.868$
$\frac{w(4.0205989w^2 - 3.0205989)}{(.010312676w^2 - 1.0103127)}$	$A_S = 10,000$ $w_S^2 = 8.579$
$\frac{w(4.0095414w^2 - 3.0095414)}{(.0047735612w^2 - 1.0047736)}$	$A_S = 31,400$ $w_S^2 = 12.56$
$\frac{w(4.0044245w^2 - 3.0044245)}{(.0022128638w^2 - 1.0022129)}$	$A_S = 100,000$ $w_S^2 = 18.43$
$\frac{w(4.0020528w^2 - 3.0020528)}{(.0010265121w^2 - 1.0010265)}$	$A_S = 314,000$ $w_S^2 = 27.04$
$\frac{w(4.0009526w^2 - 3.0009526)}{(.00047633373w^2 - 1.0004763)}$	$A_S = 1000,000$ $w_S^2 = 39.69$
$\frac{w(4.0004421w^2 - 3.0004421)}{(.00022106632w^2 - 1.0002211)}$	$A_S = 3,140,000$ $w_S^2 = 58.25$
$\frac{w(4.0002052w^2 - 3.0002052)}{(.00010260382w^2 - 1.0001026)}$	$A_S = 10,000,000$ $w_S^2 = 85.50$

TABLE 3

Some filter functions for Problem 2 of the text
calculated by the author

$\frac{w(4.0969693w^2 - 3.0969693)}{-0.048774972w^2 + 1.0487750}$	$w_s^2 = 4.0316$
$\frac{w^2(6.2178761w^2 - 5.2177861)}{-0.16500373w^2 + 1.1650037}$	$w_s^2 = 2.4341$
$\frac{w^3(8.4674314w^2 - 7.4674314)}{-0.32514225w^2 + 1.3251422}$	$w_s^2 = 1.8958$
$\frac{w^3(9.2603653w^2 - 8.2603653)}{0.072832209w^4 - 0.77351737w^2 + 1.7006852}$	$w_s^2 = 1.6885$
$\frac{w^4(10.792565w^2 - 9.7925652)}{-0.50816471w^2 + 1.5081647}$	$w_s^2 = 1.6409$
$\frac{w^4(12.425653w^2 - 11.425653)}{0.24947778w^4 - 1.6051787w^2 + 2.3557009}$	$w_s^2 = 1.4596$
$\frac{8.7661629w^4 - 8.9620823w^2 + 1.1959194}{-0.19591943w^2 + 1.1959194}$	$w_s^2 = 2.2688$
$\frac{w(19.472184w^4 - 25.250862w^2 + 6.7786783)}{-0.45531619w^2 + 1.4553162}$	$w_s^2 = 1.6885$
$\frac{w(23.460677w^4 - 31.445512w^2 + 8.9848350)}{0.15364651w^4 - 1.2205469w^2 + 2.0669004}$	$w_s^2 = 1.5065$
$\frac{w^2(34.494763w^4 - 50.075369w^2 + 16.580606)}{-0.75936324w^2 + 1.7593632}$	$w_s^2 = 1.4596$

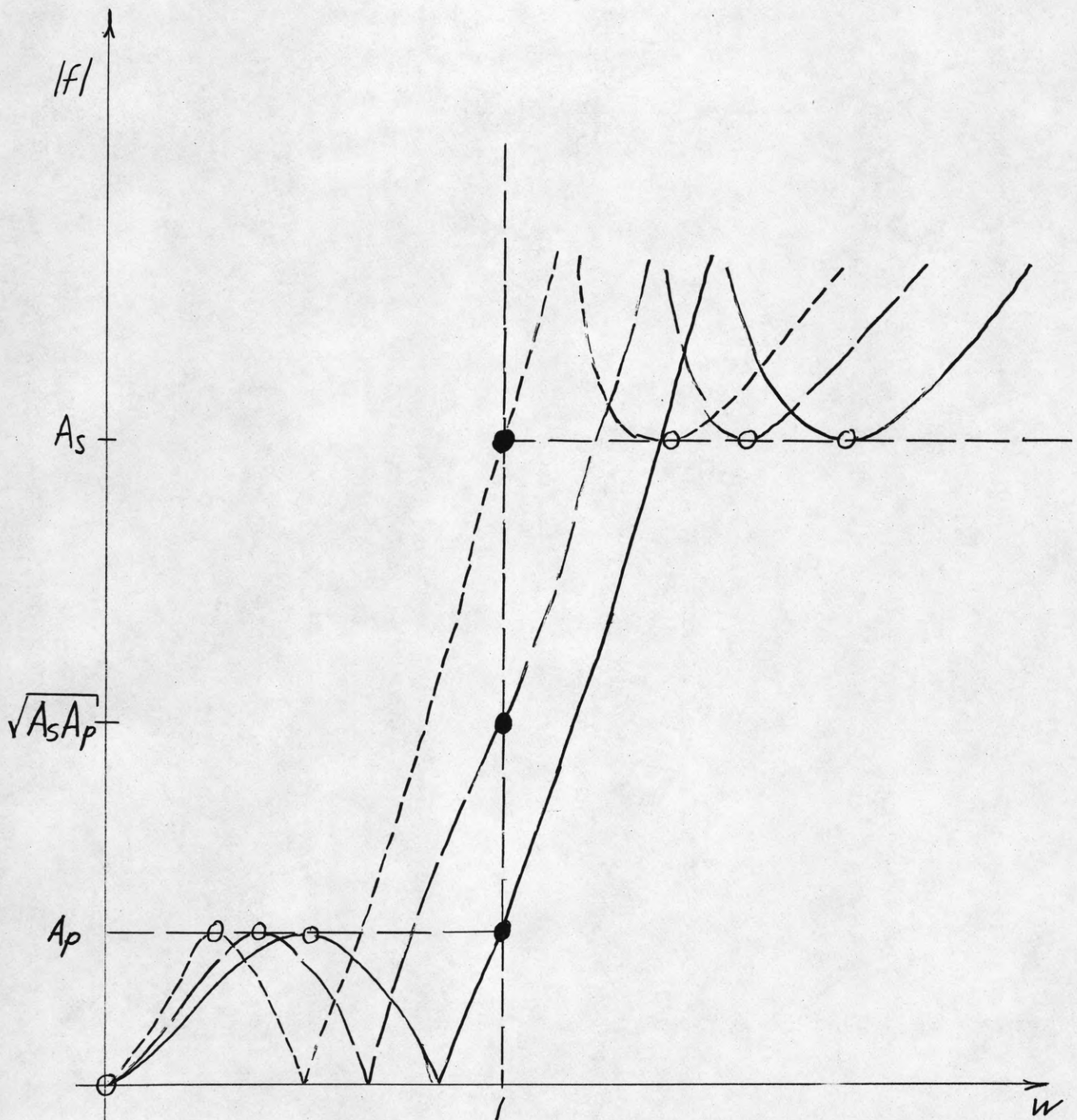


Figure 18. Behavior of filter function f for various choices of band-edge coordinate.

and zeros, the better the rate of cutoff (measured by the closeness of w_s to 1). Also, the filter function with no multiple critical points will give the best rate of cutoff of all filter functions of the same degree.

PROBLEM 3 (Equi-ripple band-pass filter function)^(2,28)

Construct a filter function f that has the following characteristics. There is a pole of order q at the origin. There are $n(1)$ poles in the lower stop-band, and the extremum ordinates in this band are A_{s1} . There are m zeros in the pass-band, and the extremum ordinates in this band are A_p . The upper stop-band has $n(2)$ poles and the extremum ordinates here have value A_{s2} . There is a pole at infinity. The band-edge coordinates are (A_p, w_1) and (A_p, w_2) . These specifications are illustrated in Figure 19.

SOLUTION:

We use Algorithm II for this problem. The weight function is w^{-q} . The initial polynomials are $p(1) = T_{2n(1)}(w/w_1)$, $p(2) = T_{2m}(\frac{2x-w_1-w_2}{w_2-w_1})$, and $p(3) = w^{2n(2)} T_{2n(2)}(w_2/w)$, where T is the Chebyshev polynomial. In the lower stop-band $v_0(1) = v_1(1) = \dots = v_{n(1)-1}(1) = 1/A_{s1}$; $v_{n(1)}(1) = 1/A_p$. In the pass-band we have $v_0(2) = v_1(2) = \dots = v_m(2) = A_p$. In the upper stop-band we have $v_0(3) = 1/A_p$; $v_1(3) = v_2(3) = \dots = v_{n(2)}(3) = 1/A_{s2}$.

This is a well-known problem in the literature; as far as the author knows this is the first general solution.

A calculation has been carried out using a Fortran

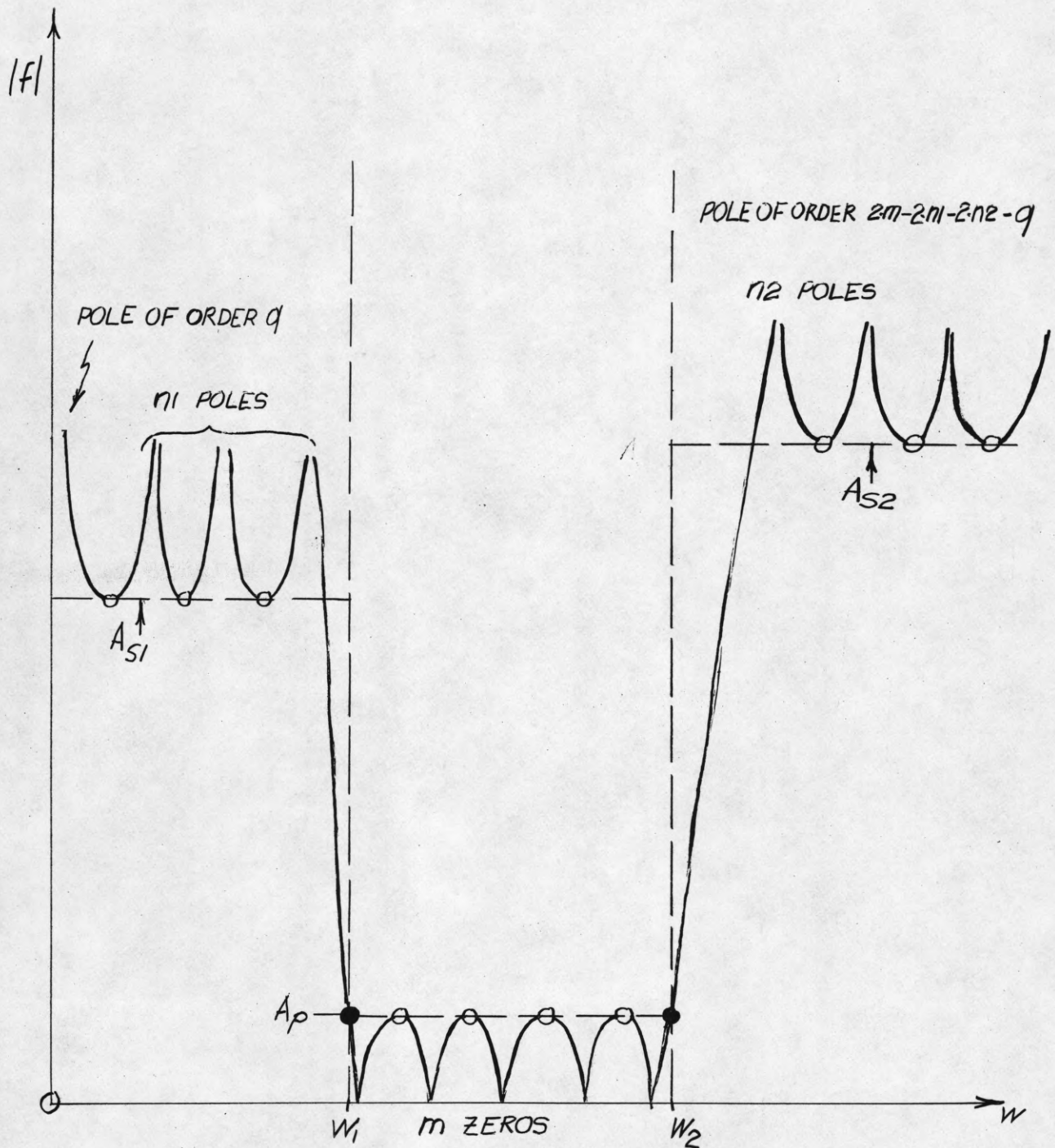


Figure 19. Typical band-pass filter function. (Problem 3).

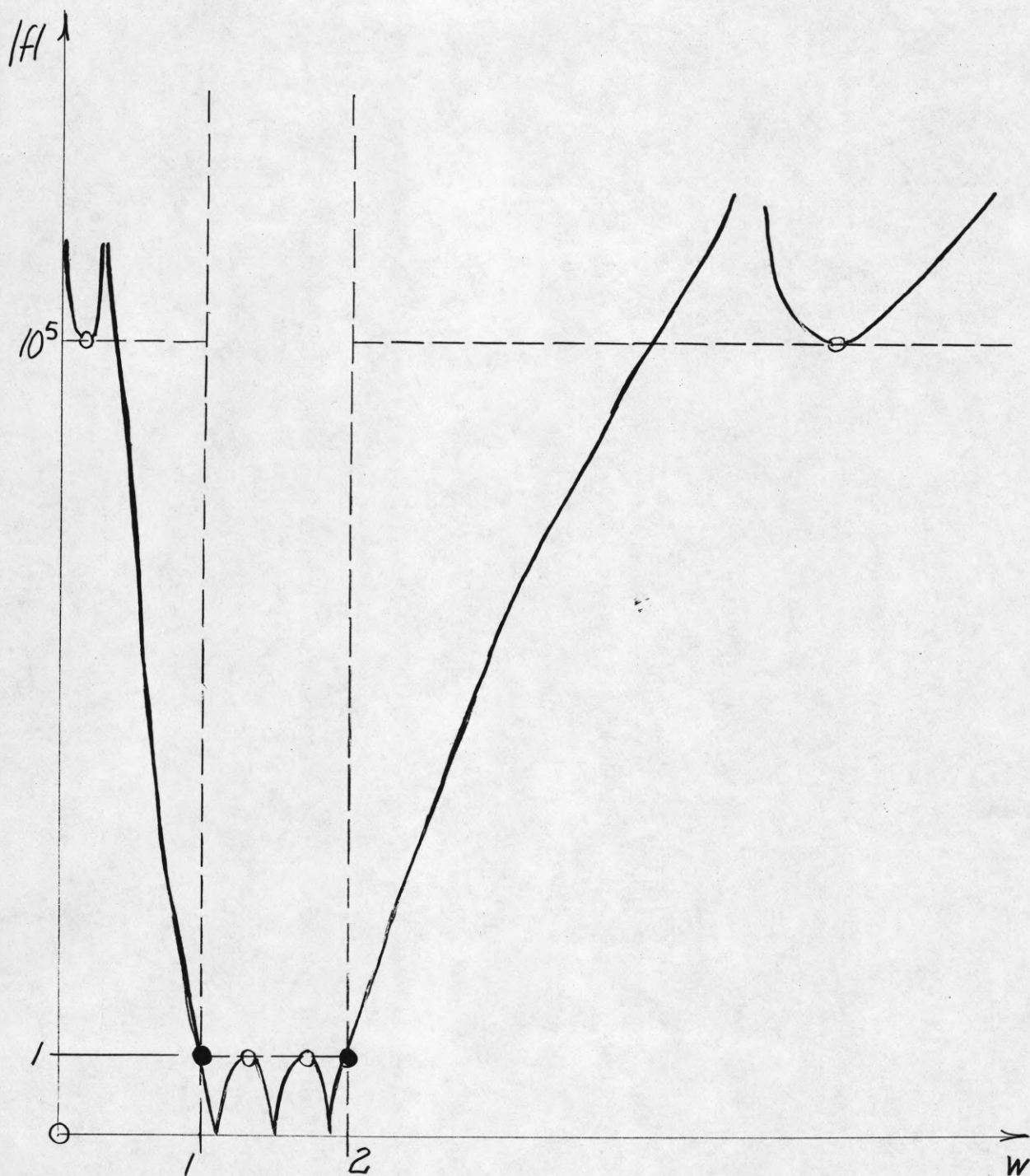


Figure 20. The band-pass function of the text.

program of the author to determine the function with the characteristic shown in Figure 20. The computed function is

$$f(w) = \frac{(-32.035396 + 54.061943w^2 - 27.030972w^4 + 4.0044245w^6)}{w(0.0021934906 - 0.25054837w^2)(-4.0353284 + 0.008832082w^2)}$$

10 iterations are required to determine this function to seven figures.

Here is an example of the use of these band-pass filter functions. In telecommunications work it is often necessary to separate various frequency bands or 'channels' carried on a common transmission path. The following set of three ideal filter characteristics in Figure 21 might be desired for such use.

Then the set of filter functions shown in Figures 22, 23, and 24 may be useful in meeting these specifications. Note that there is no symmetry about the frequency that is the geometrical mean of the upper and lower band-edge frequencies. The filter functions obtainable in Problem 3 do not have this symmetry in general. This property may result in economy.

PROBLEM 4 (Equi-ripple double band-pass filter)

Construct a filter function having equi-ripple double band-pass filter characteristics. Figure 25 indicates the characteristic desired.

SOLUTION:

We use Algorithm II for this problem. The modifications to the previous problem are so slight they are not given.

An analogous filter function class has been considered by Watanabe⁽²⁸⁾. He does not obtain equi-rippled behavior in

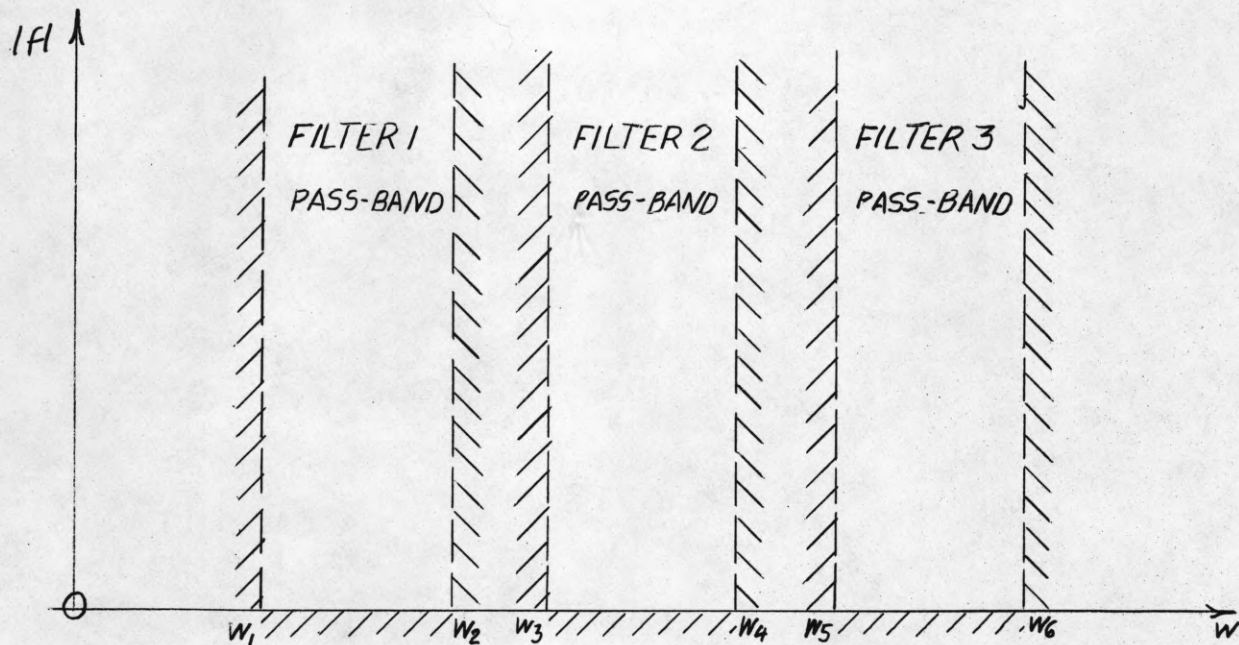


Figure 21. A set of ideal characteristics for three filters.

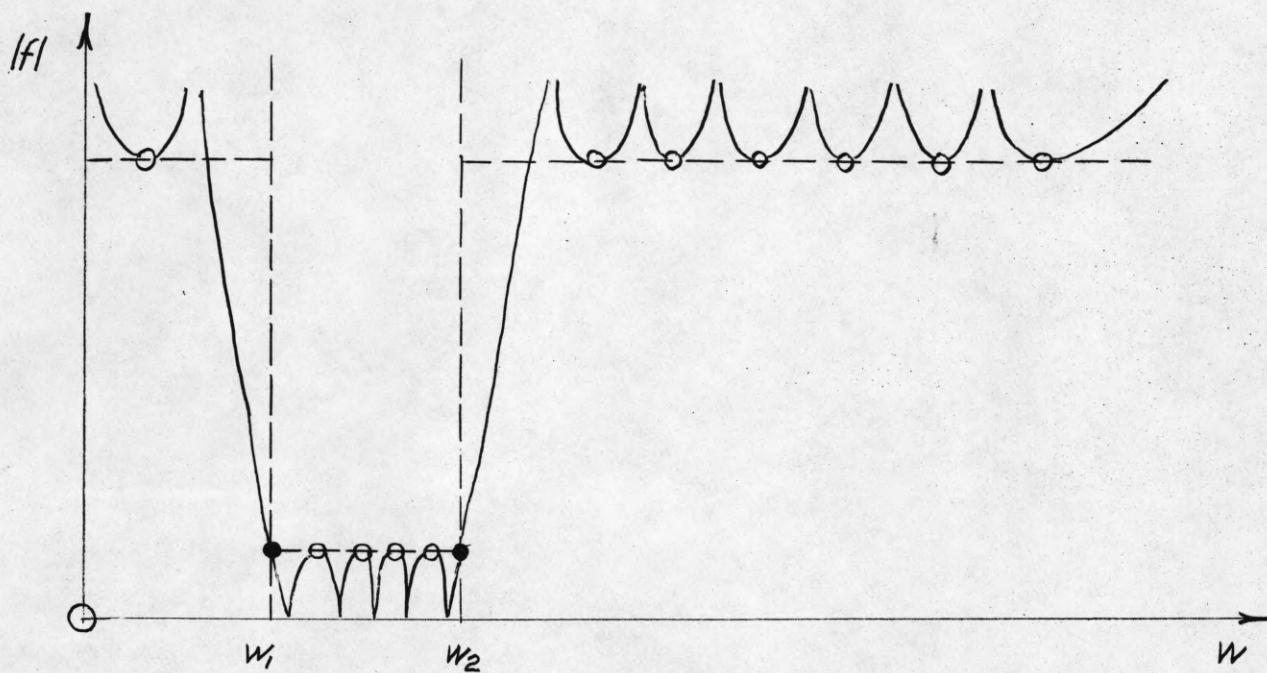


Figure 22. Lowest channel filter function.

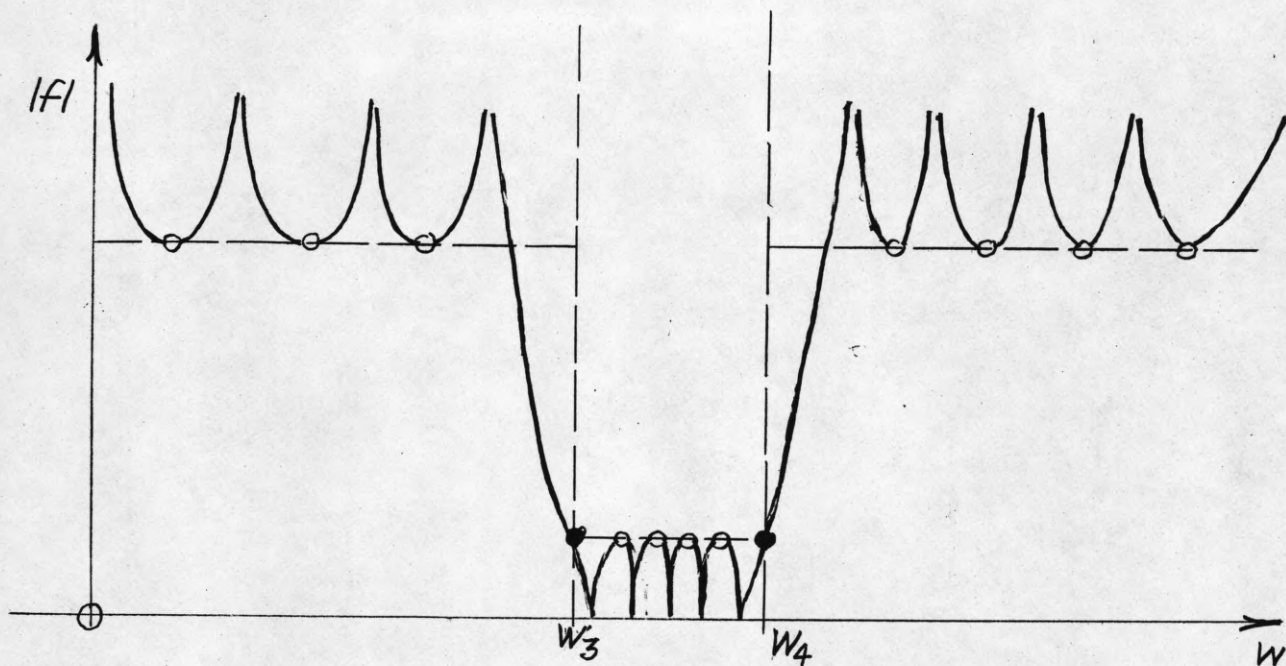


Figure 23. Middle channel filter function.

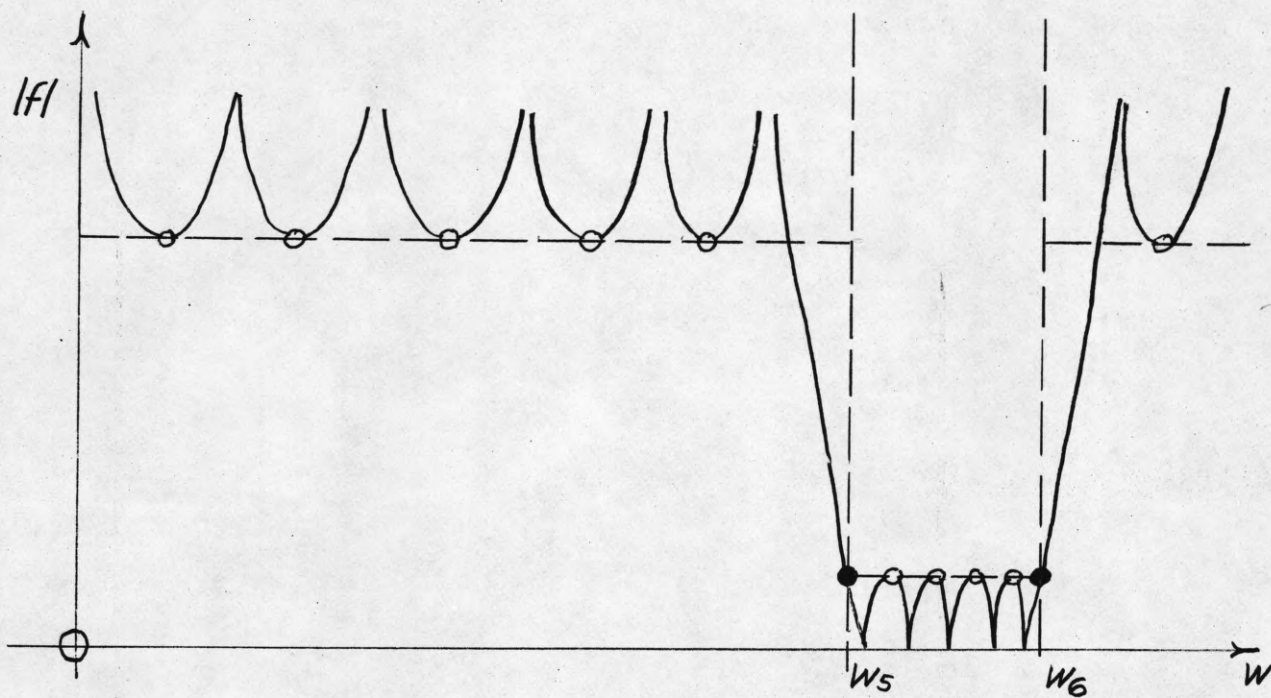


Figure 24. Highest channel filter function.

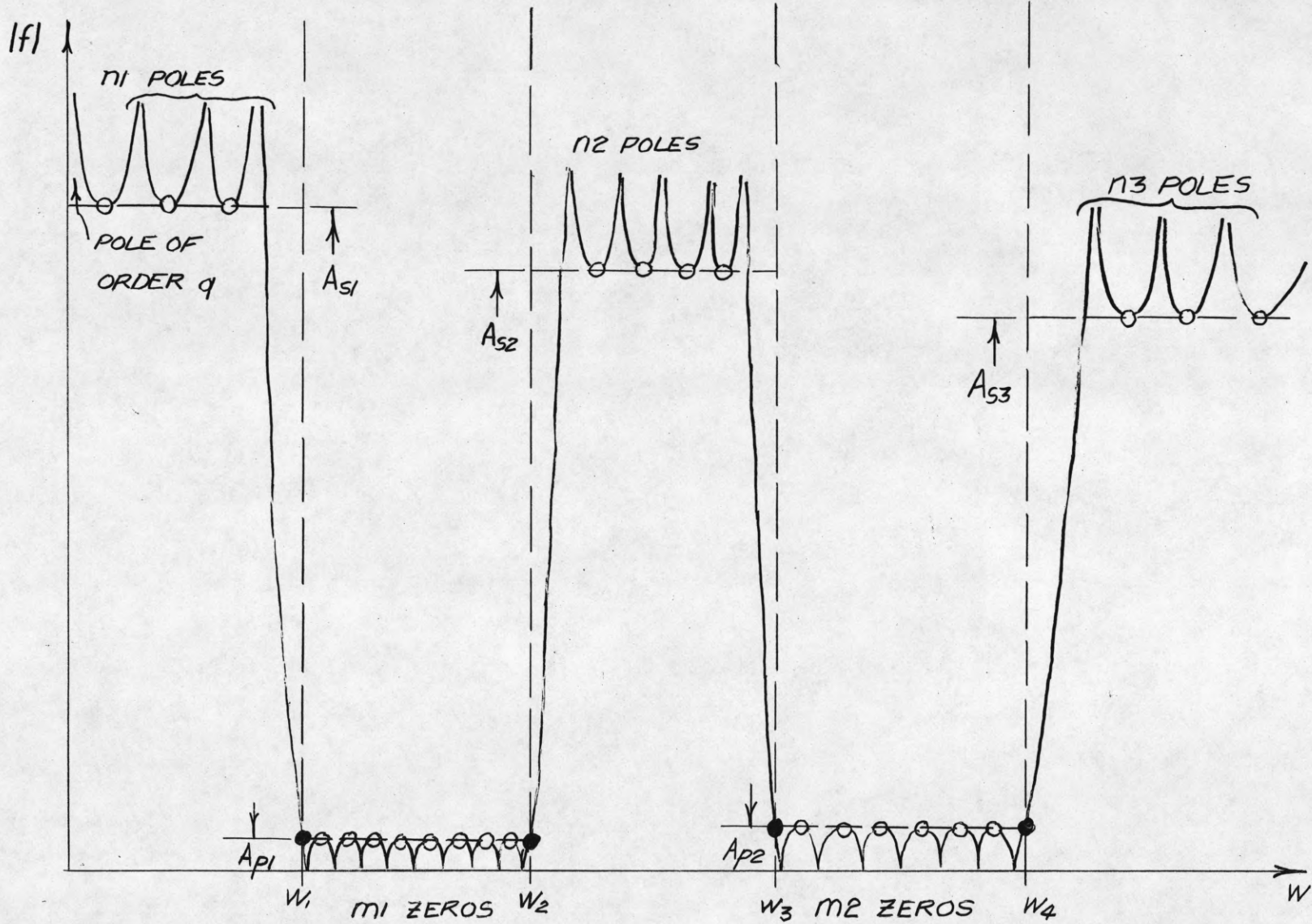


Figure 25. Equi-ripple double band-pass filter function.

both stop- and pass-bands, however.

A calculation of a filter function of this type has been carried out on the Control Data 1604 computer using a Fortran program of the author. The parameters in the calculation to be described are $q=3$, $n(1)=2$, $n(2)=4$, $n(3)=2$, $m(1)=6$, $n(2)=4$, $A_{p1}=1$, $A_{p2}=1$, $A_{s1}=10^5$, $A_{s2}=10^5$, $A_{s3}=10^5$, $w_1=1$, $w_2=2$, $w_3=3$, $w_4=4$. Figure 26 illustrates the convergence of the extremum locations and magnitudes to the final limit. After 20 iterations there is no improvement in accuracy, undoubtedly due to the round-off error and the lack of sufficient digits for more accuracy. Each iteration takes about five seconds on the 1604 computer.

Table 4 shows the maximum deviation at each iteration for the pass-band and the stop-band.

The calculated function f is

$$f = \frac{a_0 w^{20} + \dots + a_{10} w^0}{w^3 (b_0 w^{14} + \dots + b_7)}$$

where

a_0	= 5.1524452	b_0	= 0.0016536965
a_1	= -320.37250	b_1	= -0.36263285
a_2	= 8296.9196	b_2	= 216.30575
a_3	= -116833.98	b_3	= -397.37324
a_4	= 982790.47	b_4	= 3286.8673
a_5	= -5133399.5	b_5	= -13292.811
a_6	= 16851365.	b_6	= 24129.325
a_7	= -34396765.	b_7	= -13308.659
a_8	= 41920480.	b_8	= 2178.8752
a_9	= -27675077.		
a_{10}	= 7562075.7		

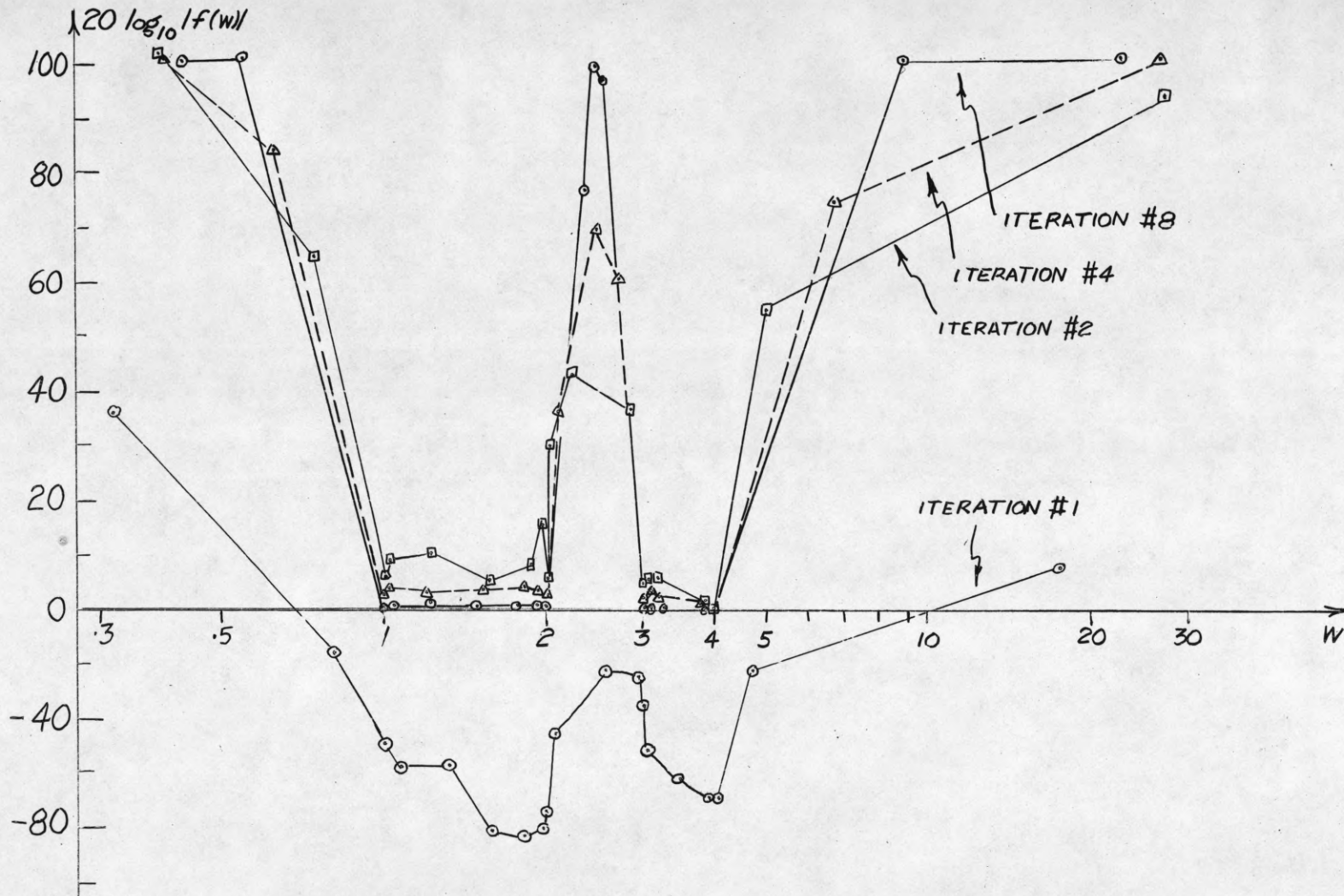


Figure 26. Extremum values for various iterations of text double band-pass filter function using Algorithm II.

TABLE 4

The maximum deviation in the stop- and pass-bands
for each iteration of the problem of the text

<u>Iteration No.</u>	$20 \log_{10} f $ at the f for which the deviation is maximum	
	<u>stop-band</u>	<u>pass-band</u>
1	-44.	-84.
2	29.	14.9
3	28.7	5.5
4	35.5	3.3
5	44.2	1.92
6	53.9	1.0
7	64.5	0.37
8	76.4	0.055
9	88.7	-0.023
10	97.6	-0.0181
11	99.94	-0.0055
12	99.9988	-0.00096
13	99.99818	-0.00016
14	99.99818	-0.000035
15	99.99984	-0.00008
16	100.0016	0.000039
20	99.99914	0.0000318
30	100.0052	-0.000035
40	99.9988	-0.000056
50	100.00967	-0.000105

It is of engineering interest to estimate the cost of a calculation of this sort, and to compare this cost with the filter cost. Computer time is about 500 dollars per hour, and hence for 20 iterations of 5 seconds duration, the cost is about 20 dollars. A program takes about a week to prepare and is used for about 50 filter designs. The programming cost per design is thus about 10 dollars. The total calculation cost is thus about 30 dollars. If each inductor costs about 10 dollars and each capacitor about 5 dollars the filter in Problem 4 costs about 300 dollars. Thus the calculation cost is about 10 percent of the filter cost. This is quite a reasonable proportion.

This completes the discussion of our four problems. The rest of this chapter is devoted to some generalizations of the previous work.

Some of the flexibility possible with these algorithms has already been discussed -- the freedom in the choice of the v_i , and in the choice of the band-edge coordinates.

It is not difficult to take account of a finite functional value at infinity; only slight changes in the algorithms are necessary.

Suppose the characteristic of Figure 27 is desired. In this figure the usual notational practice is followed. Ordinarily, one of the two solid o's at the band-edge must be eliminated. However, if the given extremum ordinate at the right hand extremum, for instance, is removed, then it is possible to obtain the functional behavior shown in Figure 28,

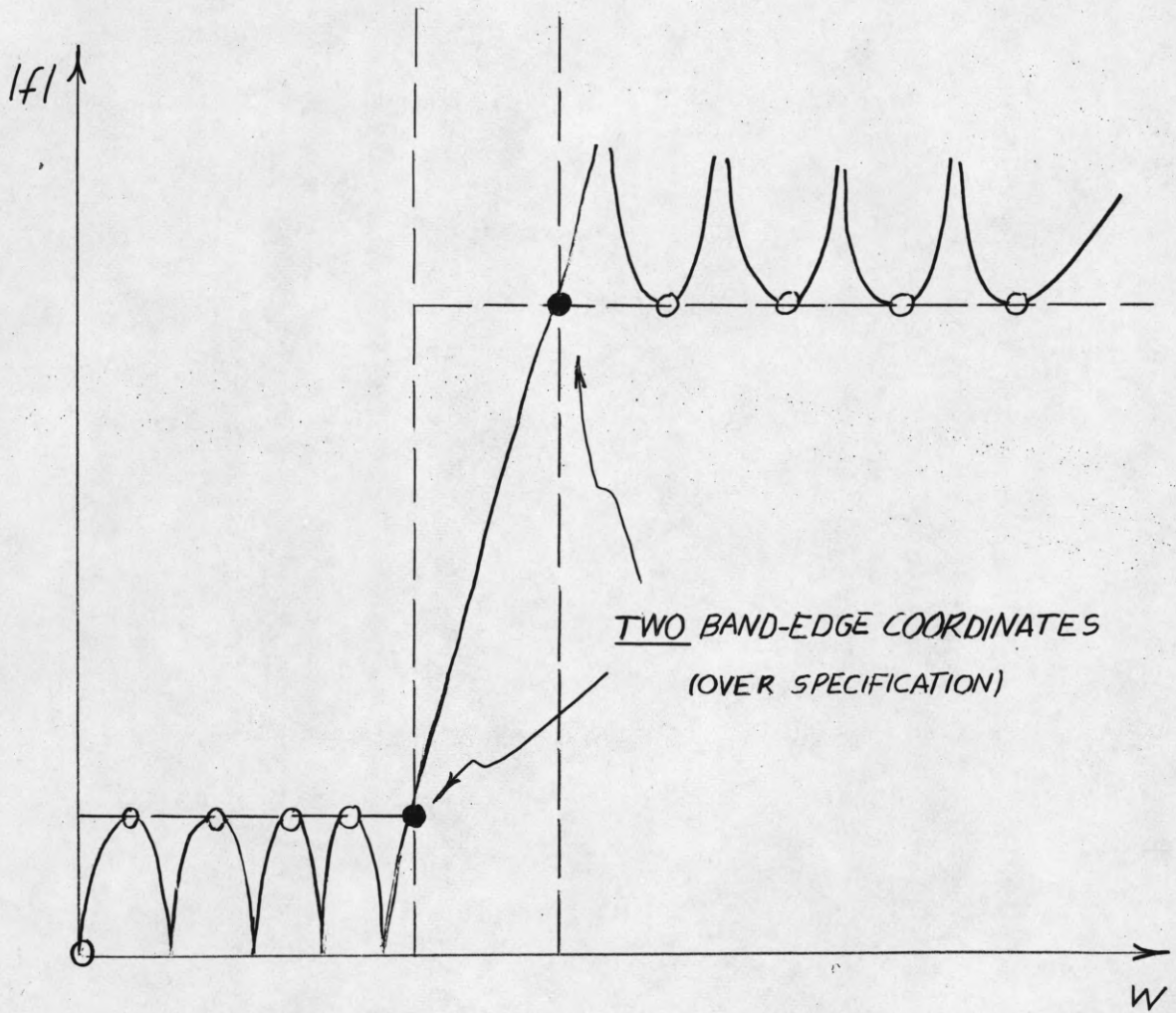


Figure 27 Proposed specification

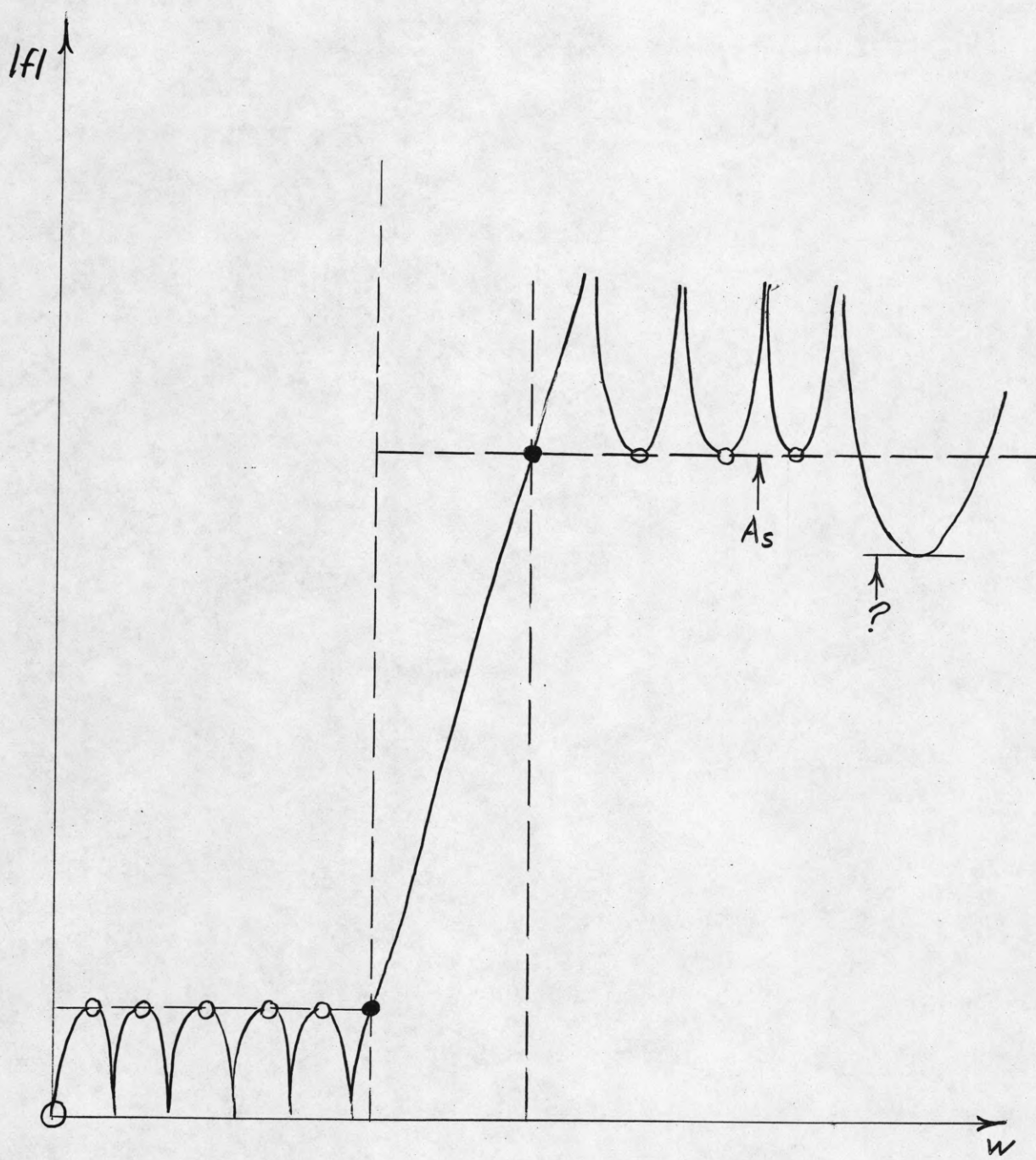


Figure 28. Illustration of characteristic possible if when two band edge coordinates are fixed.

retaining the two band-edge coordinates. The magnitude at the right hand extremum is, of course, not known in advance. The resulting filter may or may not be interesting.

Another method of obtaining analogous behavior, that illustrated in Figure 29, is also of interest. The final extremum ordinates are not known in advance, but they are to be equal. But the v_i must be known at each iteration. The method proposed consists in calculating the actual extremum magnitudes at each iteration and then choosing a new v_i that is some mean, of these extremum magnitudes. The method of the previous example is followed for the calculation of the interpolator.

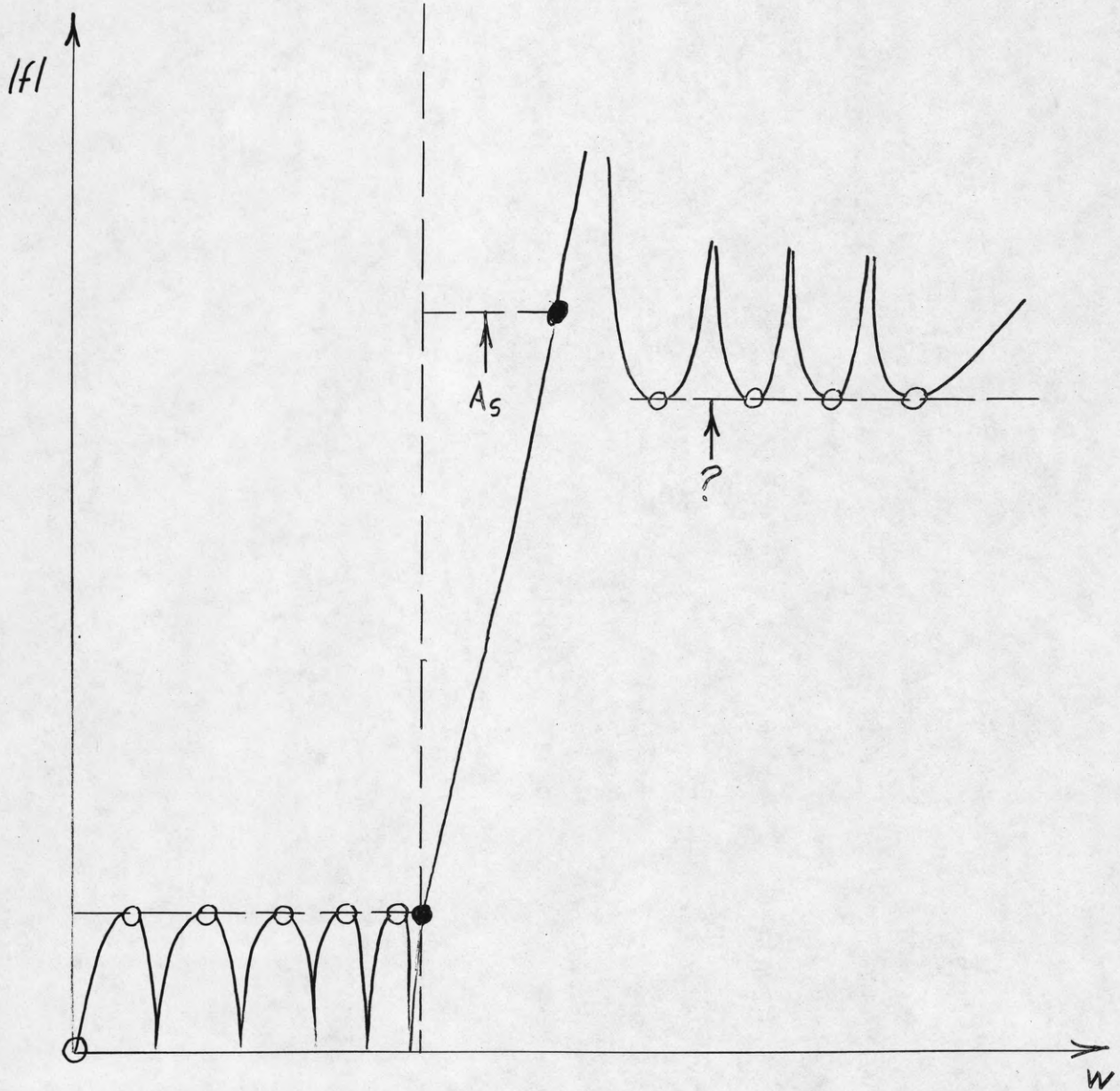


Figure 29. Text example of possible modifications.

CHAPTER 6

ADVANTAGES AND DISADVANTAGES OF THESE METHODS

The obvious disadvantage is that these methods are numerical and require a great deal of computation. Thus, the use of a digital computer is practically mandatory. If a computer is available, however, these algorithms and methods are convenient. Since the operations are elementary and no complicated functions are required, the programming is simple. The analysis of an example shows that the relative computing cost is small when compared with the filter cost.

These methods are natural and do not require any tricks. The engineer is able to keep track of the various parameters. No use is made of advanced function theory. Thus these methods may be used by those with a familiarity with modern network synthesis.

A disadvantage not yet encountered in practice but which is nevertheless real is that convergence is not guaranteed.

The main objection to these methods is one shared by more or less all approximation procedures -- the problem attacked is removed from the world of the filter designer. Our methods treat only amplitude-frequency characteristics and cannot incorporate the effect of component losses, etc. This is a real limitation.

CHAPTER 7

CONCLUSION

A numerical procedure consisting of two algorithms has been presented for generating useful filter amplitude-frequency transfer functions. Examples have been given of the use of these methods in generating four useful classes of filter functions. Modifications to permit flexibility in the design of useful filter functions have been described.

If a digital computer is available for calculation, it appears that the computation cost is about ten percent of the filter cost. Thus these methods should prove useful in the designing of filters.

BIBLIOGRAPHY

1. Achieser, N.I., Theory of Approximation, Ungar Publishing Co., New York, 1956.
2. Baum, R.F., Design of Unsymmetrical Band-pass Filters, Institute of Radio Engineers Transactions on Circuit Theory, Vol. CT-4, 33-40, June 1957.
3. Bedrosian, S.D., Elliptic Functions in Network Synthesis, Journal of the Franklin Institute, Vol. 271, No. 1, pp. 12-30.
4. Belevitch, V., Tchebycheff Filters and Amplifiers, Wireless Engineer, Vol. 29, pp. 106-110, April, 1952.
5. Bernstein, S., Lecons sur les Proprietes Extremales et la Meilleure Approximation des Fonctions Analytiques d'une Variable Reelle, Gauthier-Villars, Paris, 1926.
6. Bresler, A.B., On the Approximation Problem in Network Synthesis, Proceedings of the Institute of Radio Engineers, Vol. 40, No. 12, pp. 1724, December, 1952.
7. Cauer, W., Synthesis of Linear Communication Networks, McGraw-Hill, New York, 1958.
8. Darlington, S., Synthesis of Reactance Four-Poles, Journal of Mathematics and Physics, Vol. 18, pp. 257-353, 1939.
9. Favard, J., Cours d'Analyse, Vol. 2, Gauthier-Villars, Paris, 1960.
10. Green, E., Synthesis of Ladder Networks to give Butterworth or Chebyshev Response, Institution of Electrical Engineers (GB) Monograph 88, January 15, 1954.
11. Grossman, A., Synthesis of Tchebysheff Parameter Symmetrical Filters, Proceedings of the Institute of Radio Engineers, Vol. 45, No. 4, pp. 454, April, 1957.
12. Guillemin, E.A., Synthesis of Passive Networks, Wiley and Sons, New York, 1957.
13. Hildebrand, F.B., Introduction to Numerical Analysis, McGraw-Hill, New York, 1956.
14. Langer, R., On Numerical Approximation, University of Wisconsin Press, Madison, 1959.
15. Maclean, D.J.H., Cocci Insertion Loss Design of Electric Filters, Stanford Electronics Laboratories, Technical Report No. 2, Stanford, June 7, 1956.

16. Matthai, G.L., Conformal Mappings for Filter Synthesis, Proceedings of the Institute of Radio Engineers, Vol. 41, No. 11, pp. 1658, November, 1953.
17. Matthai, G.L., Filter Transfer Function Synthesis, Proceedings of the Institute of Radio Engineers, Vol. 41, No. 3, pp. 377, March, 1953.
18. Milort, W., Theory and Design Data for the Synthesis of Tchebysheff Parameter Filters, Phillips Telecommunications Review, Vol. 20, No. 4, pp. 158-185, May, 1959.
19. Novodrovskii, and Pinsker, Protsess Uravnivaniya Maksimymov, Uspechi Matem. Nauk, Vol. 6, pp. 174-181, 1951.
20. Papoulis, A., Approximation Problem in Filter Design, Institute of Radio Engineers National Convention Record, Vol. 5, No. 4, pp. 175, 1957.
21. Piloty, H., Zolotareffsche Rationale Funktionen, Z. Angew. Math. Mech., Vol. 34, pp. 175-189, 1956.
22. Pokrovskii, On a Class of Polynomials with Extremal Properties, Translations of the American Mathematical Society, Vol. 19, pp. 199-220, 1962.
23. Remez, E., Sur le Calcul Effectif des Polynomes de Tschebysheff, Comptes Rendus, Academie des Sciences, Paris, Vol. 199, pp. 337-340, 1934.
24. Scott, R.E., Network Synthesis by Potential Analogs, Proceedings of the Institute of Radio Engineers, Vol. 40, No. 8, pp. 970, August, 1952.
25. Timan, A.F., Teoriya priblisheniya funktsii deistvitelnovo peremenenovo, Fizmatgiz, Moscow, 1960.
26. Todd, J. (ed.), A Survey of Numerical Analysis, McGraw-Hill, New York, 1962.
27. Van Valkenburg, M.E., Introduction to Modern Network Synthesis, Wiley and Sons, New York, 1960.
28. Watanabe, H., Approximation Theory for Filter-Networks, Institute of Radio Engineers Transactions on Circuit Theory, Vol. CT-9, No. 3, pp. 341-356, September, 1961.
29. Winkler, S., Approximation Problems of Network Synthesis, Institute of Radio Engineers Transactions on Circuit Theory, Vol. CT-2, No. 3, pp. 5-21, September, 1954.
30. Zuhovickii, S.I., On the Approximation of Real Functions in the sense of P.L. Tchebysheff, Translations of the American Mathematical Society, Vol. 19, pp. 221-252, 1962.