NASA TECHNICAL NOTE

NASA TN D-6709
c.1

NASA TN D-6709

DESIGN OF RECURSIVE DIGITAL FILTERS HAVING SPECIFIED PHASE AND MAGNITUDE CHARACTERISTICS

by

Robert E. King
Langley Research Center

and

Gregory W. Condon
Langley Directorate,
U.S. Army Air Mobility R&D Laboratory

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • APRIL 1972

1. Report No.
NASA TN D-6709

2. Government Accession No.

3. Recipient's Catalog No.

4. Title and Subtitle
DESIGN OF RECURSIVE DIGITAL FILTERS HAVING SPECIFIED PHASE AND MAGNITUDE CHARACTERISTICS

5. Report Date
April 1972

6. Performing Organization Code

7. Author(s)
Robert E. King; and Gregory W. Condon, Langley Directorate, U.S. Army Air Mobility R&D Laboratory

8. Performing Organization Report No.
L-8162

9. Performing Organization Name and Address
NASA Langley Research Center
Hampton, Va. 23365

10. Work Unit No.
760-72-03-01

11. Contract or Grant No.

12. Sponsoring Agency Name and Address
National Aeronautics and Space Administration
Washington, D.C. 20546

13. Type of Report and Period Covered
Technical Note

14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

A method for a computer-aided design of a class of optimum filters, having specifications in the frequency domain of both magnitude and phase, is described. The method, an extension to the work of Steiglitz, uses the Fletcher-Powell algorithm to minimize a weighted squared magnitude and phase criterion. Results using the algorithm for the design of filters having specified phase as well as specified magnitude and phase compromise are presented.

17. Key Words (Suggested by Author(s))
Digital filter
Computer-aided design
Phase specification

18. Distribution Statement
Unclassified – Unlimited

19. Security Classif. (of this report)
Unclassified

20. Security Classif. (of this page)
Unclassified

21. No. of Pages
39

22. Price*
$3.00

*For sale by the National Technical Information Service, Springfield, Virginia 22151

DESIGN OF RECURSIVE DIGITAL FILTERS HAVING SPECIFIED PHASE AND MAGNITUDE CHARACTERISTICS

By Robert E. King
Langley Research Center

and

Gregory W. Condon
Langley Directorate, U.S. Army Air Mobility R&D Laboratory

SUMMARY

A method for a computer-aided design of a class of optimum filters, having specifications in the frequency domain of both magnitude and phase, is described. The method, an extension to the work of Steiglitz, uses the Fletcher-Powell algorithm to minimize a weighted squared magnitude and phase criterion. Results using the algorithm for the design of filters having specified phase as well as specified magnitude and phase compromise are presented.

INTRODUCTION

Recursive filters, wherein the output sequence is both a function of the input as well as past output samples, are commonly used in digital signal processing and analysis. Such digital filters in many applications offer distinct advantages of precision and versatility over their continuous or analog counterparts. There exist a number of design procedures for implementing digital filters (see ref. 1) each one of which strives to attain some analogy between discrete and continuous systems. Transform methods such as the matched-z, bilinear-z, and standard-z which lead to specific property invariances are available (see ref. 2) to the designer familiar with continuous filter design.

For frequency-domain synthesis (see refs. 3 and 4), realization is normally by means of cascade or parallel combinations of pole and zero pairs in the complex plane. The synthesis problem is, in fact, reduced to one of approximation since the filter topology is generally specified. In none of the available design procedures, which can yield filters having excellent magnitude-frequency characteristics, however, do the resultant filters, in themselves, have particularly useful phase characteristics. Indeed, in striving for particular magnitude characteristics by using any of the available design methods, there is no control over the filter phase properties.

In practice, it is often desirable to specify a digital filter in the frequency domain by its phase (see ref. 5) or even a compromise between magnitude and phase. The procedure in this paper meets these requirements through the use of an iterative computer-aided design leading to an optimum set of parameters for a specified filter topology and is an extension of the technique described by Steiglitz (see ref. 6) for determining the optimum coefficients of a cascade filter having magnitude specifications alone. The extension makes possible the design of a new class of digital filters having the prescribed phase characteristics.

SYMBOLS

A	filter multiplier
D_k^i	denominator of ith stage of $H(z)$ at Ω_k
E_k^M	magnitude error at Ω_k
E_k^ϕ	phase error at Ω_k
$\vec{e}_k$	error vector at Ω_k
$\partial\vec{e}_k/\partial A$	derivative of error vector at Ω_k with respect to zero frequency gain
f_k	frequency at kth specification point, Hz
f_s	sampling frequency, Hz
$H(z)$	unity gain discrete transfer function
$\lvert H_k\rvert$	magnitude of $H(z)$ at Ω_k
$\overline{H}_k$	conjugate of $H(z)$ at Ω_k
$\partial\lvert H_k\rvert/\partial\vec{p}$	gradient vector of magnitude of $H(z)$ at Ω_k with respect to parameter vector
$I(\ )$	imaginary part of quantity
$i,\ldots,N$	denotes filter stage

$\vec{J}_k$	Jacobian at Ω_k, $\left[A^* \frac{\partial \lvert H_k \rvert}{\partial \vec{p}} \;\vdots\; \frac{\partial \phi_k}{\partial \vec{p}}\right]$
k	sample point
M_k	specification magnitude at Ω_k
N_k^i	numerator of ith stage of $H(z)$ at Ω_k
$\vec{p}$	parameter vector
$\vec{p}_i$	set of filter parameters for the ith stage, a_i, b_i, c_i, and d_i
$q_1^i(k)$	first system state of ith stage at kth sample point
$q_2^i(k)$	second system state of ith stage at kth sample point
$R(\ )$	real part of quantity
$u_i(k)$	input to ith stage at kth sample point
V	criterion functional, that is, $V(A,\vec{p})$
V_k	criterion functional at Ω_k, that is, $V_k(A,\vec{p})$
$\hat{V}$	reduced criterion functional, that is, $V(A^*,\vec{p})$
$\partial V/\partial A$	slope of criterion functional with respect to zero frequency gain
$\partial V_k/\partial \vec{e}_k$	gradient vector of criterion functional at Ω_k with respect to error vector at Ω_k
$\vec{W}_k$	weighting matrix at Ω_k
W_k^M	magnitude weighting at Ω_k
W_k^ϕ	phase weighting at Ω_k
$w^i(k)$	dummy variable of ith stage at kth sample point

$Y(z)$	digital filter discrete transfer function
$y_i(k)$	output of ith stage at kth sample point
z	transform variable
z_k	discrete transform variable at Ω_k, $e^{j\pi\Omega_k}$
θ_k	specification phase at Ω_k, radians
λ	collective phase weight
ϕ_k	phase of $H(z)$ at Ω_k, radians
$\partial\phi_k/\partial\vec{p}$	gradient vector of phase of $H(z)$ at Ω_k with respect to parameter vector
Ω_k	fractional frequency at kth specification point

An asterisk on a symbol denotes an optimum value. A circumflex denotes optimization with respect to A. A superscript T denotes the transpose.

DISCUSSION

The Filter Form

The fundamental advantages of the N-stage cascade canonical form of recursive digital filter whose signal flow graph is shown in figure 1 and which is described by the product operator

$$\left.\begin{aligned} Y(z) &= A\prod_{i=1}^{N}\frac{1 + a_i z^{-1} + b_i z^{-2}}{1 + c_i z^{-1} + d_i z^{-2}} \\ Y(z) &= AH(z) \end{aligned}\right\} \tag{1}$$

are (1) its relative insensitivity to perturbations in the denominator coefficients, an important consideration in digital filters, especially of high order and particularly where finite register lengths (see ref. 1) are involved; (2) its simplicity of implementation; and (3) the simplicity of factoring the filter operator to determine its roots. This form has found extensive application in practical filters for signal processing, and a version employing serial arithmetic (ref. 7) is commercially available.

For completeness, an alternative description of the filter is given in terms of the system states q_1^i and q_2^i and clearly demonstrates the recursive nature of the filter. The set of difference equations describing the filter and required in developing a computer algorithm is presented. Thus, for the ith stage in figure 1 at the kth sample point

$$w^i(k) = A_i u_i(k) - c_i q_1^i(k) - d_i q_2^i(k)$$

$$q_1^i(k + 1) = w^i(k)$$

$$q_2^i(k + 1) = q_1^i(k)$$

$$y_i(k) = w^i(k) + a_i q_1^i(k) + b_i q_2^i(k)$$

where

$$u_i(k) = y_{i-1}(k)$$

is the input to the ith stage and is identical to the output of the (i - 1) stage and

$$A_i = \begin{cases} A & (i = 1) \\ 1 & (i \neq 1) \end{cases}$$

The Synthesis Problem

The design problem considered in this paper can be stated as follows: When the magnitude and phase specifications (M_k and θ_k, respectively) at the kth fractional Nyquist frequencies $\Omega_k = 2f_k/f_s$ (where f_s is the sampling frequency in Hz) are known, determine the set of optimum parameters $\vec{p}^*$ of an N-stage cascade filter having the form of equation (1) so that the resultant digital filter will have a minimum sum squared magnitude and phase error for all specified frequencies.

By constraining the filter topology, the optimum synthesis problem becomes one of parametric optimization with respect to a given criterion of fit. The composite criterion which can weight the magnitude and phase requirements independently and as functions of frequency is chosen as the inner product

$$V(A,\vec{p}) = \sum_k \left\langle \vec{e}_k, \vec{W}_k \vec{e}_k \right\rangle = \sum_k V_k \tag{2}$$

where

$$\vec{e}_k = \begin{bmatrix} A\left|H_k\right| - M_k \\ \\ \phi_k - \theta_k \end{bmatrix} = \begin{bmatrix} E_k^M \\ \\ E_k^\phi \end{bmatrix}$$

is the error vector and

$$\vec{W}_k = \begin{bmatrix} W_k^M & 0 \\ \\ 0 & \lambda W_k^\phi \end{bmatrix}$$

is the diagonal weighting matrix. Clearly, $V\left(A,\vec{p}\right)$ is a nonlinear function of the parameter vector $\vec{p} = \left(a_1,b_1,c_1,d_1,\ldots,a_N,b_N,c_N,d_N\right)^T$, which involves the $4N$ filter coefficients, and of the filter multiplier A.

The Minimization Algorithm

Through formal differentiation of the criterion function (eq. (2)) with respect to the multiplier A, the minimization procedure can be slightly simplified to that of finding the minimum of a reduced functional $\hat{V}\left(\vec{p}\right) = V\left(A^*,\vec{p}\right)$ involving only $4N$ parameters. Thus

$$\frac{\partial V}{\partial A} = \sum_k \left\langle \frac{\partial \vec{e}_k}{\partial A}, \frac{\partial V_k}{\partial \vec{e}_k} \right\rangle = 2 \sum_k \left[\left|H_k\right| W_k^M \;\vdots\; 0\right] \vec{e}_k$$

and $\partial V/\partial A = 0$ implies

$$2 \sum_k \left|H_k\right| W_k^M \left(A^*\left|H_k\right| - M_k\right) = 0$$

or

$$A^* = \frac{\sum_k \left|H_k\right| W_k^M M_k}{\sum_k \left|H_k\right|^2 W_k^M} \tag{3}$$

An additional necessary condition for existence of an extremum is that the gradient vector be zero; thereby, the optimum parameter vector $\vec{p}^*$ is obtained. From equation (2)

$$\frac{\partial \hat{V}}{\partial \vec{p}} = 2 \sum_{k} \left\langle \vec{J}_k, \vec{W}_k \vec{e}_k \right\rangle \tag{4}$$

where the (4N × 2) Jacobian $\vec{J}_k$ is

$$\vec{J}_k^T = \nabla_{\vec{p}} \vec{e}_k = \left[A^* \frac{\partial |H_k|}{\partial \vec{p}} \;\vdots\; \frac{\partial \phi_k}{\partial \vec{p}} \right]^T \tag{5}$$

Clearly, each element of the gradient vector is the sum of two weighted functions of the magnitude and phase error. By writing

$$|H_k|^2 = H_k \overline{H}_k$$

where $\overline{H}_k$ is the conjugate of H_k evaluated at the fractional frequency Ω_k, it is readily shown (see ref. 6), where $\vec{p}_i$ is the set of filter parameters for the ith stage, that

$$\frac{\partial |H_k|}{\partial \vec{p}_i} = \frac{1}{|H_k|} R \left(\overline{H}_k \frac{\partial H_k}{\partial \vec{p}_i} \right)$$

For the cascaded filter topology in terms of the elements of $\vec{p}_i$,

$$\frac{\partial |H_k|}{\partial a_i} = |H_k| R \left(\frac{z_k^{-1}}{N_k^i} \right)$$

$$\frac{\partial |H_k|}{\partial b_i} = |H_k| R \left(\frac{z_k^{-2}}{N_k^i} \right)$$

$$\frac{\partial |H_k|}{\partial c_i} = -|H_k| R \left(\frac{z_k^{-1}}{D_k^i} \right)$$

and

$$\frac{\partial |H_k|}{\partial d_i} = -|H_k| R \left(\frac{z_k^{-2}}{D_k^i} \right)$$

where, with $z_k = e^{j\pi\Omega_k}$,

$$N_k^i = N^i(z_k) = 1 + a_i z_k^{-1} + b_i z_k^{-2}$$

and

$$D_k^i = D^i(z_k) = 1 + c_i z_k^{-1} + d_i z_k^{-2}$$

By letting

$$H_k = |H_k| e^{j\phi_k}$$

it follows that

$$\phi_k = I(\log_e H_k)$$

whence

$$\frac{\partial \phi_k}{\partial \vec{p}} = I\left(\frac{\partial}{\partial \vec{p}} \log_e H_k\right) = I\left(\frac{1}{H_k} \frac{\partial H_k}{\partial \vec{p}}\right)$$

which takes on a particularly simple form for the cascade topology. For the ith stage parameters, in fact,

$$\frac{\partial \phi_k}{\partial a_i} = I\left(\frac{z_k^{-1}}{N_k^i}\right)$$

$$\frac{\partial \phi_k}{\partial b_i} = I\left(\frac{z_k^{-2}}{N_k^i}\right)$$

$$\frac{\partial \phi_k}{\partial c_i} = -I\left(\frac{z_k^{-1}}{D_k^i}\right)$$

and

$$\frac{\partial \phi_k}{\partial d_i} = -I\left(\frac{z_k^{-2}}{D_k^i}\right)$$

The special case of a one-stage (N = 1) filter is illustrated. Here

$$H_k = A\,\frac{1 + az_k^{-1} + bz_k^{-2}}{1 + cz_k^{-1} + dz_k^{-2}}$$

$$\hat{V} = \sum_k \left(A^*\left|H_k\right| - M_k\right)^2 W_k^M + \lambda \sum_k \left(\phi_k - \theta_k\right)^2 W_k^\phi$$

and

$$\frac{\partial \hat{V}}{\partial a} = 2\sum_k \left(E_k^M W_k^M \frac{\partial\left|H_k\right|}{\partial a} + \lambda E_k^\phi W_k^\phi \frac{\partial \phi_k}{\partial a}\right) = \sum_k \left[Q_k^M R\left(\frac{z_k^{-1}}{N_k^i}\right) + \lambda R_k^\phi I\left(\frac{z_k^{-1}}{N_k^i}\right)\right]$$

Similarly,

$$\frac{\partial \hat{V}}{\partial b} = \sum_k \left[Q_k^M R\left(\frac{z_k^{-2}}{N_k^i}\right) + \lambda R_k^\phi I\left(\frac{z_k^{-2}}{N_k^i}\right)\right]$$

$$\frac{\partial \hat{V}}{\partial c} = \sum_k \left[Q_k^M R\left(\frac{z_k^{-1}}{D_k^i}\right) + \lambda R_k^\phi I\left(\frac{z_k^{-1}}{D_k^i}\right)\right]$$

$$\frac{\partial \hat{V}}{\partial d} = \sum_k \left[Q_k^M R\left(\frac{z_k^{-2}}{D_k^i}\right) + \lambda R_k^\phi I\left(\frac{z_k^{-2}}{D_k^i}\right)\right]$$

where

$$Q_k^M = 2E_k^M W_k^M \left|H_k\right|$$

and

$$R_k^{\phi} = 2E_k^{\phi}W_k^{\phi}$$

are the weighted errors. It is obvious that the frequency intervals of the input data (specifications) need not be uniform and may, in fact, be intentionally unequal to allow for nonuniform frequency weighting.

Complementary Root Reflection and Stability

In deriving the frequency response of a discrete operator by letting z_k lie on the unit circle Γ, it is possible to take advantage of a unique property of the discrete transform pertaining to its magnitude when a root lying outside the unit circle is imaged or reflected into the unit circle. It is easy to show that the magnitude of a phasor $z - z_0$, where z_0 is a root of the discrete transform lying outside the unit circle, is equal to

$$\left|z - z_0\right| = \left|z_0\right| \left|z - \frac{1}{z_0}\right|; \; z \in \Gamma$$

Since z_0 has been assumed to be outside the unit circle, $1/z_0$ must be inside, the term $|z_0|$ correcting for magnitude changes. Thus, if in the optimization procedure a pole should stray outside the unit circle and thereby lead to an unstable filter, root reflection guarantees stability with no magnitude change. There is no analogous simple identity for the phase of a reflected root. Experience with the procedure has shown that provided the design requirements can be met by means of a stable filter, that is, that a feasible solution exists, an optimum will indeed be found through repeated application of root reflection.

The Computer Algorithm

A complete listing of the filter design algorithm, which is an adaptation of the program written by Steiglitz, is given in the appendix. The main program is termed STGZ3 which calls four principal subroutines: (1) FUNCT performs the functional and gradient computation for each iteration as well as putting out the final optimum parameters and plots, (2) FLPWL is a Fletcher-Powell conjugate gradient routine, (3) INSIDE computes root reflection, and (4) ROOTS determines the poles and zeros of the filter. Single-precision arithmetic has been employed.

When minimization of the functional has been attained in the first pass or the minimization algorithm has iterated 300 times, a test is made to ascertain that all the roots are within the unit circle, a necessary requirement for the poles for stability reasons and for the zeros to insure minimum phase. If the design should result in an unstable configuration, the roots are reflected about the unit circle and minimization is resumed in a second pass. If a minimum does indeed exist and all the roots then lie within the unit circle, the program computes and prints out the frequency response and commences plotting.

Minimization is deemed to be achieved when the absolute difference in functionals between successive iterations $\epsilon = \left|\hat{V}_{new} - \hat{V}_{old}\right|$ or the norm of the gradient vector falls below preassigned limits. Convergence is generally fast for magnitude or phase filters but can be very slow for the case of compromise filters.

When the design specifications cannot be met after LIM iterations (see appendix), the program will stop; this situation indicates that the optimum could not be found and the resultant characteristic which may be unusable is plotted. Generally, feasible designs have been determined in less than 2000 iterations.

Minimization of the criterion function does not guarantee determination of a global minimum but rather determination of a local minimum. Depending upon the parameter vector utilized for initialization of the algorithm computation, different minima may be achieved. Experience has shown that stage-by-stage optimization, that is, utilization of the ith-stage optimum parameter vector as the initial parameter vector for the (i + 1) stage of an N-stage filter, yields lower minimum values of the criterion function than does single-pass optimization.

APPLICATIONS

Linear-Phase Filter

This example considers a digital filter having application as a phase discriminator with a linear phase characteristic and arbitrary magnitude characteristic and is shown in figure 2. In this example all magnitude weights were set to zero and all phase weights to unity, the multiplier A being arbitrarily made unity since it has no effect on the phase characteristic.

The phase requirements were $\theta_k = 1 - 2\Omega_k$ $(0 \leqq \Omega_k \leqq 1)$, and a two-stage filter was specified. When an initial parameter vector $\vec{p} = (0, 0, 0, 0.25, 0, 0, 0, 0)^T$ was used, the algorithm converged to the optimum, with $\epsilon = 10^{-4}$, in 52 iterations and a Control Data 6600 computer time of 14 seconds. The optimum parameter values computed were to four places

$A = 1.0$

$a_1 = 0$	$b_1 = -0.9871$	$c_1 = 0$	$d_1 = 0.0395$
$a_2 = 0$	$b_2 = -0.9871$	$c_2 = 0$	$d_2 = -0.0127$

It is interesting to note that the phase requirements were met to within 0.008π radian for approximately 95 percent of the frequency range.

Constant-Phase Filters

Two cases were considered to obtain filters having constant phases of $-\pi/2$ and $\pi/2$ radians over a frequency range $0.3 \leqq \Omega_k \leqq 0.7$. As in the previous case, the form of the magnitude characteristic was of no concern; hence, zero magnitude weighting was specified. With the same initial parameter state used in the previous example, the first case (lag network) optimized in 1673 iterations and 42 seconds to yield a hyperbolic magnitude characteristic and phase errors of less than 0.0003π radian throughout the specified band.

The computed parameters for the lag case were

$A = 1.0$

$a_1 = 0.5580$	$b_1 = -0.1857$	$c_1 = -0.4752$	$d_1 = 0.0363$
$a_2 = 0.5580$	$b_2 = -0.1857$	$c_2 = -0.3712$	$d_2 = -0.5686$

The positive phase filter (lead network), however, took only 165 iterations and 17 seconds to yield the desired phase characteristic with errors nowhere exceeding 0.001π radian in the specified band.

The optimum filter parameters for this second case were determined to be

$A = 1.0$

$a_1 = -0.4768$	$b_1 = -0.1548$	$c_1 = 0.5022$	$d_1 = -0.1082$
$a_2 = -0.4768$	$b_2 = -0.1548$	$c_2 = 0.4515$	$d_2 = -0.2008$

It is noted that for both cases, the phase weights outside the specified band were set to zero, and thereby allowed for arbitrary phase in these regions. Figures 3(a) and 3(b) show the resultant frequency characteristics for the lag and lead cases, respectively, of two-stage filters. The combination of the two filters, although they have antagonistic magnitude characteristics, suggests the possibility of a phase-splitting digital network.

Limited-Band Constant-Gain Linear-Phase Filter

The third example demonstrates a compromise design of a digital filter having constant-magnitude and linear-phase characteristics, over a limited frequency band, typical of phase discriminators. Here, except for $\lambda = 0$, the specifications were stated as

$$M_k = \begin{cases} 1 & (0.3 \leqq \Omega_k \leqq 0.7) \\ 0 & (\text{Elsewhere}) \end{cases}$$

$$\theta_k = \begin{cases} 1 - 2\Omega_k & (0.3 \leqq \Omega_k \leqq 0.7) \\ 0 & (\text{Elsewhere}) \end{cases}$$

Equal error and frequency weights were employed and the effects of changes in λ are shown in figure 4 for a two-stage design. Figure 4(a) shows the case of $\lambda = 0$, that is, a magnitude-only filter being specified, and coincidentally yields the linear-phase-filter characteristic derived in the first example. (See fig. 2.) Figures 4(b) and 4(c) show the magnitude and phase characteristics for the cases of $\lambda = 10$ and $\lambda = 1000$, respectively. The increasing weight on phase and resultant degradation in the magnitude characteristic are shown. The optimum parameters were

$\lambda = 0$:

$A = 0.2063$

$a_1 = 0.0000$	$b_1 = -1.0000$	$c_1 = 0.0000$	$d_1 = 0.1539$
$a_2 = 0.0000$	$b_2 = -1.0000$	$c_2 = 0.0000$	$d_2 = 0.1539$

$\lambda = 10$:

$A = 0.3658$

$a_1 = -0.9754$	$b_1 = 0.7300$	$c_1 = 0.4529$	$d_1 = 0.7211$
$a_2 = 0.8632$	$b_2 = 0.5632$	$c_2 = -0.6119$	$d_2 = 0.7443$

$\lambda = 1000$:

$A = 0.4232$

$a_1 = -1.1739$ $b_1 = 0.8489$ $c_1 = 0.7596$ $d_1 = 0.6691$

$a_2 = 1.1739$ $b_2 = 0.8489$ $c_2 = -0.7596$ $d_2 = 0.6691$

Low-Pass Zero-Phase Filter

The fourth example considers a compromise filter, having two and three stages, with specifications that are intentionally conflicting. A filter described by

$$M_k = \begin{cases} 1.0 & (0.0 \leqq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leqq 1.0) \end{cases}$$

$$\theta_k = \begin{cases} 0 & (0.0 \leqq \Omega_k \leqq 0.5) \\ \text{Unspecified} & (\text{Elsewhere}) \end{cases}$$

is specified.

Figures 5 and 6 show the results for the two- and three-stage designs, respectively, with figures 5(a) and 6(a) showing the magnitude-only ($\lambda = 0$) case. The degradation in the magnitude characteristics when greater emphasis is placed on the phase specifications is evident in figures 5(b) and 6(b) for $\lambda = 10$ and in figures 5(c) and 6(c) for $\lambda = 1000$. Comparison of figure 6 with figure 5 demonstrates the improvement brought about by increasing the number of stages. The optimum parameters for the two-stage filter were

$\lambda = 0$:

$A = 0.1196$

$a_1 = 1.0240$ $b_1 = 1.0000$ $c_1 = -0.1713$ $d_1 = 0.7676$

$a_2 = 1.0240$ $b_2 = 1.0000$ $c_2 = -0.5324$ $d_2 = 0.2286$

$\lambda = 10$:

$A = 0.4879$

$a_1 = 0.2018$	$b_1 = 0.6684$	$c_1 = 0.3560$	$d_1 = 0.4612$
$a_2 = 0.6597$	$b_2 = 0.4335$	$c_2 = 0.0806$	$d_2 = 0.7671$

$\lambda = 1000$:

$A = 0.5343$

$a_1 = 0.0205$	$b_1 = 0.7169$	$c_1 = -0.0836$	$d_1 = 0.6255$
$a_2 = 0.6286$	$b_2 = 0.7905$	$c_2 = 0.2123$	$d_2 = 0.6681$

The optimum parameters for the three-stage filter were

$\lambda = 0$:

$A = 0.0510$

$a_1 = 0.8537$	$b_1 = 1.0000$	$c_1 = -0.1068$	$d_1 = 1.0000$
$a_2 = 0.8537$	$b_2 = 1.0000$	$c_2 = -0.4046$	$d_2 = 0.5990$
$a_3 = 0.8537$	$b_3 = 1.0000$	$c_3 = -0.6799$	$d_3 = 0.2069$

$\lambda = 10$:

$A = 0.5109$

$a_1 = 1.3302$	$b_1 = 0.5515$	$c_1 = -0.1731$	$d_1 = 0.8097$
$a_2 = 0.6844$	$b_2 = 0.7157$	$c_2 = 1.1880$	$d_2 = 0.5850$
$a_3 = -0.0373$	$b_3 = 0.7012$	$c_3 = 0.3825$	$d_3 = 0.5262$

$\lambda = 1000$:

$A = 0.4515$			
$a_1 = 1.5107$	$b_1 = 0.5286$	$c_1 = -0.1771$	$d_1 = 0.8972$
$a_2 = 0.5825$	$b_2 = 0.7490$	$c_2 = 1.3094$	$d_2 = 0.4191$
$a_3 = -0.1663$	$b_3 = 0.7485$	$c_3 = 0.2002$	$d_3 = 0.6393$

A three-stage design of this example is used to demonstrate the existence of two distinct local minima, dependent upon the initial parameter vector. In the first case, a single-pass optimization was accomplished with $\vec{p} = (0, 0, 0, 0.25, 0, 0, 0, 0)^T$ for the initial parameter vector and resulted in the optimum filter shown in figure 6(a). In the second case, a stage-by-stage optimization was accomplished by utilizing the optimum parameter vector from a two-stage design for the initial parameter vector of a three-stage design and resulted in the optimum filter shown in figure 7. Comparison of these results demonstrates the existence of two distinct local minima, the stage-by-stage minimization yielding superior results.

CONCLUDING REMARKS

A method has been developed for a computer-aided design of cascade canonical digital filters having prescribed magnitude or phase characteristics or a compromise between the two. The method, which uses an unconstrained minimization algorithm, allows for arbitrary error and frequency weighting. Representative designs of phase and compromise filters have demonstrated the utility of the technique. Although convergence is generally fast for magnitude phase filters, it may be slow for the case of compromise filters.

Langley Research Center,
National Aeronautics and Space Administration,
Hampton, Va., February 17, 1972.

APPENDIX

PROGRAM LISTING

This appendix contains a program listing written for the Control Data 6600 computer at the Langley Research Center, Hampton, Virginia, and is an adaptation of that written by Kenneth Steiglitz at Princeton University for the design of specified magnitude-only filters.

```
              PROGRAM STG73(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,PUNCH)
000003        EXTERNAL FUNCT
000003        DIMENSION H(184),X(16),G(16)
000003        COMMON/RAW/W(100),Y(100),M,PHASED(100),ALAMDA,FR,WTM(100),
             C WTP(100),KTYP
000003        COMMON/RAW1/ICALL,KCALL,LIM
000003        CALL CALCOMP
000004        CALL LEROY
000005        WRITE(6,51)
000011     51 FORMAT(* INPUT DATA*)
000011        M=0
000012     30 M=M+1
000014        READ(5,21)W(M),Y(M),PHASED(M),WTM(M),WTP(M)
000031     21 FORMAT(5F10.5)
000031        WRITE(6,22)M,W(M),Y(M),PHASED(M),WTM(M),WTP(M)
000051     22 FORMAT(* I=*,I3,* W=* F 7.4,* Y=* F 7.4,* PHASED=*,F7.4,
             C * WTM=*,F7.4,* WTP=*,F7.4)
000051        IF(W(M).LT.1.00)GOTO30
000054        DO 15 J=1,16
000056     15 X(J)=0.00
000061        X(4)=.25
000062     99 READ(5,60)L,LIM,EST,EPS,HMAX,ALAMDA,FR,KTYP
000106   60   FORMAT(2I5,5F10.5,I10)
000106        IF(FR.LT..001) FR=1.
000112        N=4*L
000113        IF(EOF,5) 999,888
000117    888 CONTINUE
000117        WRITE(6,61)L,LIM,EST,EPS,HMAX,FR,ALAMDA
000141     61 FORMAT(* L=*,I3,* LIM=*,I5,* EST=*,F10.5,* EPS=*,F10.5,
             C* HMAX=*,F10.5,* FREQRANGE=*,F10.5,* LAMBDA=*,F10.5)
000141        ICALL=0
000142     98 KCALL=0
000143        CALL FLPWL(FUNCT,N,X,F,G,EST,EPS,EC,IER,H)
000155        CALL ROOTS(N,X)
000157        CALL INSIDE(N,X,KFLAG)
000162        WRITE(6,26)IER,KFLAG,ICALL,KCALL
000176     26 FORMAT(* IER=*,I5,* KFLAG=*,I5,* ICALL=*,I5,* KCALL=*,I5)
000176        IF(KCALL.GT.300) GO TO 98
000202        IF((KFLAG.NE.0.OR.IER.NE.0).AND.ICALL.LE.LIM) GO TO 98
000213        CALL ROOTS(N,X)
000215        ICALL=-10
000216        CALL FUNCT(N,X,F,G,HMAX)
000222        GOTO99
000223   999  CALL CALPLT(0.,0.,999)
000226        STOP
000230        END
```

```
              SUBROUTINE FUNCT(N,X,F,G,HMAX)
000010        DIMENSION OMEGA(200),PHASEX(200),AMAG(200),PHE(100),PHA(100)
000010        DIMENSION XPL(200),YPL(200),CX(200),CY(200)
000010        DIMENSION H(136),X(16),G(16),YHT(100),E(100)
000010        COMPLEX ZQBAR,ZQ,ZZCUR,ZZCUR2
000010        COMPLEX    Z(100),TUM(100,4),DEN(100,4),Q,QBAR,ZCUR,ZCUR2,ONEC
000010        COMMON/RAW/W(100),Y(100),M,PHASED(100),ALAMDA,FR,WTM(100),
             C  WTP(100),KTYP
000010        COMMON/RAW1/ICALL,KCALL,LIM
000010        ONEC=CMPLX(1.00,0.00)
000012        PI=-3.14159265358979
000014        K=N/4
000015        IF(ICALL.NE.0)GOTO101
000015        DO 102 I=1,M
000020    102 Z(I)= CEXP( CMPLX(0.00,W(I)*PI))
000040    101 A1=0.00
000041        A2=0.00
000042        DO 40 I=1,M
000043        ZCUR=Z(I)
000046        ZCUR2=ZCUR*ZCUR
000053        Q= CMPLX(1.00,0.00)
000056        DO 33 J=1,K
000057        J4=(J-1)*4
000061        TUM(I,J)=1.00+X(J4+1)*ZCUR+X(J4+2)*ZCUR2
000103        DEN(I,J)=1.00+X(J4+3)*ZCUR+X(J4+4)*ZCUR2
000126     33 Q=Q*TUM(I,J)/DEN(I,J)
000147        QBAR= CONJG(Q)
000152        YHT(I)=Q*QBAR
000160        A2=A2+YHT(I)*WTM(I)
000163        YHT(I)= SQRT(YHT(I))
000167     40 A1=A1+YHT(I)*Y(I)*WTM(I)
000201        IF(KTYP.NE.0) GO TO 666
000202        A=A1/A2
000204        GO TO 667
000205    666  A=1.
000207    667 CONTINUE
000207        DO 612 I=1,M
000211        ZZCUR=Z(I)
000214        ZZCUR2=ZZCUR*ZZCUR
000221        ZQ=CMPLX(1.,0.)
000224        PHAS=0.
000225        DO 611 J=1,K
000226        J4Z=(J-1)*4
000230        ZQBAR=ONEO+X(J4Z+1)*ZZCUR+X(J4Z+2)*ZZCUR2
000247        ZA1=ZQBAR
000251        ZA2=CMPLX(0.,-1.)*ZQBAR
000261        PHAS=PHAS+ATAN2(ZA2,ZA1)
000265        ZQ=ZQ*ZQBAR
000273        ZQBAR=ONEO+X(J4Z+3)*ZZCUR+X(J4Z+4)*ZZCUR2
000315        ZA1=ZQBAR
000317        ZA2=CMPLX(0.,-1.)*ZQBAR
000327        PHAS=PHAS-ATAN2(ZA2,ZA1)
000332        ZQ=ZQ/ZQBAR
000342    611 CONTINUE
000350        PHA(I)=-PHAS/PI
000353    612 PHE(I)=PHA (I)-PHASED(I)
000360        DO 57 J=1,16
000361     57 G(J)=0.00
000364        V1=0.
000365        V2=0.
000365        DO 42 I=1,M
000367        ZCUR=Z(I)
000372        ZCUR2=ZCUR*ZCUR
000377        YHT(I)=A*YHT(I)
000402        E(I)=YHT(I)-Y(I)
000405        V1=V1+E(I)*E(I)*WTM(I)
000410        V2=V2 +PHE(I)*PHE(I)*WTP(I)
000414        P=E(I)*WTM(I)
000416        T=PHE(I)*WTP(I)*2.*ALAMDA
```

```
000422        DO 422 J=1,K
000423        J4=(J-1)*4
000425        Q=2.*R*YHT(I)/TUM(I,J)
000441        G(J4+1)=G(J4+1)+Q*ZCUR
000451        G(J4+2)=G(J4+2)+Q*ZCUR2
000461        Q=-2.*R*YHT(I)/DEN(I,J)
000475        G(J4+3)=G(J4+3)+Q*ZCUR
000505        G(J4+4)=G(J4+4)+Q*ZCUR2
000514    422 CONTINUE
000516        DO 42  J=1,K
000520        J4=(J-1)*4
000522        G(J4+1)=G(J4+1)+T*AIMAG(ZCUR/TUM(I,J))
000537        G(J4+2)=G(J4+2)+T*AIMAG(ZCUR2/TUM(I,J))
000554        G(J4+3)=G(J4+3)+T*AIMAG(-ZCUR/DEN(I,J))
000571        G(J4+4)=G(J4+4)+T*AIMAG(-ZCUR2/DEN(I,J))
000604     42 CONTINUE
000613        F=V1+ALAMDA*V2
000616        ICALL=ICALL+1
000620        KCALL=KCALL+1
000621        IF(KCALL.GT.300)RETURN
000624        IF((ICALL/10)*10.EQ.ICALL-1)WRITE(6,25)ICALL,F,(G(J),J=1,N)
000657     25 FORMAT(* CALL NO.*,I4,* F=*,F15.8/( 35X, 4F15.8))
000657        IF(ICALL.GT.2000) GO TO 450
000663        IF(ICALL.GT.0)RETURN
000665        GO TO 449
000666    450 IF(ICALL.GT.LIM+1) GO TO 449
000673        RETURN
      C......PRINT OUT
000673    449 WRITE(6,50)F
000701     50 FORMAT(* FINAL FUNCTION VALUE =*,F15.8)
000701        WRITE(6,51)A
000707     51 FORMAT(* A=*,F15.8)
000707        WRITE(6,52)(X(J),J=1,N)
000731     52 FORMAT(* FINAL X =*/(* *,4F15.8))
000731        WRITE(6,54)(G(J),J=1,N)
000753     54 FORMAT(* FINAL GRADIENT =*/(* *,4F15.8))
000753        DO 55 I=1,M
000760     55 WRITE(6,56)I,W(I),Y(I),YHT(I),E(I),PHASED(I),PHA(I),PHE(I)
001011     56 FORMAT(* I=*,I3,* W=*,F8.5,* Y=*,F8.5,* YHT=*,F8.5,* E=*,F8.5,
             C * PHASED=*,F8.5,* PHA=*,F8.5,* PHE=*,F8.5)
001011        WRITE(6,59)K
001016     59 FORMAT(* FINAL TABLE FOR A*,I2,* STAGE FILTER*)
001016        YMIN33=0.
001017        YMAX33=0.
001020        YMAX11=0.
001021        S=FR/200.
001023        DO 60 I=1,201
001027        FREQ=S*FLOAT(I-1)
001032        ZCUR= CEXP( CMPLX(0.00,FREQ*PI))
001040        ZCUR2=ZCUR*ZCUR
001045        Q= CMPLX(1.00,0.00)
001050        PHASE=0.00
001051        DO 61 J=1,K
001055        J4=(J-1)*4
001057        QBAR=ONEO+X(J4+1)*ZCUR+X(J4+2)*ZCUR2
001076        A1=QBAR
001100        A2=CMPLX(0.00,-1.00)*QBAR
001110        PHASE=PHASE+ ATAN2(A2,A1)
001114        Q=Q*QBAR
001122        QBAR=ONEO+X(J4+3)*ZCUR+X(J4+4)*ZCUR2
001144        A1=QBAR
001146        A2=CMPLX(0.00,-1.00)*QBAR
001156        PHASE=PHASE- ATAN2(A2,A1)
001162     61 Q=Q/QBAR
001177        A1=Q* CONJG(Q)
001207        A1=A* SQRT(A1)
001212        PHASE=-PHASE/PI
001214        OMEGA(I)=FREQ
001216        AMAG(I)=A1
001217        PHASEX(I)=PHASE
```

```
001221          IF(PHASE-YMAX33) 301,301,401
001226     401  YMAX33=PHASE
001230     301  CONTINUE
001230          IF(PHASE-YMIN33) 402,302,302
001233     402  YMIN33=PHASE
001235     302  CONTINUE
001235          IF(A1-YMAX11) 300,300,400
001240     400  YMAX11=A1
001242     300  CONTINUE
001242      60  WRITE(6,62)FREQ,PHASE,A1
001261      62  FORMAT(* W=*,F15.8,* PHASE/PI=*,F15.8,* YHT=*,F15.8)
          C SCALE COMPUTATIONS
001261          IYMAX11=YMAX11+.9999
001264          YMAX1=FLOAT(IYMAX11)
001266          IF(HMAX.EQ.0.) GO TO 333
001267          YMAX1=HMAX
001270     333  CONTINUE
001270          IYMAX33=YMAX33+.999
001273          YMAX3=FLOAT(IYMAX33)
001275          IYMIN33=YMIN33-.9999
001300          YMIN3=FLOAT(IYMIN33)
001301          AXM=10H   F/(FS/2)
001303          AYM=10HMAGNITUDES
001304          AFR=200.*FR
001306          NFR=200
          C MAGNITUDE (COMPUTED) - FREQUENCY PLOT
001307          CALL INFOPLT(0,NFR,OMEGA,1,AMAG,1,0.,FR,0.,YMAX1,0.5,-10,AXM,-10,
               C AYM,0)
          C MAGNITUDE (DESIRED) - FREQUENCY PLOT
001327          CALL INFOPLT(1,M,W,1,Y,1, 0.,FR,0.,YMAX1,0.5,-10,AXM,-10,AYM,11)
001347          AYM=10H PHASE/PI
          C PHASE-FREQUENCY PLOT
001351          CALL INFOPLT(0,NFR,OMEGA,1,PHASEX,1,0.,FR,YMIN3,YMAX3,0.5,-10,
               C AXM,-10,AYM,0)
          C PHASE (DESIRED) - FREQUENCY PLOT
001370          CALL INFOPLT(1, M , W   ,1,PHASED,1,0.,FR,YMIN3,YMAX3,0.5,-10,
               C AXM,-10,AYM,11)
001410          RETURN
001411          END



                SUBROUTINE FLPWL(FUNCT,N,X,F,G,EST,EPS,LIMIT,IER,H)
000015          DIMENSION H(1),X(1),G(1)
000015          COMMON/RAW1/ICALL,KCALL,LIM
000015          CALL FUNCT(N,X,F,G,HMAX)
000027          IF(KCALL.GT.300) GO TO 723
000032          IF(ICALL.GT.LIM) GO TO 723
000035          GO TO 917
000035     723  IER=3
000037          RETURN
000037     917  IER=0
000040          KOUNT=0
000041          N2=N+N
000042          N3=N2+N
000043          N31=N3+1
000045       1  K=N31
000047          DO 4 J=1,N
000051          H(K)=1.00
000053          NJ=N-J
000054          IF(NJ) 5,5,2
000055       2  DO 3 L=1,NJ
000061          KL=K+L
```

```
000061    3   H(KL)=0.00                                  0246
000066    4   K=KL+1                                      0247
000072    5   KOUNT=KOUNT+1                               0248
000074        WRITE(6,501)KOUNT                           0249
000103  501   FORMAT(* KOUNT=*,I5)                        0250
000101        OLDF=F                                      0251
000106        DO 9 J=1,N                                  0252
000107        K=N+J                                       0253
000110        H(K)=G(J)                                   0254
000114        K=K+N                                       0255
000115        H(K)=X(J)                                   0256
000121        K=J+N3                                      0257
000122        T=0.00                                      0258
000123        DO 8 L=1,N                                  0259
000124        T=T-G(L)*H(K)                               0260
000131        IF(L-J) 6,7,7                               0261
000134    6   K=K+N-L                                     0262
000137        GO TO 8                                     0263
000137    7   K=K+1                                       0264
000141    8   CONTINUE                                    0265
000144    9   H(J)=T                                      0266
000150        DY=0.00                                     0267
000150        HNRM=0.00                                   0268
000151        GNRM=0.00                                   0269
000153        DO 10 J=1,N                                 0270
000154        HNRM=HNRM+ ABS(H(J))                        0271
000160        GNRM=GNRM+ ABS(G(J))                        0272
000163   10   DY=DY+H(J)*G(J)                             0273
000173        IF(DY) 11,51,51                             0274
000174   11   IF(HNRM/GNRM-EPS) 51,51,12                  0275
000200   12   FY=F                                        0276
000201        ALFA=2.00*(EST-F)/DY                        0277
000204        AMBDA=1.00                                  0278
000205        IF(ALFA) 15,15,13                           0279
000207   13   IF(ALFA-AMBDA) 14,15,15                     0280
000212   14   AMBDA=ALFA                                  0281
000214   15   ALFA=0.00                                   0282
000215   16   FX=FY                                       0283
000216        DX=DY                                       0284
000220        DO 17 I=1,N                                 0285
000222   17   X(I)=X(I)+AMBDA*H(I)                        0286
000231        CALL FUNCT(N,X,F,G,HMAX)                    0289
000243        IF(KCALL.GT.300) GO TO 724
000246        IF(ICALL.GT.LIM) GO TO 724
000251        GO TO 918
000251  724   IER=3
000253        RETURN
000253  918   FY=F
000254        DY=0.00                                     0292
000255        DO 18 I=1,N                                 0293
000257   18   DY=DY+G(I)*H(I)                             0294
000266        IF(DY) 19,36,22                             0297
000267   19   IF(FY-FX) 20,22,22                          0298
000272   20   AMBDA=AMBDA+ALFA                            0299
000274        ALFA=AMBDA                                  0300
000275        ERROR=1.E10                                 0301
000276        IF(HNRM*AMBDA-ERROR)16,16,21                0302
000302   21   IER=2                                       0303
000304        RETURN                                      0304
000304   22   T=0.00                                      0305
000305   23   IF(AMBDA) 24,36,24                          0306
000306   24   Z=3.00*(FX-FY)/AMBDA+DX+DY                  0307
000314        ALFA=AMAX1( ABS(Z), ABS(DX), ABS(DY))       0308
000326        DALFA=Z/ALFA                                0309
```

```
000327          DALFA=DALFA*DALFA-DX/ALFA*DY/ALFA                    0310
000333          IF(DALFA) 51,25,25                                   0311
000335    25    W=ALFA* SQRT(DALFA)                                  0312
000340          ALFA= (DY+W-Z)*AMBDA/(DY+2.00*W-DX)                  0313
000351          DO 26  I=1,N                                         0314
000356    26    X(I)=X(I)+(T-ALFA)*H(I)                              0315
000366          CALL FUNCT(N,X,F,G,HMAX)                             0320
000400          IF(KCALL.GT.300) GO TO 725
000403          IF(ICALL.GT.LIM) GO TO 725
000406          GO TO 919
000406     725  IER=3
000410          RETURN
000410     919  IF(F-FX) 27,27,28
000413    27    IF(F-FY) 36,36,28                                    0323
000416    28    DALFA=0.00                                           0324
000417          DO 29  I=1,N                                         0325
000421    29    DALFA=DALFA+G(I)*H(I)                                0326
000430          IF(DALFA) 30,33,33                                   0329
000431    30    IF(F-FX) 32,31,33                                    0330
000434    31    IF(DX-DALFA) 32,36,32                                0331
000436    32    FX=F                                                 0332
000437          DX=DALFA                                             0333
000440          T=ALFA                                               0334
000442          AMBDA=ALFA                                           0335
000443          GO TO 23                                             0336
000443    33    IF(FY-F) 35,34,35                                    0337
000445    34    IF(DY-DALFA) 35,36,35                                0338
000447    35    FY=F                                                 0339
000450          DY=DALFA                                             0340
000451          AMBDA=AMBDA-ALFA                                     0341
000454          GO TO 22                                             0342
000454    36    DO 37  J=1,N                                         0343
000456          K=N+J                                                0344
000457          H(K)=G(J)-H(K)                                       0345
000463          K=K+N                                                0346
000464    37    H(K)=X(J)-H(K)                                       0347
000472          IF(OLDF-F+EPS) 51,38,38                              0348
000476    38    IER=0                                                0349
000477          IF(KOUNT-N) 42,39,39                                 0350
000501    39    T=0.00                                               0351
000502          Z=0.00                                               0352
000503          DO 40  J=1,N                                         0353
000504          K=N+J                                                0354
000505          W=H(K)                                               0355
000510          K=K+N                                                0356
000511          T=T+ ABS(H(K))                                       0357
000514    40    Z=Z+W*H(K)                                           0358
000523          IF(HNRM-EPS) 41,41,42                                0359
000526    41    IF(T-EPS) 56,56,42                                   0360
000531    42    IF(KOUNT-LIMIT) 43,50,50                             0361
000534    43    ALFA=0.00                                            0362
000535          DO 47  J=1,N                                         0363
000537          K=J+N3                                               0364
000540          W=0.00                                               0365
000542          DO 46  L=1,N                                         0366
000543          KL=N+L                                               0367
000544          W=W+H(KL)*H(K)                                       0368
000552          IF(L-J) 44,45,45                                     0369
000554    44    K=K+N-L                                              0370
000557          GO TO 46                                             0371
000557    45    K=K+1                                                0372
000561    46    CONTINUE                                             0373
000564          K=N+J                                                0374
000565          ALFA=ALFA+W*H(K)                                     0375
```

```
000572      47    H(J)=W
000576            IF(Z*ALFA) 48,1,48
000600      48    K=N31
000602            DO 49 L=1,N
000603            KL=N2+L
000605            DO 49 J=L,N
000606            NJ=N2+J
000607            H(K)=H(K)+H(KL)*H(NJ)/Z-H(L)*H(J)/ALFA
000625      49    K=K+1
000633            GO TO 5
000634      50    IFR=1
000636            RETURN
000636      51    DO 52 J=1,N
000640            K=N2+J
000641      52    X(J)=H(K)
000647            CALL FUNCT(N,X,F,G,HMAX)
000661            IF(KCALL.GT.300) GO TO 726
000664            IF(ICALL.GT.LIM) GO TO 726
000667            GO TO 920
000667       726  IFR=3
000671            RETURN
000671       920  IF(GNRM-EPS) 55,55,53
000674      53    IF(IFR) 56,54,54
000676      54    IFR=-1
000700            GO TO 1
000700      55    IFR=0
000701      56    RETURN
000702            END
```

```
                  SUBROUTINE INSIDE(N,X,KFLAG)
000006            DIMENSION X(16)
000006            J=-1
000006            KFLAG=0
000007        10  J=J+2
000011            IF(J.GT.N)RETURN
000014            B=-.500*X(J)
000016            C=X(J+1)
000020            DISC=B*B-C
000022            IF(DISC.LE.0.00)GOTO20
            C......REAL ROOTS
000024            DISC= SQRT(DISC)
000025            R1=B+DISC
000027            R2=B-DISC
000031            DR1= ABS(R1)
000033            DR2= ABS(R2)
000035            IF(DR1.LE.1.00.AND.DR2.LE.1.00)GOTO10
000051            KFLAG=1
000051            IF(DR1.GT.1.00)R1=1.00/R1
000054            IF(DR2.GT.1.00)R2=1.00/R2
000060            X(J)=-1.00*(R1+R2)
000064            X(J+1)=R1*R2
000066            GOTO10
            C......COMPLEX ROOTS
000066        20  IF(C.LE.1.00)GOTO10
000071            KFLAG=1
000071            C=1.00/C
000072            X(J+1)=C
000074            X(J)=X(J)*C
000076            GOTO10
000077            END
```

```
              SUBROUTINE ROOTS(N,X)
000005        DIMENSICN X(16)
000005        WRITE(6,40)
000010     40 FORMAT(* ROOTS*/6X,*REAL*,11X,*IMAG*,11X,*REAL*,11X,*IMAG*)
000010        J=-1
000011     10 J=J+2
000013        IF(J.GT.N)RETURN
000017        B=-.500*X(J)
000021        C=X(J+1)
000023        DISC=B*B-C
000025        IF(DISC.LE.0.00)GOTO20
       C......REAL RCOTS
000027        DISC= SQRT(DISC)
000030        R1=B+DISC
000032        R2=B-DISC
000035        WRITE(6,30)R1,R2
000044     30 FORMAT(* *,F15.8,15X,F15.8)
000044        GOTO10
       C......COMPLEX ROOTS
000046     20 DISC= SQRT(-1.00*DISC)
000052        DISCM=-1.00*DISC
000055        WRITE(6,50)B,DISC,B,DISCM
000070     50 FORMAT(* *,4F15.8)
000070        GOTO10
000072        END
```

REFERENCES

1. Gold, Bernard; and Rader, Charles M.: Digital Processing of Signals. McGraw-Hill Book Co., Inc., c.1969.

2. Golden, Roger M.: Digital Filter Synthesis by Sampled-Data Transformation. IEEE Trans. Audio & Electroacoust., vol. AU-16, no. 3, Sept. 1968, pp. 321-329.

3. Kaiser, J. F.: Digital Filters. System Analysis by Digital Computer, Franklin F. Kuo and James F. Kaiser, eds., John Wiley & Sons, Inc., c.1966, pp. 218-285.

4. Rader, Charles M.; and Gold, Bernard: Digital Filter Design Techniques in the Frequency Domain. Proc. IEEE, vol. 55, no. 2, Feb. 1967, pp. 149-171.

5. Ferguson, Michael J.; and Mantey, Patrick E.: Automatic Frequency Control Via Digital Filtering. IEEE Trans. Audio & Electroacoust., vol. AU-16, no. 3, Sept. 1968, pp. 392-397.

6. Steiglitz, Kenneth: Computer-Aided Design of Recursive Digital Filters. IEEE Trans. Audio & Electroacoust., vol. AU-18, no. 2, June 1970, pp. 123-129.

7. Jackson, Leland B.; Kaiser, James F.; and McDonald, Henry S.: An Approach to the Implementation of Digital Filters. IEEE Trans. Audio & Electroacoust., vol. AU-16, no. 3, Sept. 1968, pp. 413-421.

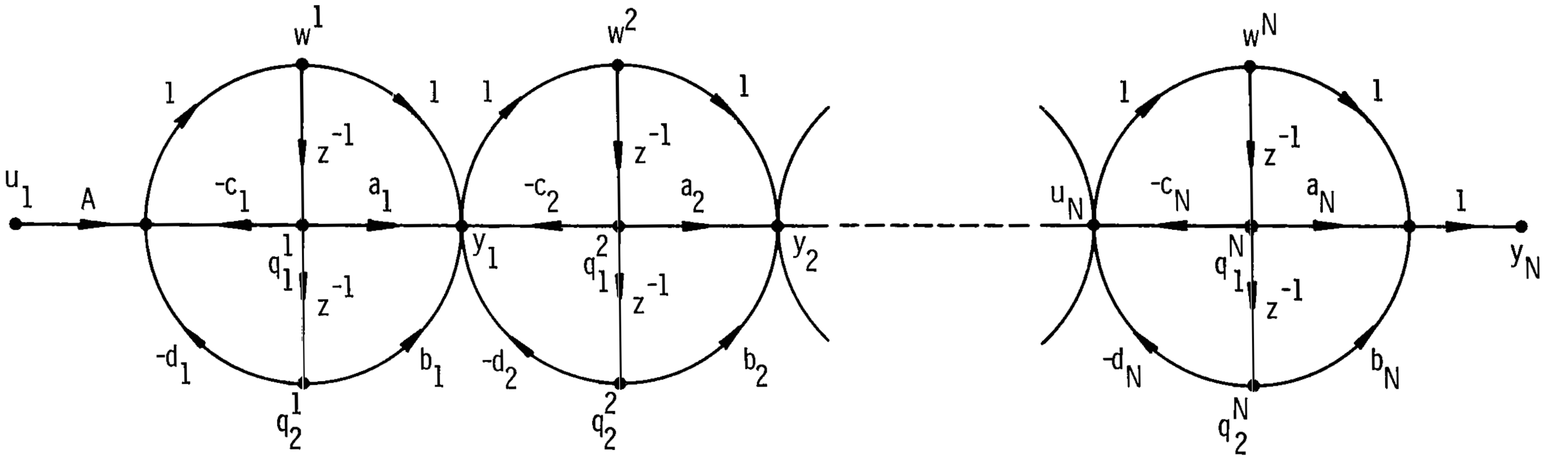


Figure 1.- Signal flow graph of cascaded digital filter.

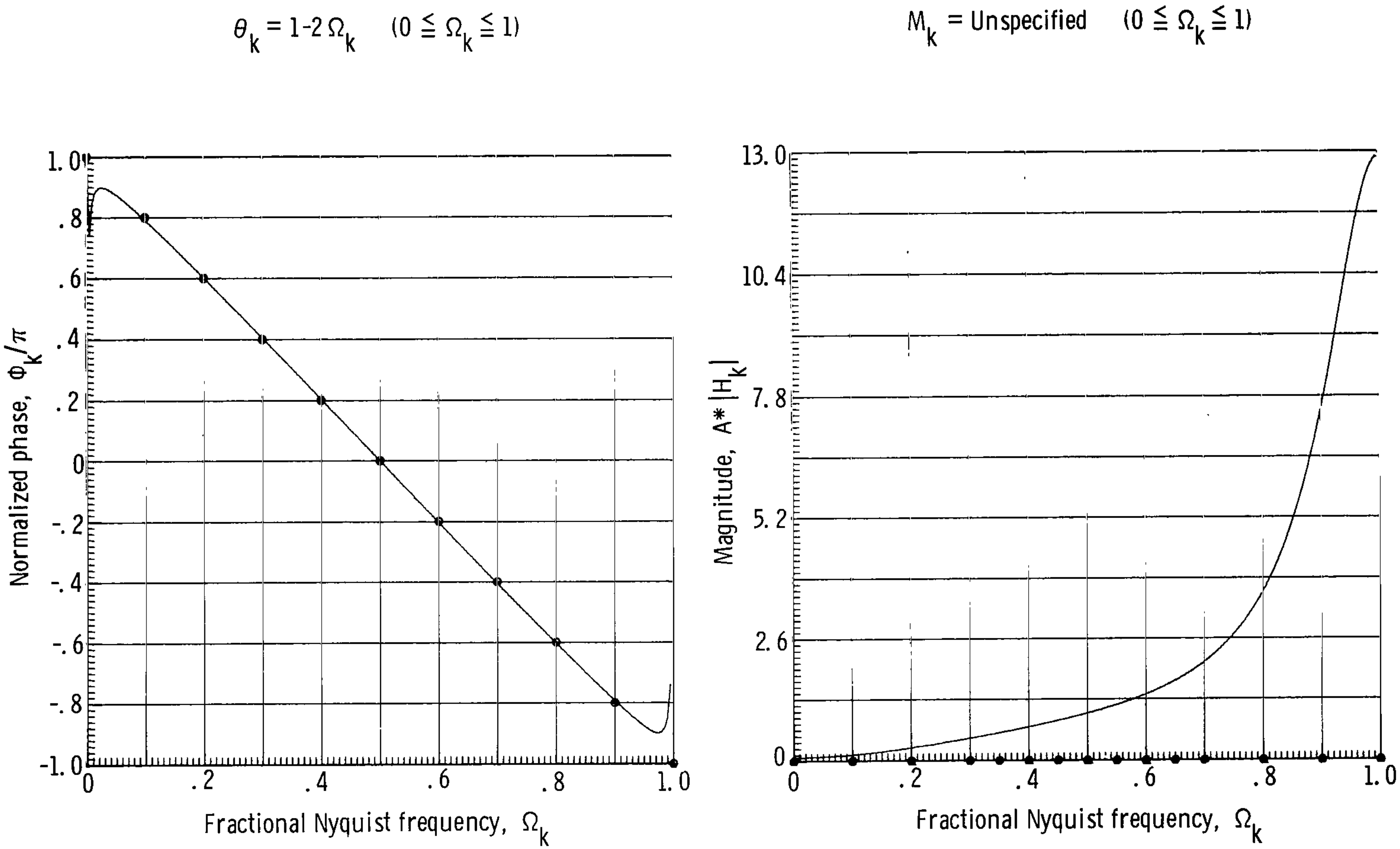
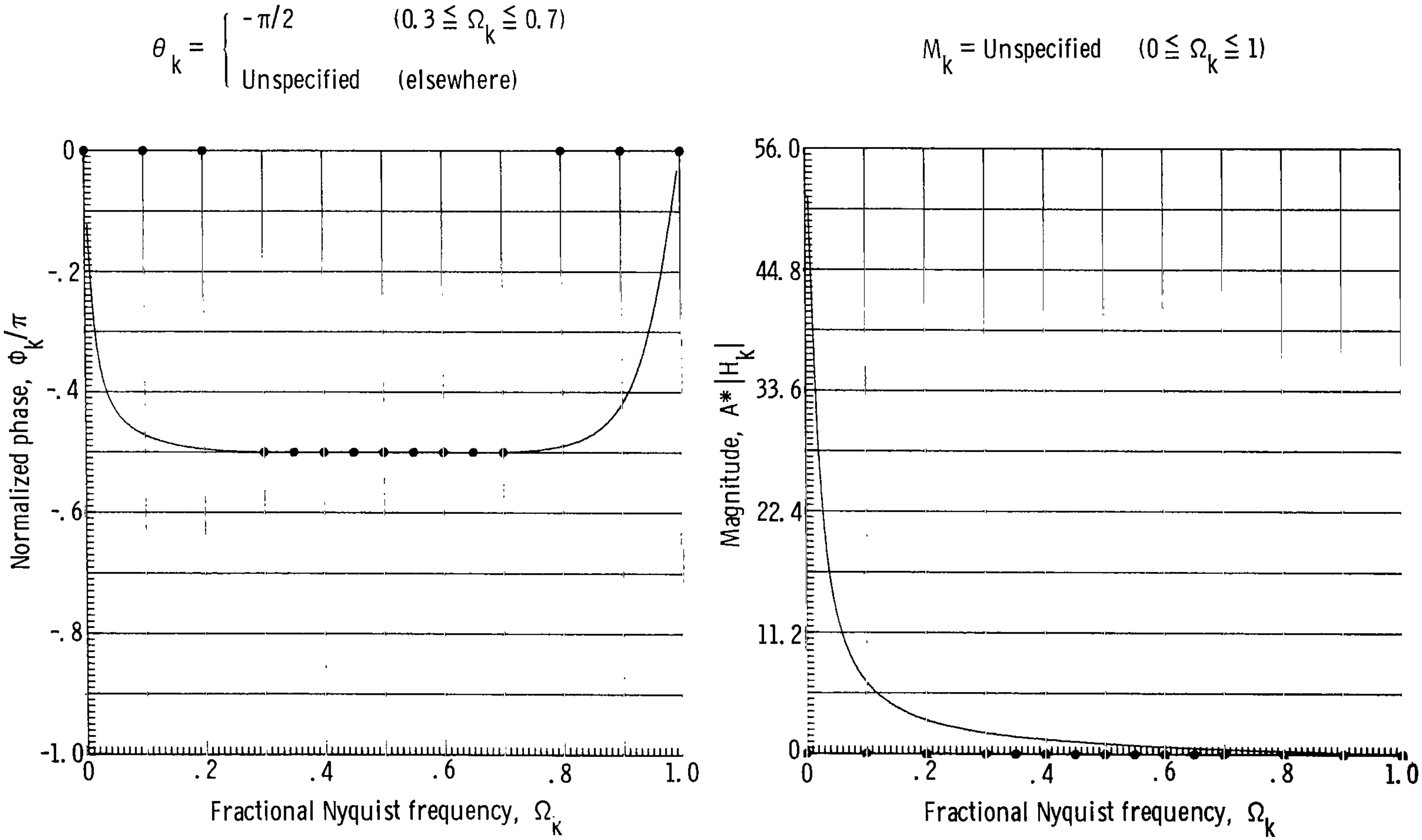
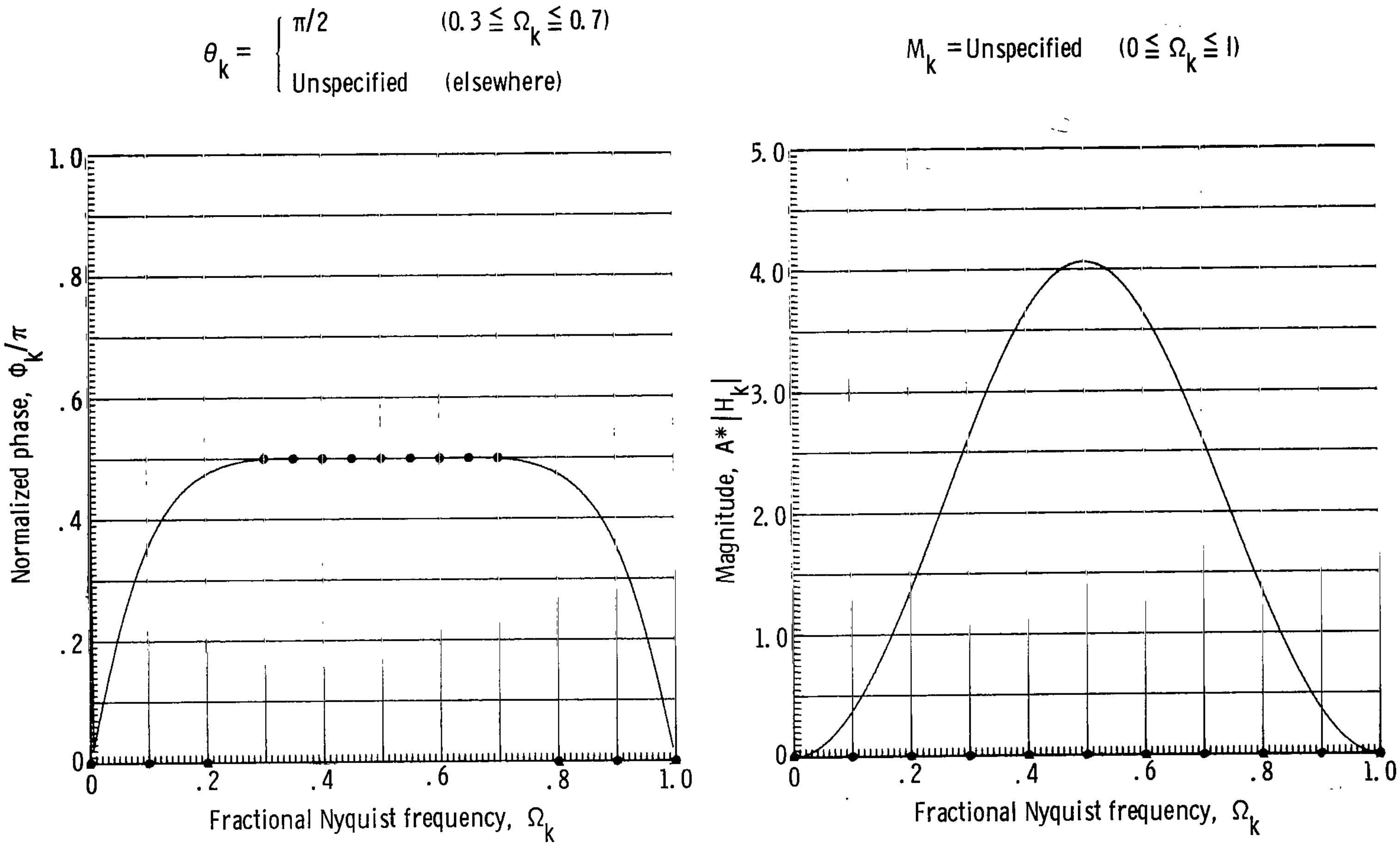


Figure 2.- Two-stage linear-phase filter.



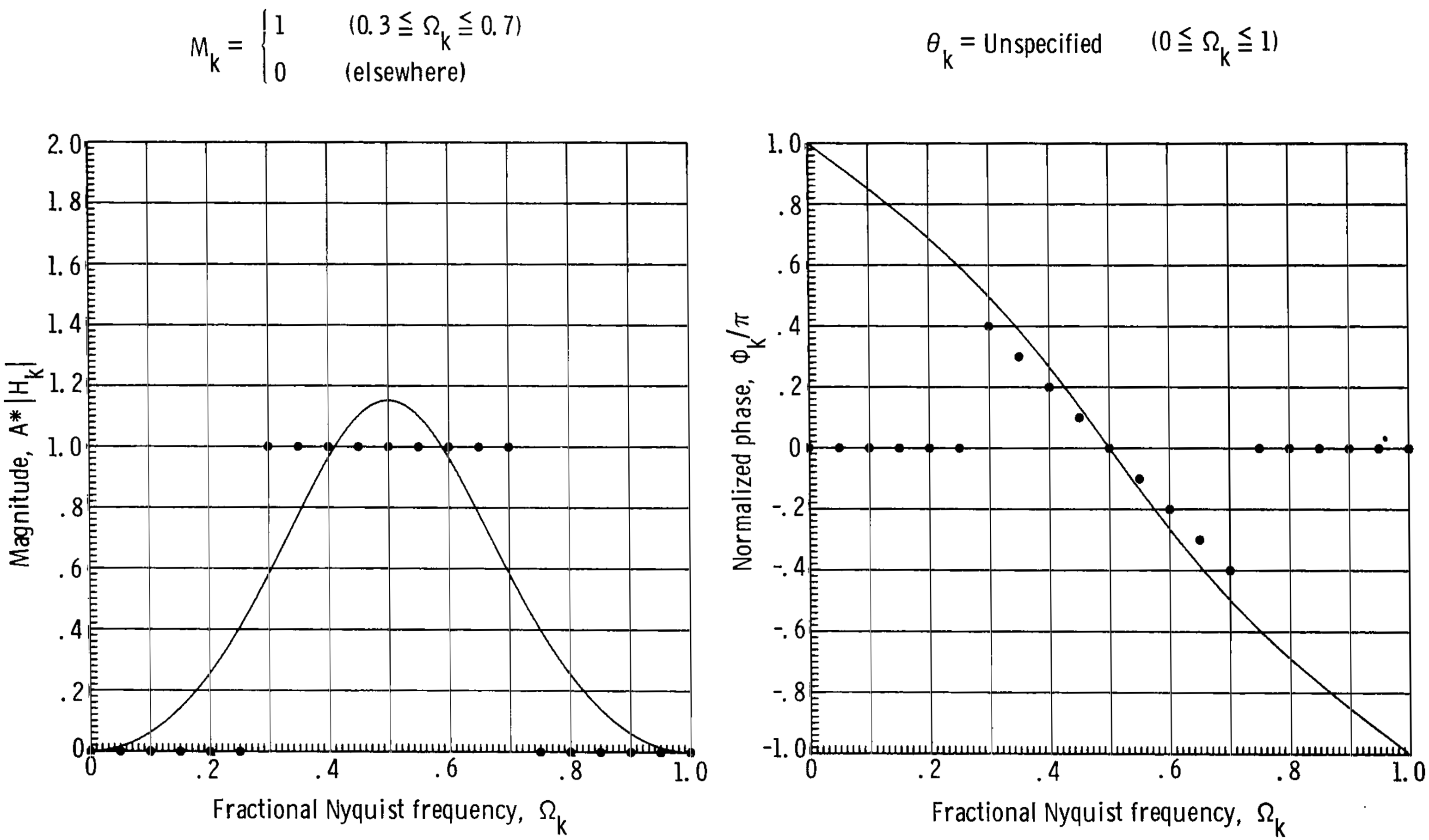
(a) Lag filter.

Figure 3.- Two-stage constant-phase filters.



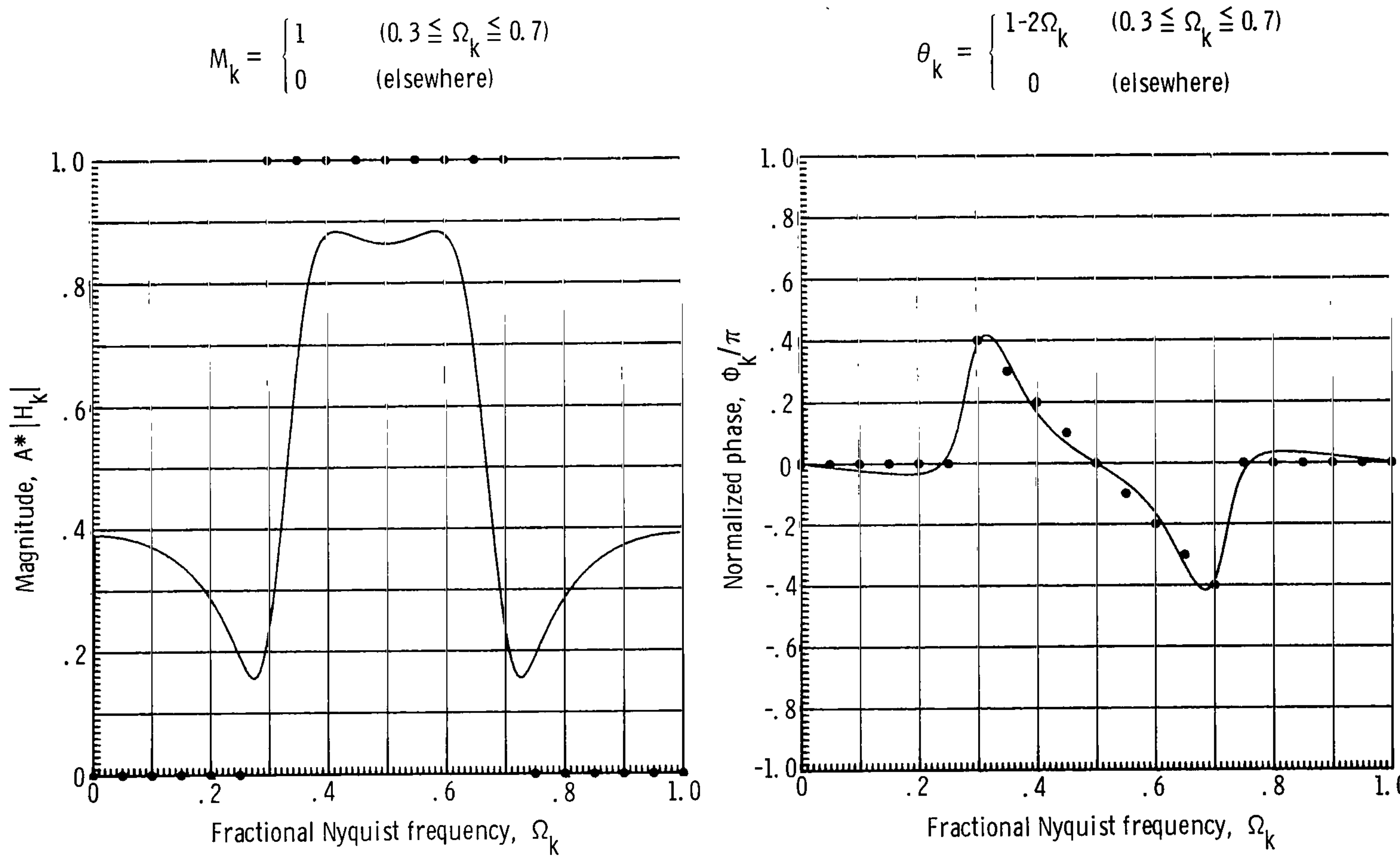
(b) Lead filter.

Figure 3.- Concluded.



(a) Unspecified phase filter. $\lambda = 0$.

Figure 4.- Two-stage limited-band constant-gain filters.

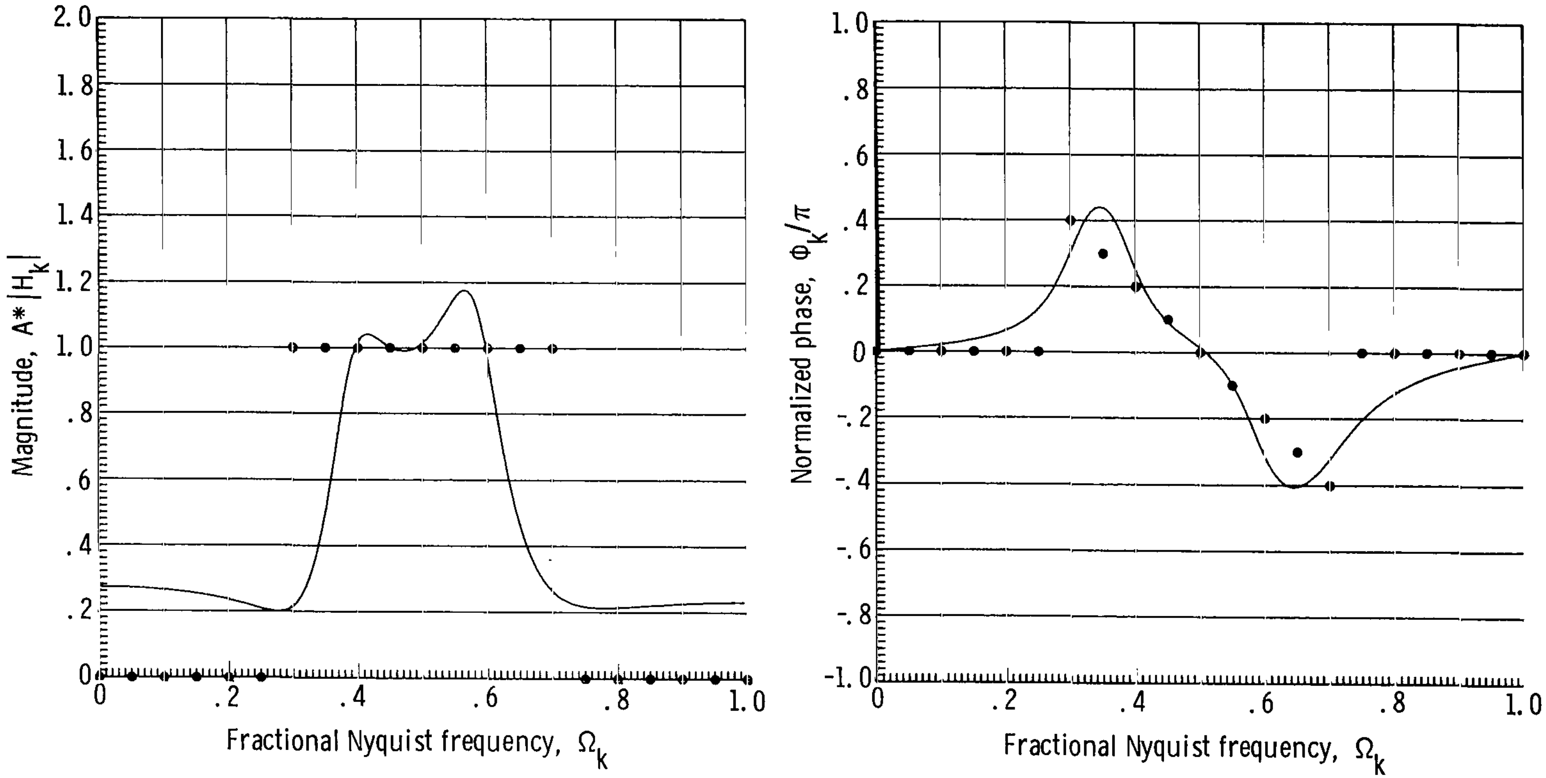


(b) Linear-phase filter. $\lambda = 10$.

Figure 4.- Continued.

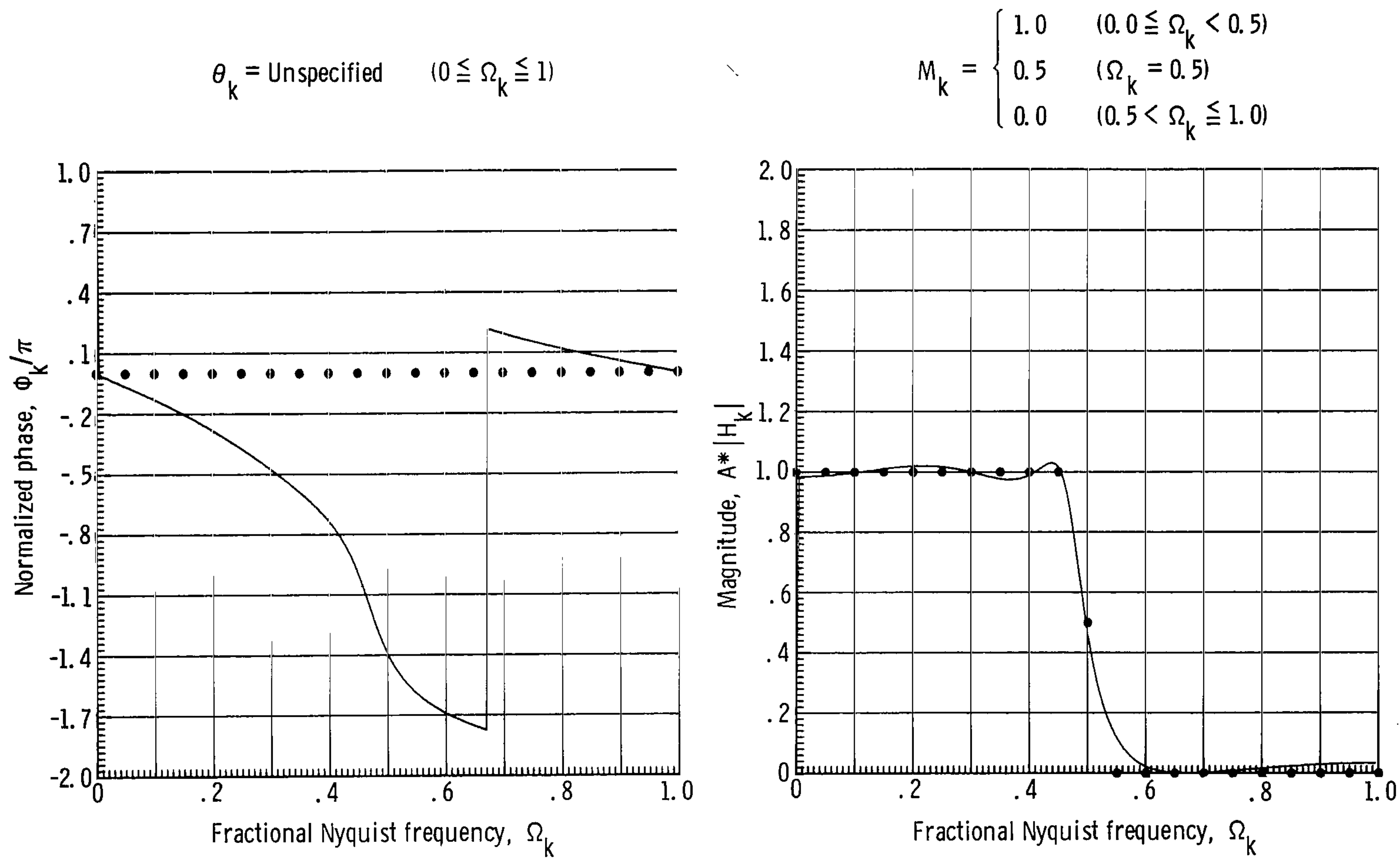
$$M_k = \begin{cases} 1 & (0.3 \leqq \Omega_k \leqq 0.7) \\ 0 & \text{(elsewhere)} \end{cases}$$

$$\theta_k = \begin{cases} 1-2\Omega_k & (0.3 \leqq \Omega_k \leqq 0.7) \\ 0 & \text{(elsewhere)} \end{cases}$$



(c) Linear-phase filter. $\lambda = 1000$.

Figure 4.- Concluded.

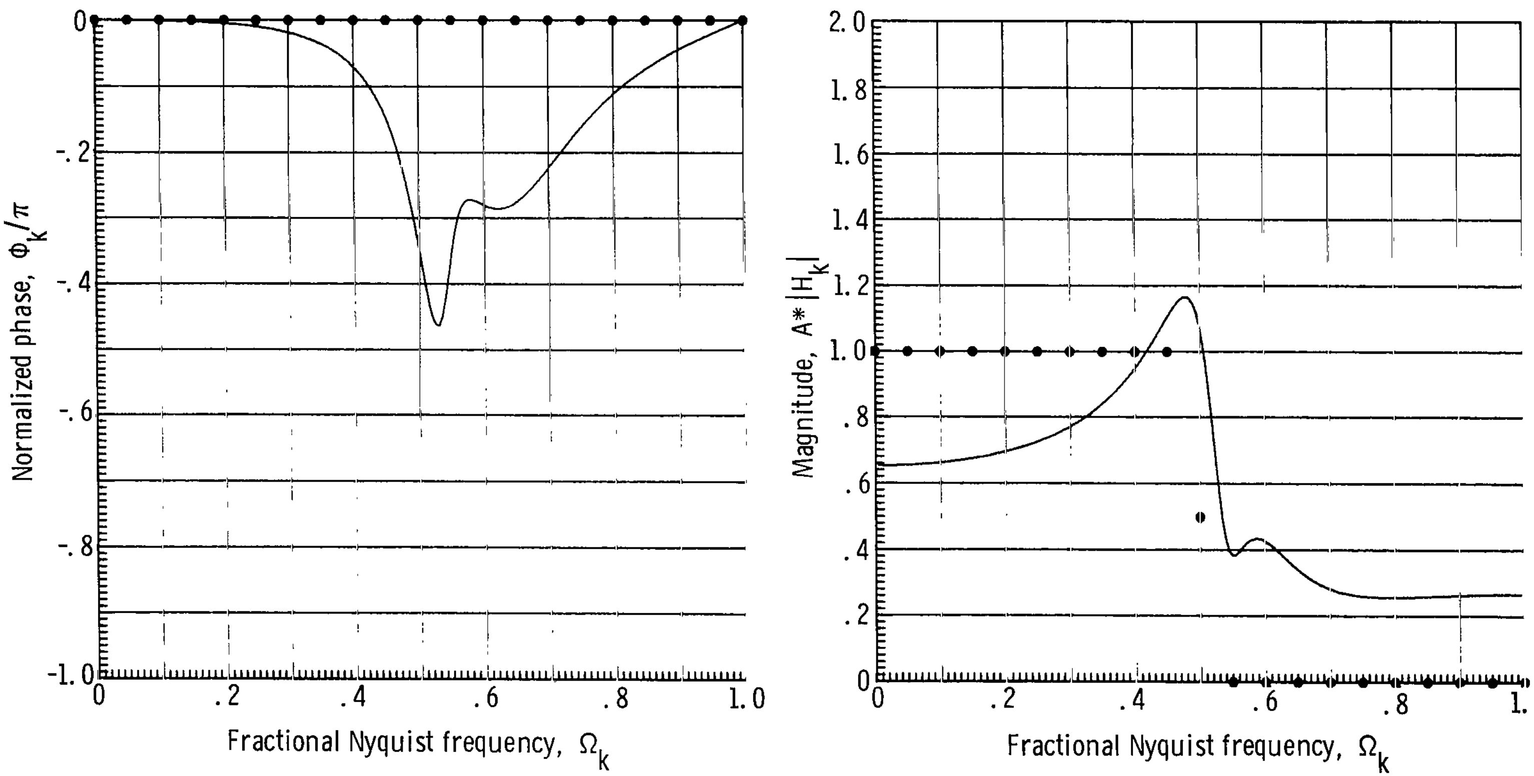


(a) Unspecified-phase filter. $\lambda = 0$.

Figure 5.- Two-stage low-pass filters.

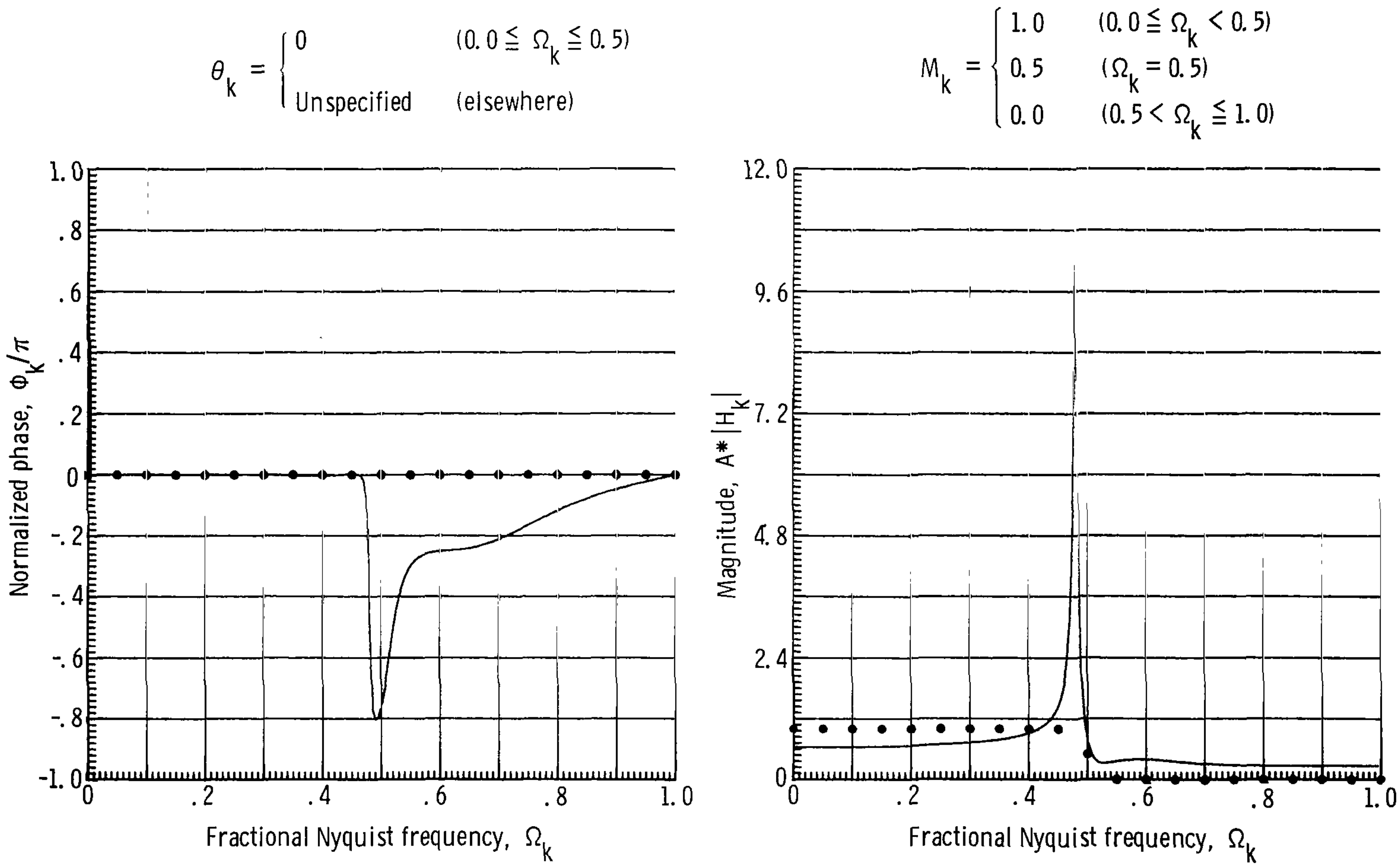
$$\theta_k = \begin{cases} 0 & (0.0 \leqq \Omega_k \leqq 0.5) \\ \text{Unspecified} & (\text{elsewhere}) \end{cases}$$

$$M_k = \begin{cases} 1.0 & (0.0 \leqq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leqq 1.0) \end{cases}$$



(b) Zero-phase filter. $\lambda = 10$.

Figure 5.- Continued.

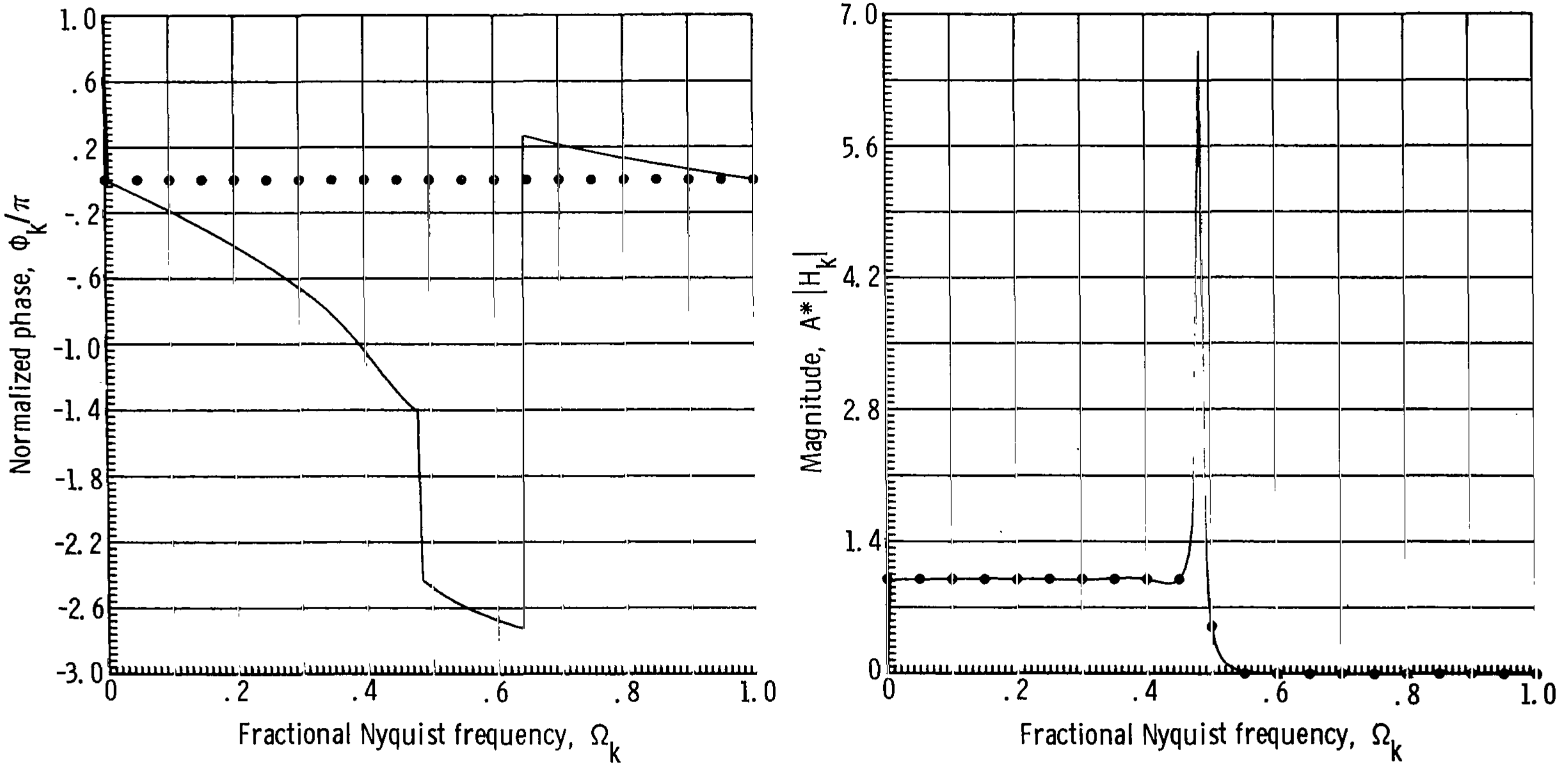


(c) Zero-phase filter. $\lambda = 1000$.

Figure 5.- Concluded.

θ_k = Unspecified $(0 \leqq \Omega_k \leqq 1)$

$$M_k = \begin{cases} 1.0 & (0.0 \leqq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leqq 1.0) \end{cases}$$

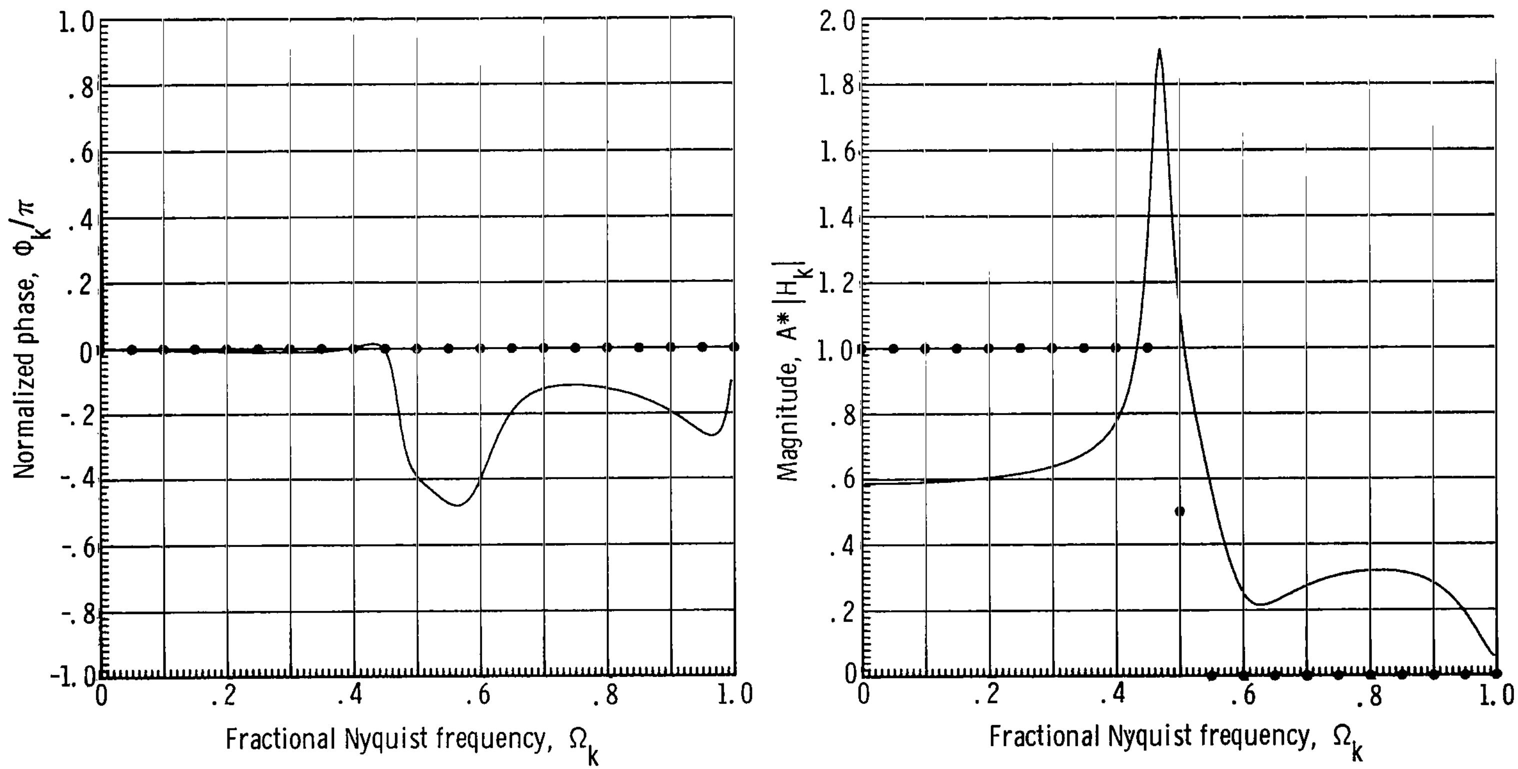


(a) Unspecified-phase filter. $\lambda = 0$.

Figure 6.- Three-stage low-pass filters.

$$\theta_k = \begin{cases} 0 & (0.0 \leqq \Omega_k \leqq 0.5) \\ \text{Unspecified} & (\text{elsewhere}) \end{cases}$$

$$M_k = \begin{cases} 1.0 & (0.0 \leqq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leqq 1.0) \end{cases}$$

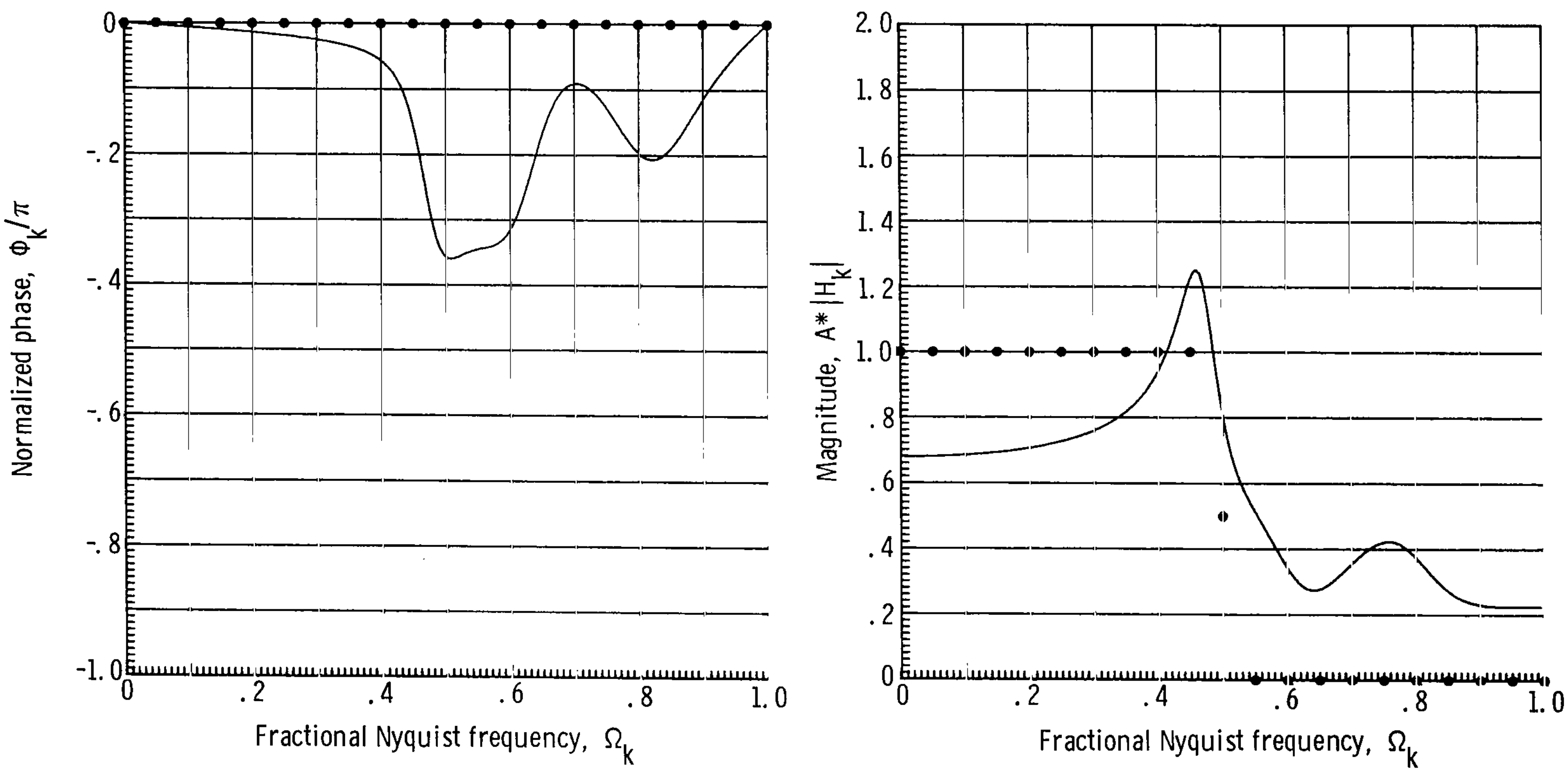


(b) Zero-phase filter. $\lambda = 10$.

Figure 6.- Continued.

$$\theta_k = \begin{cases} 0 & (0.0 \leqq \Omega_k \leqq 0.5) \\ \text{Unspecified} & (\text{elsewhere}) \end{cases}$$

$$M_k = \begin{cases} 1.0 & (0.0 \leqq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leqq 1.0) \end{cases}$$



(c) Zero-phase filter. $\lambda = 1000$.

Figure 6.- Concluded.

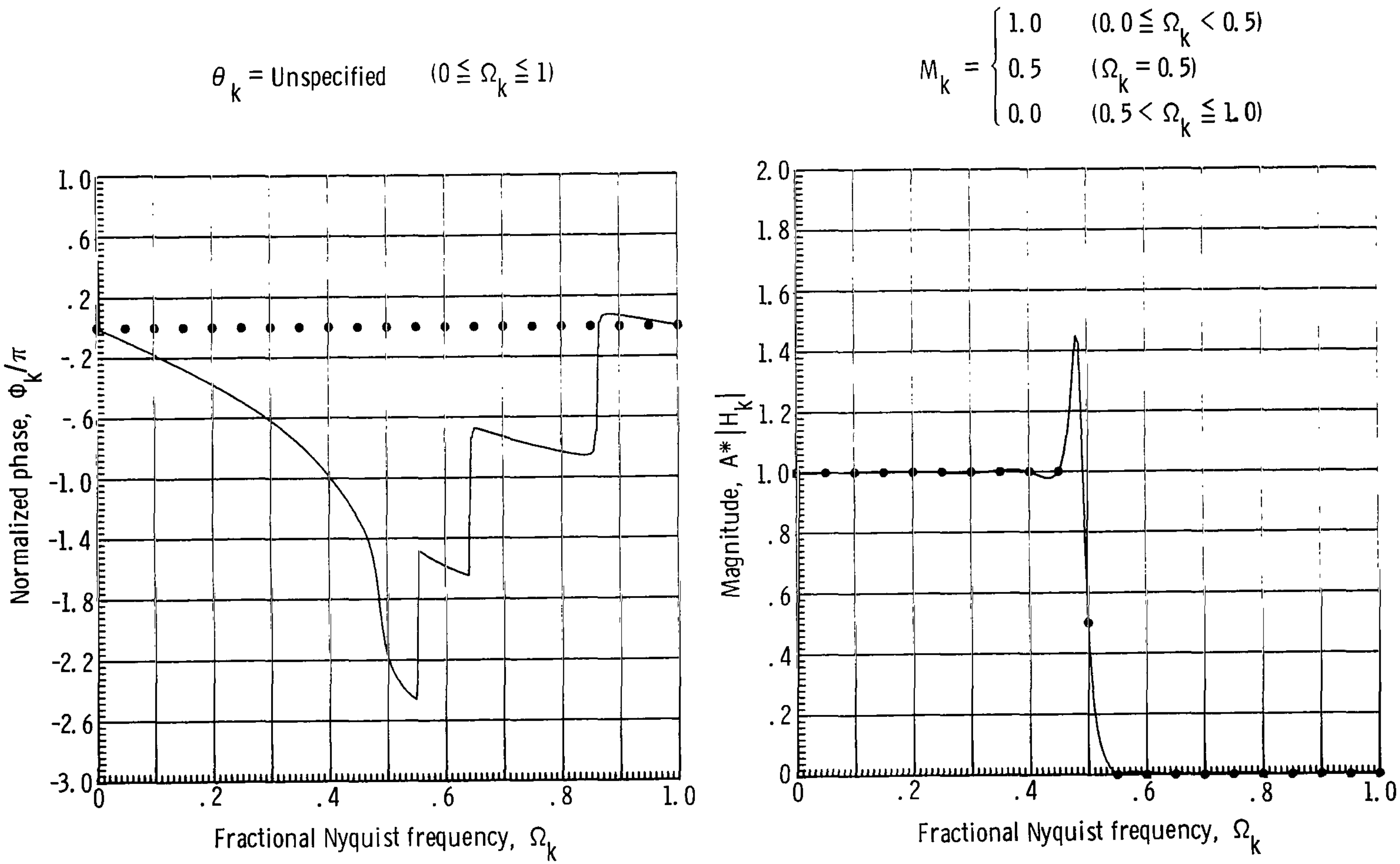


Figure 7.- Three-stage low-pass filter.

NASA-Langley, 1972 — 8 L-8162

NATIONAL AERONAUTICS AND SPACE ADMISTRATION
WASHINGTON, D.C. 20546

OFFICIAL BUSINESS
PENALTY FOR PRIVATE USE $300

FIRST CLASS MAIL

POSTAGE AND FEES PAID
NATIONAL AERONAUTICS AND
SPACE ADMINISTRATION

U.S.MAIL

029 001 C1 U 08 720317 S00903DS
DEPT OF THE AIR FORCE
AF WEAPONS LAB (AFSC)
TECH LIBRARY/WLOL/
ATTN: E LOU BOWMAN, CHIEF
KIRTLAND AFB NM 87117

POSTMASTER: If Undeliverable (Section 158 Postal Manual) Do Not Return

"The aeronautical and space activities of the United States shall be conducted so as to contribute . . . to the expansion of human knowledge of phenomena in the atmosphere and space. The Administration shall provide for the widest practicable and appropriate dissemination of information concerning its activities and the results thereof."

— NATIONAL AERONAUTICS AND SPACE ACT OF 1958

NASA SCIENTIFIC AND TECHNICAL PUBLICATIONS

TECHNICAL REPORTS: Scientific and technical information considered important, complete, and a lasting contribution to existing knowledge.

TECHNICAL NOTES: Information less broad in scope but nevertheless of importance as a contribution to existing knowledge.

TECHNICAL MEMORANDUMS: Information receiving limited distribution because of preliminary data, security classification, or other reasons.

CONTRACTOR REPORTS: Scientific and technical information generated under a NASA contract or grant and considered an important contribution to existing knowledge.

TECHNICAL TRANSLATIONS: Information published in a foreign language considered to merit NASA distribution in English.

SPECIAL PUBLICATIONS: Information derived from or of value to NASA activities. Publications include conference proceedings, monographs, data compilations, handbooks, sourcebooks, and special bibliographies.

TECHNOLOGY UTILIZATION PUBLICATIONS: Information on technology used by NASA that may be of particular interest in commercial and other non-aerospace applications. Publications include Tech Briefs, Technology Utilization Reports and Technology Surveys.

Details on the availability of these publications may be obtained from:

SCIENTIFIC AND TECHNICAL INFORMATION OFFICE

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Washington, D.C. 20546