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
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.
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 $\Omega_k$

$E_k^M$  magnitude error at $\Omega_k$

$E_k^\phi$  phase error at $\Omega_k$

$e_k$  error vector at $\Omega_k$

$\partial e_k / \partial A$  derivative of error vector at $\Omega_k$ 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

$|H_k|$  magnitude of $H(z)$ at $\Omega_k$

$H_k$  conjugate of $H(z)$ at $\Omega_k$

$\partial |H_k| / \partial p$  gradient vector of magnitude of $H(z)$ at $\Omega_k$ with respect to parameter vector

$I(\ )$  imaginary part of quantity

i, ..., N  denotes filter stage
\( \bar{J}_k \) \( k \) \( M_k \) \( N^i_k \) \( \vec{p} \) \( \vec{p}_i \) \( q^i_1(k) \) \( q^i_2(k) \) \( R(\cdot) \) \( u^i_1(k) \) \( V \) \( V_k \) \( \hat{V} \) \( \partial V / \partial A \) \( \partial V_k / \partial \bar{e}_k \) \( \bar{W}_k \) \( W^M_k \) \( W^\phi_k \) \( w^i(k) \) 

Jacobian at \( \Omega_k \):

\[
A^* \begin{bmatrix} \frac{\partial |H_k|}{\partial \vec{p}} & \frac{\partial \theta_k}{\partial \vec{p}} \end{bmatrix}
\]

sample point

specification magnitude at \( \Omega_k \)

numerator of \( i \)th stage of \( H(z) \) at \( \Omega_k \)

parameter vector

set of filter parameters for the \( i \)th stage, \( a_i, b_i, c_i, \) and \( d_i \)

first system state of \( i \)th stage at \( k \)th sample point

second system state of \( i \)th stage at \( k \)th sample point

real part of quantity

input to \( i \)th stage at \( k \)th sample point

criterion functional, that is, \( \impliedby(A,\vec{p}) \)

criterion functional at \( \Omega_k \), that is, \( \impliedby_k(A,\vec{p}) \)

reduced criterion functional, that is, \( \impliedby(A^*,\vec{p}) \)

slope of criterion functional with respect to zero frequency gain

gradient vector of criterion functional at \( \Omega_k \) with respect to error vector at \( \Omega_k \)

weighting matrix at \( \Omega_k \)

magnitude weighting at \( \Omega_k \)

phase weighting at \( \Omega_k \)

dummy variable of \( i \)th stage at \( k \)th sample point
\[ Y(z) \] digital filter discrete transfer function
\[ y_i(k) \] output of \textit{i}th stage at \textit{k}th sample point
\[ z \] transform variable
\[ z_k \] discrete transform variable at \[ \Omega_k, \ e^{j\pi\Omega_k} \]
\[ \theta_k \] specification phase at \[ \Omega_k \], radians
\[ \lambda \] collective phase weight
\[ \phi_k \] phase of \( H(z) \) at \[ \Omega_k \], radians
\[ \frac{\partial \phi_k}{\partial \mathbf{p}} \] gradient vector of phase of \( H(z) \) at \[ \Omega_k \] with respect to parameter vector
\[ \Omega_k \] fractional frequency at \textit{k}th 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

\[ Y(z) = \left\{ \prod_{1=1}^{N} \frac{1 + a_i z^{-1} + b_i z^{-2}}{1 + c_i z^{-1} + d_i z^{-2}} \right\} A H(z) \] (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 $i$th stage in figure 1 at the $k$th sample point

$$w^i(k) = A_1 u_1(k) - c_1 q_1^i(k) - d_1 q_2^i(k)$$

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

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

$$y_1(k) = w^i(k) + a_1 q_1^i(k) + b_1 q_2^i(k)$$

where

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

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

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

The Synthesis Problem

The design problem considered in this paper can be stated as follows: When the magnitude and phase specifications ($M_k$ and $\theta_k$, respectively) at the $k$th 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 $\overline{p}^*$ 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

$$V(A, \overline{p}) = \sum_k \langle \bar{e}_k, \bar{W}_k \bar{e}_k \rangle = \sum_k V_k$$  \hspace{1cm} (2)
where

$$\vec{e}_k = \begin{bmatrix} A |H_k| - M_k \\ \phi_k - \theta_k \end{bmatrix} = \begin{bmatrix} E^M_k \\ E^\phi_k \end{bmatrix}$$

is the error vector and

$$\vec{W}_k = \begin{bmatrix} W^M_k & 0 \\ 0 & \lambda W^\phi_k \end{bmatrix}$$

is the diagonal weighting matrix. Clearly, $V(A, \vec{p})$ is a nonlinear function of the parameter vector $\vec{p} = (a_1, b_1, c_1, d_1, \ldots, a_N, b_N, c_N, d_N)^T$, 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 $\tilde{V}(\vec{p}) = V(A^*, \vec{p})$ involving only $4N$ parameters. Thus

$$\frac{\partial V}{\partial A} = \sum_k \left\langle \frac{\partial \vec{e}_k}{\partial A} \frac{\partial V_k}{\partial \vec{e}_k} \right\rangle = 2 \sum_k \begin{bmatrix} |H_k| W^M_k \\ 0 \end{bmatrix} \vec{e}_k$$

and $\frac{\partial V}{\partial A} = 0$ implies

$$2 \sum_k |H_k| W^M_k (A^*|H_k| - M_k) = 0$$

or

$$A^* = \frac{\sum_k |H_k| W^M_k M_k}{\sum_k |H_k|^2 W^M_k} \quad (3)$$

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)
\[ \frac{\delta \hat{V}}{\delta \hat{p}} = 2 \sum_k \langle \hat{J}_k, \hat{W}_k \hat{c}_k \rangle \]  

(4)

where the \((4N \times 2)\) Jacobian \(\hat{J}_k\) is

\[ \hat{J}^T_k = \nabla_p \hat{e}_k = \left[ A^* \frac{\partial |H_k|}{\partial \hat{p}} ; \frac{\partial \phi_k}{\partial \hat{p}} \right]^T \]  

(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 \(\Omega_k\), it is readily shown (see ref. 6), where \(\overline{p}_i\) is the set of filter parameters for the \(i\)th stage, that

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

For the cascaded filter topology in terms of the elements of \(\overline{p}_i\),

\[ \frac{\partial |H_k|}{\partial a_i} = |H_k| R \left( \frac{z^{-1}_k}{\overline{N}_k} \right) \]

\[ \frac{\partial |H_k|}{\partial b_i} = |H_k| R \left( \frac{z^{-2}_k}{\overline{N}_k} \right) \]

\[ \frac{\partial |H_k|}{\partial c_i} = -|H_k| R \left( \frac{z^{-1}_k}{\overline{D}_k} \right) \]

and

\[ \frac{\partial |H_k|}{\partial d_i} = -|H_k| R \left( \frac{z^{-2}_k}{\overline{D}_k} \right) \]

and
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

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

which takes on a particularly simple form for the cascade topology. For the \( i \)th 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
\[
\frac{\partial \phi_k}{\partial d_1} = -i \left( \frac{z_k^{-2}}{D_k} \right)
\]

The special case of a one-stage \((N = 1)\) filter is illustrated. Here
\[
H_k = A \frac{1 + az_k^{-1} + bz_k^{-2}}{1 + cz_k^{-1} + dz_k^{-2}}
\]

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

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

Similarly,
\[
\frac{\partial \hat{V}}{\partial b} = \sum_k \left[ Q_k^M R_k \left( \frac{z_k^{-2}}{N_k} \right) + \lambda R_k^\phi \left( \frac{z_k^{-2}}{N_k} \right) \right]
\]
\[
\frac{\partial \hat{V}}{\partial c} = \sum_k \left[ Q_k^M R_k \left( \frac{z_k^{-1}}{D_k} \right) + \lambda R_k^\phi \left( \frac{z_k^{-1}}{D_k} \right) \right]
\]
\[
\frac{\partial \hat{V}}{\partial d} = \sum_k \left[ Q_k^M R_k \left( \frac{z_k^{-2}}{D_k} \right) + \lambda R_k^\phi \left( \frac{z_k^{-2}}{D_k} \right) \right]
\]

where
\[
Q_k^M = 2 E_k^M W_k^M |H_k|
\]
and

\[ R_k^0 = 2E_k^0 W_k^0 \]

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 \( \Gamma \), 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

\[ |z - z_0| = |z_0| \left|z - \frac{1}{z_0}\right|; \quad z \in \Gamma \]

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 $\epsilon = |\hat{V}_{\text{new}} - \hat{V}_{\text{old}}|$ 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 $\theta_k = 1 - 2\Omega_k \ (0 \leq \Omega_k \leq 1)$, and a two-stage filter was specified. When an initial parameter vector $\bar{p} = (0, 0, 0, 0.25, 0, 0, 0, 0)^T$ 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 $A = 1.0$

\[
\begin{align*}
a_1 &= 0 & b_1 &= -0.9871 & c_1 &= 0 & d_1 &= 0.0395 \\
a_2 &= 0 & b_2 &= -0.9871 & c_2 &= 0 & d_2 &= -0.0127
\end{align*}
\]
It is interesting to note that the phase requirements were met to within $0.008\pi$ 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 $0.3 \leq \Omega \leq 0.7$. 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 $0.0003\pi$ radian throughout the specified band.

The computed parameters for the lag case were

$$A = 1.0$$

$$a_1 = 0.5580 \quad b_1 = -0.1857 \quad c_1 = -0.4752 \quad d_1 = 0.0363$$

$$a_2 = 0.5580 \quad b_2 = -0.1857 \quad c_2 = -0.3712 \quad d_2 = -0.5686$$

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

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

$$A = 1.0$$

$$a_1 = -0.4768 \quad b_1 = -0.1548 \quad c_1 = 0.5022 \quad d_1 = -0.1082$$

$$a_2 = -0.4768 \quad b_2 = -0.1548 \quad c_2 = 0.4515 \quad d_2 = -0.2008$$

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

\[
M_k = \begin{cases} 
1 & \text{if } (0.3 \leq \Omega_k \leq 0.7) \\
0 & \text{Elsewhere}
\end{cases}
\]

\[
\theta_k = \begin{cases} 
1 - 2\Omega_k & \text{if } (0.3 \leq \Omega_k \leq 0.7) \\
0 & \text{Elsewhere}
\end{cases}
\]

Equal error and frequency weights were employed and the effects of changes in \( \lambda \) 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

\( \lambda = 0 \):

\[
\begin{align*}
A &= 0.2063 \\
a_1 &= 0.0000 & b_1 &= -1.0000 & c_1 &= 0.0000 & d_1 &= 0.1539 \\
a_2 &= 0.0000 & b_2 &= -1.0000 & c_2 &= 0.0000 & d_2 &= 0.1539
\end{align*}
\]

\( \lambda = 10 \):

\[
\begin{align*}
A &= 0.3658 \\
a_1 &= -0.9754 & b_1 &= 0.7300 & c_1 &= 0.4529 & d_1 &= 0.7211 \\
a_2 &= 0.8632 & b_2 &= 0.5632 & c_2 &= -0.6119 & d_2 &= 0.7443
\end{align*}
\]
\[ \lambda = 1000: \]

\[
A = 0.4232
\]

\[
a_1 = -1.1739 \quad b_1 = 0.8489 \quad c_1 = 0.7596 \quad d_1 = 0.6691
\]

\[
a_2 = 1.1739 \quad b_2 = 0.8489 \quad c_2 = -0.7596 \quad d_2 = 0.6691
\]

**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

\[
M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0)
\end{cases}
\]

\[
\theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & (\text{Elsewhere})
\end{cases}
\]

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

\[ \lambda = 0: \]

\[
A = 0.1196
\]

\[
a_1 = 1.0240 \quad b_1 = 1.0000 \quad c_1 = -0.1713 \quad d_1 = 0.7676
\]

\[
a_2 = 1.0240 \quad b_2 = 1.0000 \quad c_2 = -0.5324 \quad d_2 = 0.2286
\]
\( \lambda = 10: \)

\[
\begin{align*}
A &= 0.4879 \\
a_1 &= 0.2018 \quad b_1 &= 0.6684 \quad c_1 &= 0.3560 \quad d_1 &= 0.4612 \\
a_2 &= 0.6597 \quad b_2 &= 0.4335 \quad c_2 &= 0.0806 \quad d_2 &= 0.7671
\end{align*}
\]

\( \lambda = 1000: \)

\[
\begin{align*}
A &= 0.5343 \\
a_1 &= 0.0205 \quad b_1 &= 0.7169 \quad c_1 &= -0.0836 \quad d_1 &= 0.6255 \\
a_2 &= 0.6286 \quad b_2 &= 0.7905 \quad c_2 &= 0.2123 \quad d_2 &= 0.6681
\end{align*}
\]

The optimum parameters for the three-stage filter were

\( \lambda = 0: \)

\[
\begin{align*}
A &= 0.0510 \\
a_1 &= 0.8537 \quad b_1 &= 1.0000 \quad c_1 &= -0.1068 \quad d_1 &= 1.0000 \\
a_2 &= 0.8537 \quad b_2 &= 1.0000 \quad c_2 &= -0.4046 \quad d_2 &= 0.5990 \\
a_3 &= 0.8537 \quad b_3 &= 1.0000 \quad c_3 &= -0.6799 \quad d_3 &= 0.2069
\end{align*}
\]

\( \lambda = 10: \)

\[
\begin{align*}
A &= 0.5109 \\
a_1 &= 1.3302 \quad b_1 &= 0.5515 \quad c_1 &= -0.1731 \quad d_1 &= 0.8097 \\
a_2 &= 0.6844 \quad b_2 &= 0.7157 \quad c_2 &= 1.1880 \quad d_2 &= 0.5850 \\
a_3 &= -0.0373 \quad b_3 &= 0.7012 \quad c_3 &= 0.3825 \quad d_3 &= 0.5262
\end{align*}
\]
\[ \lambda = 1000: \]

\[ A = 0.4515 \]

\[ a_1 = 1.5107 \quad b_1 = 0.5286 \quad c_1 = -0.1771 \quad d_1 = 0.8972 \]

\[ a_2 = 0.5825 \quad b_2 = 0.7490 \quad c_2 = 1.3094 \quad d_2 = 0.4191 \]

\[ a_3 = -0.1663 \quad b_3 = 0.7485 \quad c_3 = 0.2002 \quad d_3 = 0.6393 \]

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 \( \mathbf{p} = (0, 0, 0, 0, 0, 0, 0, 0.25, 0, 0, 0) \) 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, TAPF5=INPUT, TAPF6=CUTOLT, PUNCH)
0000003 EXTERNAL FUNCT
0000003 DIMENSION M(104),X(161),G(16)
0000003 CMMV/RMV/W(1001),Y(1001),K,PHASED(1001),ALAMDA,FR,W(1001)
0000003 C WTP(1001),KYP
0000003 CMMV/RMVI(CALL,KCALL,LIN)
0000003 CALL CALCCMP
0000003 CALL LFRY
0000005 WRITE(*,-1)
0000011 51 FORMAT(* INPUT DATA*)
0000011 M=0
0000012 30 K=M+1
0000014 50 FORMAT(21),(M),Y(1),PHASED(1),W(1),KYP(1)
0000015 21 FORMAT(F10.5)
0000017 22 FORMAT(1=,13,= W=F7.4,= Y=F7.4,= PHASED=F7.4,= F7.4)
0000021 (C * WTM=+,F7.4,= WTP=+,F7.4)
0000025 IF(MLT.1,CC)GCT=30
0000029 QH=15 J=1,16
0000033 1* XIJ=0,00
0000037 X(14)=-.25
0000041 RPACIS(5,5)=L,LK,FST,FPS,HPAX,ALAMDA,FR,KYP
0000045 50 FORMAT(215,5F15.5,1)
0000049 IF(FR.LT.001) FR=1.
0000053 QH=4M(1)
0000057 IF(FXF,5) EGRID
0000061 488 CONTINUE
0000065 488 IF(IF=511L,LIP,FST,FPS,HPAX,FR,ALAMDA)
0000069 61 FORMAT(* L=,12,= L1=+,15,= EST=+,F10.5,= FPS=+,F10.5,= HP=+,F10.5,= FREQUENCY=+,F10.5,= LAMBDA=+,F10.5)
0000073 ICALL=0
0000077 IF(IF=0) KCALL=0
0000081 CALL FLPSL(FUNCT,N,X,F,G,FST,FPS,FR,IFP,H)
0000085 CALL RMTSN(N,X)
0000089 CALL INSIDEIN(X,KFLAG)
0000093 WRITE(6,261)EKR,KFLAG,ICALL,KCALL
0000097 IF(KFLAG=15,= KFLAG=+,15,= ICALL=+,IF,= KCALL=+,15)
0000099 IF(KCALL,.GT.1000) GTO 0R
00000103 IF(KFLAG,AF,EF,EF,AE,EO,AND,ICALL,LF,LIN) GC TC OR
00000117 CALL HMTSN(X)
00000121 ICALL=-10
00000125 IF(IF=14) KCALL=0
00000129 CALL FUNCT(N,X,F,G,HPAX)
00000133 GOTO 0C
00000137 CALL CALPLT(C,*,*,91)
00000141 STOP
00000145 END
```
APPENDIX – Continued

003427  DC 423 J=1, 4  0120
003429  J4=1 J=1 4
003441  G(J4+1)=G(J4+1) J*CLR  0128
003441  005 G(J4+2)=G(J4+2) 0*CLR?  0129
003450  006 J=7, M*Y[1] / TUM[1, 1]  0130
003475  G(J4+1)=G(J4+1) 0*CLR  0131
003475  007 G(J4+2)=G(J4+2) 0*CLR?  0132
003451  008 G(J4+3)=G(J4+3) 0*CLR?  0133
003453  009 CONTINUE  0134
007571  ON 42 J=1, 4  0135
007571  J4=1 J=1 4  0136
007571  00 G(J4+1)=G(J4+1) J*CLR  0138
007571  005 G(J4+2)=G(J4+2) 0*CLR?  0139
007571  006 J=7, M*Y[1] / TUM[1, 1]  0140
007571  007 G(J4+1)=G(J4+1) 0*CLR  0141
007571  008 G(J4+2)=G(J4+2) 0*CLR?  0142
007571  009 CONTINUE  0143
008013  045 F=1*4 ALM0DAV2  0144
008016  004 ICALL #ICALL+1  0145
008020  006 ICALL=ICALL+1  0146
008021  006 ICALL=ICALL+1  0147
008024  ICALL=ICALL+1  0148
008024  WRITE(6,3)  0149
008024  ICALL=ICALL+1  0150
008024  WRITE(6,5)  0151
008024  ICALL=ICALL+1  0152
008024  ICALL=ICALL+1  0153
008024  ICALL=ICALL+1  0154
008024  ICALL=ICALL+1  0155
008024  ICALL=ICALL+1  0156
008024  ICALL=ICALL+1  0157
008024  ICALL=ICALL+1  0158
008024  ICALL=ICALL+1  0159
008024  ICALL=ICALL+1  0160
008024  ICALL=ICALL+1  0161
008024  ICALL=ICALL+1  0162
008024  ICALL=ICALL+1  0163
008024  ICALL=ICALL+1  0164
008024  ICALL=ICALL+1  0165
008024  ICALL=ICALL+1  0166
008024  ICALL=ICALL+1  0167
008024  ICALL=ICALL+1  0168
008024  ICALL=ICALL+1  0169
008024  ICALL=ICALL+1  0170
008024  ICALL=ICALL+1  0171
008024  ICALL=ICALL+1  0172
008024  ICALL=ICALL+1  0173
008024  ICALL=ICALL+1  0174
008024  ICALL=ICALL+1  0175
008024  ICALL=ICALL+1  0176
008024  ICALL=ICALL+1  0177
008024  ICALL=ICALL+1  0178
008024  ICALL=ICALL+1  0179
008024  ICALL=ICALL+1  0180
008024  ICALL=ICALL+1  0181
008024  ICALL=ICALL+1  0182
008024  ICALL=ICALL+1  0183
008024  ICALL=ICALL+1  0184
008024  ICALL=ICALL+1  0185
008024  ICALL=ICALL+1  0186
008024  ICALL=ICALL+1  0187
008024  ICALL=ICALL+1  0188
008024  ICALL=ICALL+1  0189
008024  ICALL=ICALL+1  0190
APPENDIX – Continued

```plaintext
001721  IF(PHASE=1YMAX3), 301, 301, 301
001725  401  YMAX3=PHASE
001730  301  CONTINUE
001730  401  IF(PHASE=1YMIN3), 402, 302, 302
001733  402  YMIN3=PHASE
001735  302  CONTINUE
001735  401  IF(AL=1YMAX11), 300, 300, 300
001740  300  YMAX11=AL
001742  300  CONTINUE
001745  62  FORMAT(6=PhASEx, 15, Z, *. PHASE/F=15.6, A YMIN/F=15.6)
001761  C SCALE COMPUTATIONS
001761  401  IYMAX11=1YMAX11, , , 9999
001764  401  YMAX1=FLATAT(IYMAX11)
001766  401  IF(HMAX, =EQ., 0.), GO TO 332
001767  401  YMAX1=HMAX
001770  333  CONTINUE
001770  401  IYMAX33=1YMAX33, , , 9999
001773  401  YMAX3=FLAT AT(IYMAX33)
001775  401  IYMIN33=1YMIN33, , , 9999
001778  401  YMIN3=FLAT AT(IYMIN33)
001781  410  ADD=1F+ F/S(2)
001784  410  AMAG=1AMAGITUDES
001787  410  AFRI=1F0. # F
001790  410  NFR=700
001793  410  NFR=200
001800  C MAGNITUDE (COMPUTED) – FREQUENCY PLOT
001800  CALL INPLT(6, NFR, FFR, X, F, , . , , , , , YMAX1, C, 5, 10, AXF, 10, C, 0
001806  C MAGNITUDE (DESIRED) – FREQUENCY PLOT
001807  CALL INPLT(1, M, F, Y, 1, C, YMAX1, C, 5, 10, AXF, 10, C, 0
001814  C PHASE–FREQUENCY PLOT
001815  CALL INPLT(7, NFF, YFF, 1, F, YMAX1, 10), FY, YMIN3, YMAX3, C, 5, 10, AXF, 10, C, 0
001822  C PHASE–FREQUENCY PLOT
001823  CALL INPLT(7, NFF, YFF, 1, F, YMAX1, C, 5, 10, AXF, 10, C, 0
001830  775  IF(R=1)
001833  776  IF(I=1)
001836  777  IF(K=1)
001839  778  IF(L=1)
001842  617  IF(R=1)
001845  618  IF(M=1)
001848  619  IF(NO=1)
001851  617  IF(R=1)
001854  618  IF(M=1)
001857  619  IF(NO=1)
001860  617  IF(R=1)
001863  618  IF(M=1)
001866  619  IF(NO=1)
001869  617  IF(R=1)
001872  618  IF(M=1)
001875  619  IF(NO=1)
001878  617  IF(R=1)
001881  618  IF(M=1)
001884  619  IF(NO=1)
001887  617  IF(R=1)
001890  618  IF(M=1)
001893  619  IF(NO=1)
001896  617  IF(R=1)
001899  618  IF(M=1)
001902  619  IF(NO=1)
001905  617  IF(R=1)
001908  618  IF(M=1)
001911  619  IF(NO=1)
20
```
<table>
<thead>
<tr>
<th>Column 1</th>
<th>Column 2</th>
<th>Column 3</th>
<th>Column 4</th>
<th>Column 5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
<tr>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
<td>Data</td>
</tr>
</tbody>
</table>

**APPENDIX - Continued**
APPENDIX – Continued

000327  OALFA=OALFA=OALFA/DALFA=CY/ALFA
000333  IFCALFA) 51,25,25
000335  W=WALFA SORTICALFA)
000340  ALFA = (DY=W-J*AMBDAXOY*2, CF=O-W-DO)
000351  DO 26 I=1,N
000356  X(J)=J*IT-ALFA)+H(I)
000366  CALL FUNCTIONX,F,G,HFAK)
000400  IF(CALLGT.300) GO TO 725
000403  IF(CALLGT.LIM) GO TO 725
000406  GO TO 919
000409  725 ITEN=3
000410  725 IFEN
000413  27  IF(F-FX) 36,36,24
000416  28 OALFA=O,00
000417  DO 29 I=1,N
000421  29 OALFA=OALFA G(I)*H(I)
000423  IF(CALFA 30,33,33
000424  30 IF(F-FX) 32,31,33
000426  31 IF(FX=OALFA) 32,36,32
000436  32 FX=F
000437  OX=OALFA
000440  T=OALFA
000442  AMBDA=OALFA
000443  GO TO 23
000446  33 IF FLY-F) 35,34,35
000445  34 IF(FY-OALFA) 35,36,35
000447  35 FY=F
000450  OY=OALFA
000451  AMPFA=AMBDA=ALFA
000454  GO TO 22
000457  DO 37 J=1,N
000456  K=N+J
000457  H(K)=G(J)-H(K)
000463  K=K+N
000464  37 H(K)=X(J)-H(K)
000477  IF(IFDF-FPS) 51,38,38
000476  38 IFR=O
000477  IF(KOUNT-N) 42,39,39
000501  39 T=O,00
000502  39 Z=O,00
000504  39 DC 40 J=1,N
000504  K=N+J
000506  W=H(K)
000510  K=K+N
000511  T=T+FPS(H(K)
000514  40 Z=Z+H(K)
000521  41 IF(IFRE-FPS) 41,41,42
000526  41 IF(IFEPS) 54,56,47
000531  47 IF(KOUNT-LIMIT) 43,5,5C
000534  43 ALFA=O,C0
000535  DO 47 J=1,N
000537  K=J+N
000540  W=O,00
000542  DC 4= L=1,N
000543  K=H+L
000544  W=H+H(K)*H(K)
000552  IF(L-J) 44,45,46
000554  44 K=K+N-L
000557  GO TO 46
000557  45 K=K+1
000561  46 CONTINUE
000564  K=N+J
000565  ALFA=OALFA+W=O(H(K)
APPENDIX – Continued

SUBROUTINE INSIDE(N, X, KFLAG)
DIMENSION X(16)
J=1
KFLAG=0
10 J=J+2
000011 IF(J.GT.N)RETURN
000014 E=-5000*X(J)
000016 C=X(J+1)
000020 DISC=C*B-C
000022 IF(DISC.LT.0.00)GOTO20
C......REAL ROOTS
000024 DISC= SQRT(DISCS)
000025 R1=B+DISC
000027 R2=B-DISC
000031 CR1= ABS(R1)
000033 DR2= ABS(R2)
000035 IF(DR1.LE.1.00.AND.DR2.LE.1.00)GOTO10
000051 KFLAG=1
000054 IF(DR1.GT.1.00)R1=1.00/R1
000060 X(J+1)*R1*R2
000064 X(J+1)=R1*R2
000066 GOTO10
C......COMPLEX ROOTS
20 IF(C.GT.1.00)GOTO10
000071 KFLAG=1
000072 X(J+1)=C
000074 X(J)=X(J)*C
000076 GOTO10
000077 END
SUBROUTINE ROOTS(N,X)
    DIMENSION X(16)
    WRITE(6,40)
    40 FORMAT(* ROOTS*/6X,**REAL*11X,**IMAG*11X,**REAL*11X,**IMAG*)
    J=1
    10 J=J+2
    IF(J.GT.N)RETURN
    B=-.5*X(J)
    C=X(J+1)
    DISC=B*B-C
    IF(DISC.LE.0.0)GOTO 20
    C.****REAL ROOTS
    DISC = SORT(DISC)
    R1=B+DISC
    R2=B-DISC
    WRITE(6,30)R1,R2
    30 FORMAT(* **F15.815X**F15.8)
    GOTO10
    C.****COMPLEX ROOTS
    20 DISC = SORT(-1.00*DISC)
    DISCM=-1.00*DISC
    WRITE(6,50)B,DISC,B,DISCM
    50 FORMAT(* **F15.815X**F15.8)
    GOTO10
    END
REFERENCES


Figure 1.- Signal flow graph of cascaded digital filter.
\[ \theta_k = 1 - 2 \Omega_k \quad (0 \leq \Omega_k \leq 1) \]

\[ M_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

**Figure 2.- Two-stage linear-phase filter.**
\[ \theta_k = \begin{cases} -\pi/2 & (0.3 \leq \Omega_k \leq 0.7) \\ \text{Unspecified} & \text{(elsewhere)} \end{cases} \]

\[ M_k \text{ = Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

(a) Lag filter.

Figure 3.- Two-stage constant-phase filters.
\[ \theta_k = \begin{cases} \frac{\pi}{2} & (0.3 \leq \Omega_k \leq 0.7) \\ \text{Unspecified} & \text{(elsewhere)} \end{cases} \]

\[ M_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

(b) Lead filter.

Figure 3. Concluded.
\[ M_k = \begin{cases} 1 & (0.3 \leq \Omega_k \leq 0.7) \\ 0 & \text{(elsewhere)} \end{cases} \]

\[ \theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

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

Figure 4. Two-stage limited-band constant-gain filters.
\[ M_k = \begin{cases} 
1 & (0.3 \leq \Omega_k \leq 0.7) \\
0 & \text{(elsewhere)}
\end{cases} \]

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

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

Figure 4.- Continued.
\[ M_k = \begin{cases} 1 & (0.3 \leq \Omega_k \leq 0.7) \\ 0 & \text{(elsewhere)} \end{cases} \]

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

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

Figure 4.— Concluded.
(a) Unspecified-phase filter. $\lambda = 0$.

Figure 5.- Two-stage low-pass filters.
\[ \theta_k = \begin{cases} 0 & (0.0 \leq \Omega_k \leq 0.5) \\ \text{Unspecified} & \text{elsewhere} \end{cases} \]

\[ M_k = \begin{cases} 1.0 & (0.0 \leq \Omega_k < 0.5) \\ 0.5 & (\Omega_k = 0.5) \\ 0.0 & (0.5 < \Omega_k \leq 1.0) \end{cases} \]

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

Figure 5.- Continued.
\[ \theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & \text{(elsewhere)} 
\end{cases} \]

\[ M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0) 
\end{cases} \]

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

Figure 5.- Concluded.
\( \theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \)

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

Figure 6.- Three-stage low-pass filters.
\( \theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & \text{(elsewhere)} 
\end{cases} \)

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

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

Figure 6.- Continued.
\[ \theta_k = \begin{cases} 
0 & (0.0 \leq \Omega_k \leq 0.5) \\
\text{Unspecified} & \text{(elsewhere)}
\end{cases} \]

\[ M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0)
\end{cases} \]

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

Figure 6. - Concluded.
\[ \theta_k = \text{Unspecified} \quad (0 \leq \Omega_k \leq 1) \]

\[ M_k = \begin{cases} 
1.0 & (0.0 \leq \Omega_k < 0.5) \\
0.5 & (\Omega_k = 0.5) \\
0.0 & (0.5 < \Omega_k \leq 1.0) 
\end{cases} \]

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