NASA CONTRACTOR REPORT

NASA CR-1577

A METHOD FOR DETERMINING THE CHARACTERISTIC FUNCTIONS ASSOCIATED WITH THE AEROELASTIC INSTABILITIES OF HELICOPTER ROTORS IN FORWARD FLIGHT

by Vincent J. Piarulli and Richard P. White, Jr.

Prepared by
ROCHESTER APPLIED SCIENCE ASSOCIATES, INC.
Rochester, N. Y.
for Langley Research Center

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION • WASHINGTON, D. C. • JUNE 1970

1. Report No. NASA CR-1577	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle A METHOD FOR DETERMINING THE CHARACTERISTIC FUNCTIONS ASSOCIATED WITH THE AEROELASTIC INSTABILITIES OF HELICOPTER ROTORS IN FORWARD FLIGHT		5. Report Date June 1970
		6. Performing Organization Code
7. Author(s) Vincent J. Piarulli and Richard P. White, Jr.		8. Performing Organization Report No. 69 - 10
9. Performing Organization Name and Address Rochester Applied Science Associates, Inc. 100 Allens Creek Road Rochester, N.Y. 14618		10. Work Unit No. 721-02-10-04
		11. Contract or Grant No. NAS 1-9261
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report
		14. Sponsoring Agency Code

15. Supplementary Notes

This report supplements material presented in NASA CR-1332 entitled "A Method for Analyzing the Aeroelastic Stability of a Helicopter Rotor in Forward Flight," 1969.

16. Abstract

A method has been developed for determining the characteristic functions (modal content) of aeroelastic instabilities experienced by helicopter rotors in forward flight. The method assumes a knowledge of the characteristic values which characterize the frequency and growth rate of an unstable mode of a helicopter rotor in a given flight condition. Characteristic values may be found from the previously developed program (ref. 1) which is capable of analyzing a coupled set of linear, second-order differential equations with periodically varying coefficients. The necessary formulation was programed for the case of a system with three degrees of freedom. Calculations were carried out for comparison with available experimental data.

17. Key Words (Suggested by Author(s)) Helicopter rotor; Rotor blade; Flutter	18. Distribution Statement Unclassified – Unlimited

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 75	22. Price* $3.00

*For sale by the Clearinghouse for Federal Scientific and Technical Information
Springfield, Virginia 22151

CONTENTS

	Page
SUMMARY	1
INTRODUCTION	1
SYMBOLS	2
DEVELOPMENT OF THE EQUATIONS FOR DETERMINING CHARACTERISTIC FUNCTIONS	3
DESCRIPTION OF COMPUTER PROGRAM	11
APPLICATION OF THE METHOD	13
CONCLUDING REMARKS	21
APPENDIX A: Listing of Computer Program for Determining Characteristic Eigenvalues (Update of Program Listed in Reference 1)	23
APPENDIX B: Listing of Computer Program for Determining Characteristic Modal Functions	67
REFERENCES	73

A METHOD FOR DETERMINING THE CHARACTERISTIC FUNCTIONS ASSOCIATED WITH THE AEROELASTIC INSTABILITIES OF HELICOPTER ROTORS IN FORWARD FLIGHT

By Vincent J. Piarulli and Richard P. White, Jr.
ROCHESTER APPLIED SCIENCE ASSOCIATES, INC.

SUMMARY

A method has been developed for determining the characteristic functions (modal content) of aeroelastic instabilities experienced by helicopter rotors in forward flight. The method assumes a knowledge of the characteristic values which characterize the frequency and growth rate of an unstable mode of a helicopter rotor in a given flight condition. Characteristic values may be found from the previously developed program (Reference 1) which is capable of analyzing a coupled set of linear, second-order differential equations with periodically varying coefficients.

The necessary formulation was programmed for the case of a system with three degrees of freedom. Calculations were carried out for comparison with available experimental data.

INTRODUCTION

An analysis of the aeroelastic stability of a helicopter rotor in forward flight was carried out in Reference 1. In that study a computer program was developed which is capable of determining the characteristic values of a given set of coupled, linear, second-order differential equations with periodically varying coefficients.

Stability properties determined by that program consist solely of the real and imaginary parts of the system characteristic values. A knowledge of these quantities alone is akin to knowing the natural frequencies and rates of exponential growth or decay associated with each of the natural modes without knowing the actual modal content.

The study described herein was directed at developing the characteristic functions (modal content) associated with the stability of a helicopter rotor in forward flight. Characteristic functions clearly give an indication as to the degrees of freedom excited in a particular unstable mode. More importantly, however, a close inspection of the characteristic functions should yield insight towards the redesign required to eliminate an instability.

The method developed here relies heavily on the basic formulation and computer program previously developed in Reference 1. Therefore, the reader is referred to Reference 1 for a more complete description of the fundamentals of the general approach and to Appendix A of the present report for a corrected listing of that computer program.

SYMBOLS

a_{mn}, b_{mn}	dimensionless, periodically varying coefficients occurring in the basic set of differential equations
$[a]$,$[b]$	matrix form of periodically varying coefficients a_{mn} and b_{mn}
$[A]^{(k)}$, $[B]^{(k)}$	arrays of Fourier coefficients required to represent the matrices $[a]$ and $[b]$ in complex Fourier series
$[\mathcal{A}]^{(k)}$, $[\mathcal{B}]^{(k)}$	complex arrays related to the quantities $[A]^{(k)}$ and $[B]^{(k)}$ through Eqs. (14).
N	number of degrees of freedom of the elastomechanical system
N_f	number of Fourier components retained in Fourier representation of characteristic functions
$\{p\}^{(k)}$	columns of Fourier coefficients of the characteristic functions
$\{q\}^{(k)}$	columns related to $\{p\}^{(k)}$ through the change in index defined by Eq. (13)
R	rotor radius, m
V_f	magnitude of free-stream velocity (aircraft forward speed), m/s
$[T]$	array formed by computer program in order to solve for characteristic functions
$\{x\}$	large column containing all of the $\{p\}^{(k)}$
z	nondimensional quantity corresponding to time

ζ_n	displacement of the $n^{\underline{th}}$ coupled mode of free vibration of the elasto-mechanical system
$\{\zeta\}$	column form of ζ_n
λ	system characteristic value
λ_R	real part of λ
λ_I	imaginary part of λ
μ	advance radio, $\frac{V_f}{\Omega R}$ nondimensional
$\phi_m(z)$	characteristic function corresponding to $m^{\underline{th}}$ generalized coordinate
$\{\phi\}(z)$	column representation of $\phi_m(z)$
Ω	rotor rotational speed, rad/s
$\bar{\omega}_k$	natural frequency of the $k^{\underline{th}}$ coupled mode of free vibration of the rotating system, rad/s

DEVELOPMENT OF THE EQUATIONS FOR DETERMINING CHARACTERISTIC FUNCTIONS

The perturbation equations of motion of a rotor blade in forward flight were shown in Reference (1) to be expressible in the following form.

$$\frac{d^2\zeta_m}{dz^2} + \sum_{n=1}^{N}\left[a_{mn}\frac{d\zeta_n}{dz} + b_{mn}\,\zeta_n\right] = 0 \qquad (m=1,2,\ldots N) \qquad (1)$$

where: z is a dimensionless indicator of time defined by $z = \Omega t/2$

the ζ_n's are the N generalized coordinates defining the motions of the dynamic system

the a_{mn}'s and b_{mn}'s are periodic functions of z such that

$$a_{mn}(z + \pi) = a_{mn}(z)$$

$$b_{mn}(z + \pi) = b_{mn}(z)$$

The theory of Floquet (Reference 2) can be used to show that a solution to Eqs.(1) must be of the form

$$\zeta_m(z) = e^{\lambda z} \phi_m(z) \tag{2}$$

where λ is a complex constant and $\phi_m(z)$ is periodic with a period π. The differential system, Eq.(1), is of order 2N, so there are 2N linearly independent solutions. Hence, there are 2N values of λ and 2N associated sets of N functions ϕ_m, defining solutions to Eqs.(1).

The stability of the system is determined by the 2N values of the complex constant λ. If any one of these has a positive real part then the motion following an initial disturbance diverges with increasing time and the system is unstable. If the real part of λ is negative, the system is said to be stable.

In Reference 1, a method was developed whereby, for a given system, all the characteristic values for λ may be calculated. A computer program was developed to implement the method of Reference 1 for the case of three degrees of freedom (N=3).

The specific objectives of the study described herein were to develop the means for determining, for a given λ, the relative contribution of each generalized coordinate ζ_m to the motion. Thus for each characteristic value of λ it is required to calculate the corresponding characteristic function $\phi_m(z)$ which appears in Eq.(2).

In formulating the scheme for obtaining characteristic functions it has been found that the use of matrix algebra simplifies the representation of the equations. Therefore, Eqs.(1) and (2) are rewritten below in matrix form.

The matrix equation counterpart of Eqs.(1) and (2) are given by

$$\frac{d^2}{dz^2} \{\zeta\} + [a] \frac{d}{dz} \{\zeta\} + [b] \{\zeta\} = \{0\} \tag{3}$$

where:

$$[a] (z + \pi) = [a] (z)$$

$$[b] (z + \pi) = [b] (z)$$

and

$$\{\zeta\} = \{\phi\}(z)\ e^{\lambda z} \tag{4}$$

[a] and [b] are N by N square matrices and $\{\zeta\}$ is a column of N elements.

Since [a] and [b] are periodic they may be expressed in a Fourier series as follows:

$$[a] = \sum_{k=-\infty}^{\infty} [A]^{(k)}\ e^{2ikz} \tag{5}$$

$$[b] = \sum_{k=-\infty}^{\infty} [B]^{(k)}\ e^{2ikz} \tag{6}$$

Since $\{\phi\}(z)$ is also periodic with period π it may also be represented by a complex Fourier series. Thus,

$$\{\phi\}(z) = \sum_{k=-\infty}^{\infty} \{p\}^{(k)}\ e^{2ikz} \tag{7}$$

Substituting Eq.(7) into Eq.(4) yields:

$$\{\zeta\} = \sum_{k=-\infty}^{\infty} \{p\}^{(k)}\ e^{2ikz+\lambda z} \tag{8}$$

Upon differentiating the above expression for $\{\zeta\}$, substituting into Eq.(3), and then grouping and setting equal to zero the coefficients of like powers of e^{2iz}, the following equation is obtained.

$$(\lambda + 2in)^2\{p\}^{(n)} + \sum_{k=-\infty}^{\infty} [(2in - 2ik + \lambda)[A]^{(k)} + [B]^{(k)}]\{p\}^{(n-k)} = \{0\}$$

$$n=0,\pm 1,\pm 2,\ldots \pm\infty \tag{9}$$

Since Eq.(9) is valid for all positive and negative integer values of n, an infinite number of homogeneous matrix equations are theoretically available to arrive at the relative values for:

$$\{p\}^{(0)}\ ,\ \{p\}^{(\pm 1)}\ ,\ \{p\}^{(\pm 2)}\ \ldots\ \{p\}^{(\pm \infty)}$$

In practice it is necessary to deal with a finite number of equations. Therefore Eq.(9) is written for values of

$$n = 0, \pm 1, \pm 2, \pm 3\ \ldots\ N_f$$

where N_F is some finite number such as five or ten. It should be noted, however, that N_f is equal to the number of Fourier components which are solved for in the Fourier representation of $\{\phi\}(z)$. Therefore N_f would probably be no greater than half the number of Fourier components calculated for the arrays of periodically varying functions [a] and [b] which occur in Eqs.(3), (5), and (6).

Since a homogeneous set of $(N \times N_f)$ equations is being dealt with, one of the unknowns is arbitrary. Generally speaking, the elements in $\{p\}^{(0)}$ would be non-zero. Therefore, one of these elements may be arbitrarily set equal to 1.

Unfortunately, while Eq.(9) is a valid and concise equation governing the relative contributions of the generalized coordinates to the total motion of the system, a problem does arise in programming this equation. The zero and negative values of n and k occurring in Eq.(9) cause difficulty when programming in Fortran. This difficulty also had to be overcome in Reference 1. Since the program being described in this report must be compatible with that developed in Reference 1, some further manipulations of Eq.(9) are required in order to arrive at equations suitable for programming in Fortran.

Firstly, Eq.(9) may be rewritten as follows where now the governing equation consists only of a finite number of terms.

$$\left.\begin{aligned}
&[(\lambda + 2in)^2[I] + (\lambda + 2in)[A]^{(0)} + [B]^{(0)}]\{p\}^{(n)} \\
&+ \sum_{k=1}^{N_f+n} [(\lambda + 2in - 2ik)[A]^{(k)} + [B]^{(k)}]\{p\}^{(n-k)} \\
&+ \sum_{k=1}^{N_f-n} [(\lambda + 2in + 2ik)[A]^{(-k)} + [B]^{(-k)}]\{p\}^{(n+k)} = \{0\} \\
&\qquad\qquad\qquad\qquad n=0, \pm 1, \pm 2, \ldots \pm N_f
\end{aligned}\right\} \quad (10)$$

Now, since the periodically varying functions [a] and [b] occurring in Eq.(3) are real matrices (Reference 1) it can be shown from Eqs.(5) and (6) that:

$$\left.\begin{aligned} [A]^{(-k)} &= \overline{[A]^{(k)}} \\ \text{and} \qquad [B]^{(-k)} &= \overline{[B]^{(k)}} \end{aligned}\right\} \qquad (11)$$

where the bar indicates a complex conjugate.

The difficulty of having n assume zero and negative integer values is removed by letting

$$n = m - N_f - 1 \qquad (12)$$

and defining $(2N_f + 1)$ vectors (each containing N elements) as follows:

$$\left.\begin{aligned} \{q\}^{(1)} &= \{p\}^{(-N_f)} \\ \{q\}^{(2)} &= \{p\}^{(-N_f+1)} \\ &\vdots \\ \{q\}^{(N_f)} &= \{p\}^{(-1)} \\ \{q\}^{(N_f+1)} &= \{p\}^{(0)} \\ &\vdots \\ \{q\}^{(2N_f+1)} &= \{p\}^{(N_f)} \end{aligned}\right\} \quad \text{or } \{p\}^{(n)} = \{q\}^{(m)} \qquad (13)$$

Finally, after making the definitions:

$$\left.\begin{array}{ll} [A]^{(1)} = [A]^{(0)} & [B]^{(1)} = [B]^{(0)} \\ [A]^{(2)} = [A]^{(1)} & [B]^{(2)} = [B]^{(1)} \\ [A]^{(2N_f+1)} = [A]^{(2N_f)} & [B]^{(2N_f+1)} = [B]^{(2N_f)} \end{array}\right\} \quad (14)$$

and then utilizing Eqs.(11), (12), (13), and (14) in Eq.(10), the following equation results.

$$\sum_{k=1}^{m-1} [(\lambda+2i(k-N_f-1))[A]^{(m-k+1)} + [B]^{(m-k+1)}]\{q\}^{(k)}$$

$$+ [(\lambda+2i(m-N_f-1))^2[I] + (\lambda+2i(m-N_f-1))[A]^{(1)} + [B]^{(1)}]\{q\}^{(m)}$$

$$+ \sum_{k=m+1}^{2N_f+1} [(\lambda+2i(k-N_f-1)) \overline{[A]^{(k-m+1)}} + \overline{[B]^{(k-m+1)}}]\{q\}^{(k)} = \{0\}$$

$$m=1,2,3,\ldots 2N_f+1 \quad (15)$$

The quantities $[A]^{(j)}$ and $[B]^{(j)}$ correspond exactly to variables which are defined and may be computed in the computer program developed in Reference 1.

Eq.(15) may be written as one single set of homogeneous equations as follows:

$$
\begin{bmatrix}
\begin{bmatrix} R_{1,1} \end{bmatrix} & \begin{bmatrix} R_{1,2} \end{bmatrix} & \cdots & \begin{bmatrix} R_{1,2N_f+1} \end{bmatrix} \\
\begin{bmatrix} R_{2,1} \end{bmatrix} & \begin{bmatrix} R_{2,2} \end{bmatrix} & \cdots & \begin{bmatrix} R_{2,2N_f+1} \end{bmatrix} \\
\vdots & \vdots & & \vdots \\
\begin{bmatrix} R_{2N_f+1,1} \end{bmatrix} & \begin{bmatrix} R_{2N_f+1,2} \end{bmatrix} & \cdots & \begin{bmatrix} R_{2N_f+1,2N_f+1} \end{bmatrix}
\end{bmatrix}
\begin{Bmatrix} \{q\}^{(1)} \\ \{q\}^{(2)} \\ \vdots \\ \{q\}^{(2N_f+1)} \end{Bmatrix}
= \begin{Bmatrix} 0 \end{Bmatrix} \tag{16}
$$

where:

$$
\begin{aligned}
\begin{bmatrix} R_{m,k} \end{bmatrix} &= \begin{bmatrix} (\lambda+2i(k-N_f-1))[A]^{(m-k+1)} + [B]^{(m-k+1)} \end{bmatrix}_{k<m} \\
&= \begin{bmatrix} (\lambda+2i(m-N_f-1))^2[I] + (\lambda+2i(m-N_f-1))[A]^{(1)} + [B]^{(1)} \end{bmatrix}_{k=m} \\
&= \begin{bmatrix} (\lambda+2i(k-N_f-1))\overline{[A]^{(k-m+1)}} + \overline{[B]^{(k-m+1)}} \end{bmatrix}_{k>m}
\end{aligned} \tag{17}
$$

Using the simplest possible shorthand notation, Eq.(16) may be expressed by:

$$[T]\{x\} = \{0\} \tag{18}$$

where:

$$\{x\} = \begin{Bmatrix} \{q\}^{(1)} \\ \{q\}^{(2)} \\ \vdots \\ \{q\}^{(2N_f+1)} \end{Bmatrix} \text{ is a column of } N(2N_f+1) \text{ elements}$$

and [T] is the large $N(2N_f+1)$ by $N(2N_f+1)$ array constructed from $[R_{m,k}]$ submatrices.

Given a characteristic value λ, the $[A]$'s and $[B]$'s, and the number N_f of Fourier components desired in the representation of the characteristic function, it should now be clear how the matrix $[T]$ is constructed. In order to solve for the characteristic functions, it is necessary to fix one of the elements of $\{x\}$ and then solve for the remaining elements.

Consider for example, the case of a three degree of freedom system for which N=3. If the characteristic functions are normalized with respect to the zeroth order Fourier component of the second generalized coordinate, then referring to Eq.(8) the second of the three elements in $\{p\}^{(0)}$ is to be set equal to 1.

According to Eq.(13),

$$\{p\}^{(0)} = \{q\}^{(N_f+1)} \quad . \tag{19}$$

Thus, if the N_f is say 5, then the second element in $\{q\}^{(6)}$, or equivalently the $17^{\underline{th}}$ element in the vector $\{x\}$ is to be set equal to 1.

In the most general case consider the matrix equation (18) to be

$$\begin{bmatrix} \begin{bmatrix} T_1 \end{bmatrix} & \begin{bmatrix} C_1 \end{bmatrix} & \begin{bmatrix} T_2 \end{bmatrix} \\ \begin{bmatrix} r_1 \end{bmatrix} & \begin{bmatrix} c_0 \end{bmatrix} & \begin{bmatrix} r_2 \end{bmatrix} \\ \begin{bmatrix} T_3 \end{bmatrix} & \begin{bmatrix} C_2 \end{bmatrix} & \begin{bmatrix} T_4 \end{bmatrix} \end{bmatrix} \begin{Bmatrix} x_1 \\ x_2 \\ \vdots \\ x_{NN_f+I} \\ \vdots \\ x_{N(2N_f+1)} \end{Bmatrix} = \begin{Bmatrix} \\ 0 \\ \\ \end{Bmatrix} \tag{20}$$

where: (NN_f+I) indicates the element which is to be equal to 1 in the $\{x\}$ column,

I indicates a normalization with respect to the $I^{\underline{th}}$ generalized coordinate,

c_0 is the element in the $(NN_f+I)^{\underline{th}}$ row and column of the $[T]$ array.

It can then be easily shown that the matrix equation required to solve for the remaining x's is given by

$$\begin{bmatrix} [T_1] & [T_2] \\ [T_3] & [T_4] \end{bmatrix} \begin{Bmatrix} x_1 \\ \vdots \\ x_{NN_f+I-1} \\ x_{NN_f+I+1} \\ \vdots \\ x_{N(N_f+1)} \end{Bmatrix} = - \begin{Bmatrix} [C_1] \\ [C_2] \end{Bmatrix} \tag{21}$$

Note that Eq.(21) differs from Eq.(20) in that the row and column of [T] which contain the element T_{NN_f+I,NN_f+I} have been removed and a right-hand side of the equation has been formed from the negative of the removed column minus that element. Eq.(21) may be solved in a straightforward manner for the x's or equivalently by the characteristic functions

$$\{p\}^{(k)} \qquad \text{where:} \quad k = 0, \pm 1, \pm 2 \ldots \pm N_f \quad .$$

DESCRIPTION OF COMPUTER PROGRAM

The computer program developed for this study was coded directly from the formulation described above. Since this program was intended to be used in conjunction with the program developed in Reference 1, the number of degrees of freedom treated in this study was restricted to three (N=3) as in the characteristic value program.

The main inputs to this program are: the characteristic value (λ); the number of Fourier coefficients desired (N_f); and the Fourier components of the periodically varying coefficients in the original equations of motion ([A]'s and [B]'s). The latter quantities are computed in a special subroutine which was also used in the characteristic value program of Reference 1. The λ's, of course, are the characteristic values which are the output of the Reference 1 program. The quantity N_f is arbitrary except for the practical considerations that it is limited by the number of Fourier components (the [A]'s and [B]'s available) and the computer storage capacity.

Also provided as an input quantity is an index which indicates how the characteristic functions are to be normalized.

The input quantities are manipulated by programming logic that closely follows the formulation described in the previous section of this report in order to construct the large [T] array (Eq.20). The operations of removing the appropriate row and column are performed and the resulting set of linear algebraic equations are solved for the characteristic functions.

The functions are printed out with appropriate comments for identification purposes.

A flow diagram paralleling the above description and the previously described formulation is provided in Figure 1, and a listing of the associated computer program is given in Appendix B.

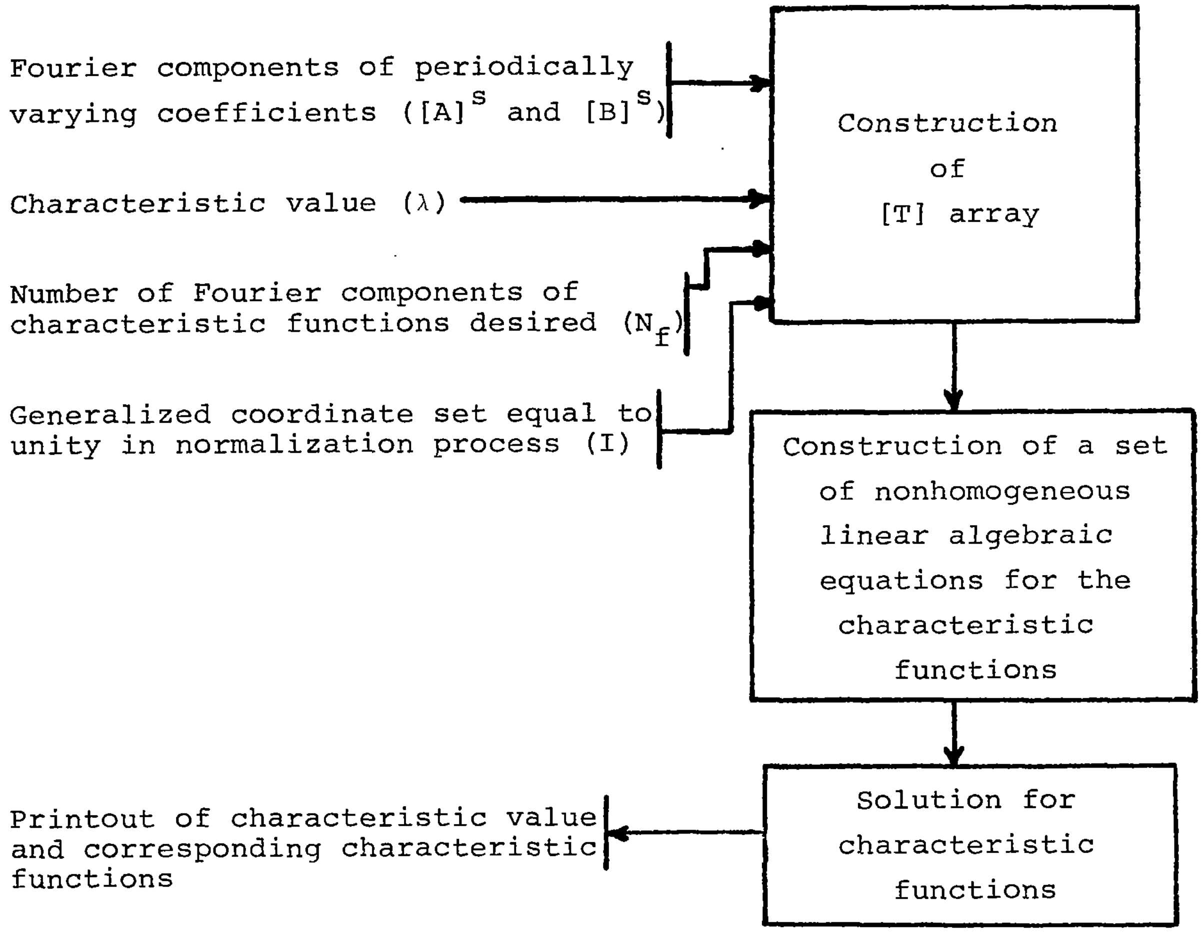

Figure 1. Basic flow diagram for characteristic modal functions

APPLICATION OF THE METHOD

The computer program developed in Reference 1 for predicting the characteristic values of a coupled set of linear, second-order differential equations with periodically varying coefficients was used in conjunction with the computer program that was developed herein to determine the stability characteristics of a model helicopter blade in hover and forward flight. The model helicopter rotor blade for which experimental flutter results are presented in References 3 and 4 had a single blade with a radius of four feet and with a flapping hinge through the axis of rotation. The blade had a constant chord of 3.5 inches and a root cutout of 6 inches. The blade was relatively rigid in torsion, but the control system was made flexible, so the primary contributions to the blade motions derived from rigid-body feathering, flapping motions and deflections in the first flapwise bending mode.

The model configuration that was chosen for investigation had a ratio of the nonrotating first uncoupled flapwise bending frequency, $\bar{\omega}_{\phi_1}$ to feathering frequency $\bar{\omega}_{\theta_0}$ of 0.63 and a chordwise center of mass at the 42.5% chord aft of the leading edge.

Determination of Model Blade Frequencies

Since the experimental flutter data presented in Reference 3 was not supported by either measured or computed uncoupled and coupled mode shapes and frequencies for the model blade (needed for the present study), they had to be determined. This was accomplished by using a refined rotor blade vibration analysis, Reference 4, in conjunction with the blade data reported for the blade in Reference 3. The results of these calculations yielded an uncoupled nonrotating first bending frequency $\bar{\omega}_{\phi_1}$ of 75 rad/sec. The root feathering spring was then adjusted so that the first nonrotating feathering-torsion mode had a frequency $\bar{\omega}_{\theta_0}$ of 119.1 rad/sec in order that a frequency ratio of $\bar{\omega}_{\phi_1}/\bar{\omega}_{\theta_0}$ of 0.63 was obtained. Using the stiffness of the feathering spring that was determined by this method, the coupled nonrotating and rotating vibration modes were then computed at various rotational speeds. A frequency diagram presenting the results of these calculations is presented in Figure 2, and the generalized components of the various coupled modes at the blade tip are given in Table I for a few of the rotational speeds at which calculations were conducted.

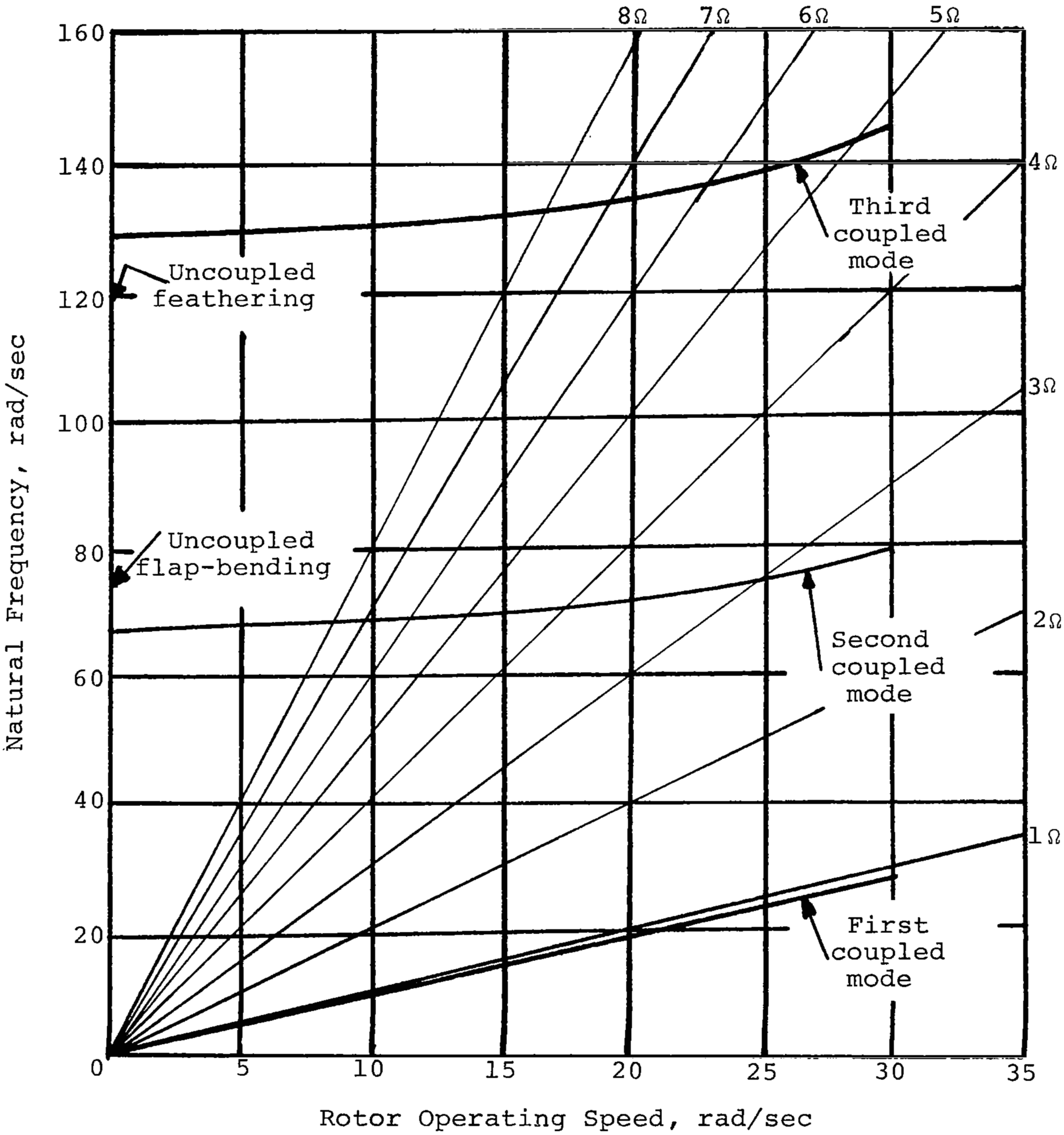

Figure 2. Coupled natural frequencies versus rotor speed

TABLE I

GENERALIZED COMPONENTS OF COUPLED MODES

Coupled Mode	Component	Rotational Speed (rad/sec)			
		0	15	20	24.5
First	Flap-Bending	-	1.0	1.0	1.0
	Feathering-Torsion	-	0.0738	0.13	0.193
Second	Flap-Bending	1.0	1.0	1.0	1.0
	Feathering-Torsion	-1.81	-1.93	-2.01	-2.1
Third	Flap-Bending	1.0	1.0	1.0	1.0
	Feathering-Torsion	9.01	8.77	8.62	8.46

It should be noted that since the generalized components have been normalized by the flap-bending deflection, the feathering-torsion deflections have units of radian per unit of tip bending deflection.

The first coupled mode is primarily a flapping mode and the second and third coupled modes are primarily highly coupled bending-feathering modes with the third mode having a significantly larger relative feathering motion than the second mode.

Theoretical Determination of Rotor Stability Characteristics

Using the first three coupled modes of the rotor blade as generalized coordinates, the characteristic values and characteristic functions were determined for various rotor speeds at advance ratios of 0, 0.1, 0.2, and 0.3. The characteristic values and characteristic functions that were determined are presented in Table II and Table III, respectively. In order to determine, at a given advance ratio, the rotational speed at which the rotor is neutrally stable, the real part of the characteristic value is plotted versus rotational speed and the rotation speed at which the real part vanishes is the critical speed. Since the real part of the characteristic value can be considered to be a measure of the system damping (growth or decay rate) the effect of structural damping can be easily determined in much the same manner as it is accomplished for fixed wing aircraft through plots of velocity versus damping.

TABLE II

CHARACTERISTIC VALUES OF MODE WHICH BECOMES UNSTABLE

μ	Rotational Speed (rad/sec)				
	10	15	18	20	23
0	-1.74±64.0i	-1.390±60.3i	-	2.32±53.9i	8.0±50.2i
0.1	-	-1.190±56.3i	-0.612±56.3i	2.50±54.1i	-
0.2	-	-0.980±60.2i	1.360±56.4i	4.04±53.5i	-
0.3	-	-0.525±59.7i	2.430±56.4i	4.20±55.0i	-

As noted in Table III, the characteristic function at each condition has contributions from all the coupled modes which have been normalized by the value of the second coupled mode. The relative magnitudes of the various modes in the characteristic function give an indication of the primary degrees of freedom that are present. For example, for a rotor speed of 18 rad/sec and an advance ratio of 0.10, the first coupled mode has a relative amplitude of approximately 0.78, the second 1.0, and the third, 0.026. These relative orders of magnitude indicate that the rotor oscillatory motion at these conditions is comprised primarily of motions in the first and second coupled modes. When these relative amplitudes of motion for the various modes are applied to the different coupled degrees of freedom to determine the relative amplitudes of the primary motions (flap, bending, feathering), the results indicate that the mode of instability is of a highly coupled bending-feathering type.

Comparison of Theoretical and Experimental Results

In order to put the theoretical results in a form in which they could be compared with the experimental results of Reference 3, the characteristic values presented in Table II were plotted versus rotor speed to determine the rotational speed at which the rotor was neutrally stable. With the critical rotor speed determined, the nondimensional flutter parameters were calculated and are presented in Table IV. When the theoretical results presented in Table IV were compared with the experimental data, it was noted

TABLE III

CHARACTERISTIC FUNCTIONS OF MODE WHICH BECOMES UNSTABLE

μ	Mode	Rotational Speed (rad/sec)				
		10	15	18	20	23
0	1	-0.161±0.0298i	-0.4290±0.005i	-	-1.020±0.33i	-1.2400±1.13i
	2	1.00	1.00		1.00	1.00
	3	-0.008±0.0036i	-0.0170±0.007i		-0.027±0.011i	-0.0326±0.015i
0.10	1	-	-0.4340±0.001i	-0.763±0.145i	-0.990±0.346i	-
	2		1.00	1.00	1.00	
	3		-0.0170±0.007i	-0.024±0.010i	-0.027±0.012i	
0.20	1	-	-0.4490±0.020i	-0.760±0.205i	-0.980±0.489i	-
	2		1.00	1.00	1.00	
	3		-0.0185±0.008i	-0.025±0.0115i	-0.027±0.010i	
0.30	1	-	-0.4780±0.048i	-0.756±0.294i	-0.865±0.48i	-
	2		1.00	1.00	1.00	
	3		-0.0210±0.009i	-0.027±0.014i	-0.029±0.017	

TABLE IV

PREDICTED ROTOR PARAMETERS AT FLUTTER BOUNDARY

Damping g	Rotor Parameters	Advance Ratio μ			
		0	0.10	0.20	0.30
0	Ω	18.60	18.60	16.90	16.30
	$\bar{\omega}_{\theta_0}/\Omega$	6.44	6.44	7.09	7.35
0.03	Ω	19.20	19.15	17.60	17.00
	$\bar{\omega}_{\theta_0}/\Omega$	6.24	6.25	6.80	7.04

that the theoretical results were extremely conservative in that the rotor speed at which the instability boundary was predicted was only 59% of that which was measured. While the support data that was presented for the model in Reference 3 did not record the uncoupled frequencies associated with the first bending mode and the first feathering modes, it was determined from the data presented in Reference 5 for the same model, that the first nonrotating uncoupled flapwise bending mode had a frequency $\bar{\omega}_{\theta_1}$ of 83 rad/sec and the first nonrotating uncoupled feathering-torsion mode had a frequency of $\bar{\omega}_{\theta_0}$ of 132 rad/sec. Since the corresponding frequencies that were calculated during this program using the reported mass-elastic data for the model were 75 rad/sec and 119.1 rad/sec, respectively, a direct comparison of the theoretical and experimental data was not deemed to be valid. However, on the basis of a straight-line interpolation of experimental flutter data for $\bar{\omega}_{\theta_0}$ of 108 rad/sec and 132 rad/sec presented in Reference 5 for the subject model in the hover condition, it was estimated that if the experimental frequencies reported for the model had been used in the theoretical prediction, the theoretically determined rotor speed would be approximately 1.24 times those predicted. It was believed, therefore, that if the predicted results were corrected by this factor a direct comparison could be made with the experimental results presented in Reference 3. The theoretical results presented in Table IV were thus adjusted to account for this difference and are presented in Table V. The results presented in Table V are

TABLE V

CORRECTED ROTOR PARAMETERS AT FLUTTER BOUNDARY

Damping g	Rotor Parameters	Advance Ratio μ			
		0	0.10	0.20	0.30
0	Ω	23.05	23.05	20.95	20.20
	$\bar{\omega}_{\theta_0}/\Omega$	5.19	5.19	5.72	5.93
0.03	Ω	23.80	23.75	21.82	21.06
	$\bar{\omega}_{\theta_0}/\Omega$	5.03	5.04	5.49	5.68

compared with the experimental results in Figure 3.

As can be seen from the results that are plotted, the corrected theoretical results while being about 35% conservative, indicate the same trend with advance ratio as do the experimental data. The effect of a normal amount of structural damping decreased the degree of conservation by only 4% indicating that structural damping was not the reason for the difference between the predicted and experimental results. It is believed that possible reasons for the difference between the theoretical and experimental results are tip losses and a significant reduction in the lift curve slope from the theoretical value of 6.28 due to the relatively low Reynolds number at which the model tests were conducted. To determine if this could possibly be the reason for the discrepancy between the theoretical and experimental results, it was assumed that the aerodynamic forces in hover were reduced by 50% due to these factors. The results of these calculations indicated that the $\bar{\omega}_{\theta_0}/\Omega$ at hover would be 3.6 or approximately the same value as determined by the experimental data. While it is not believed that the aerodynamic forces would be reduced by this amount due to Reynolds number and tip loss effects, the results do indicate that the theoretical and experimental results would probably be in much closer agreement if the effective lift curve slope associated with the model was used in the theoretical analysis.

Since the performance characteristics were not measured for the flutter model, an evaluation of the effective lift curve slope could

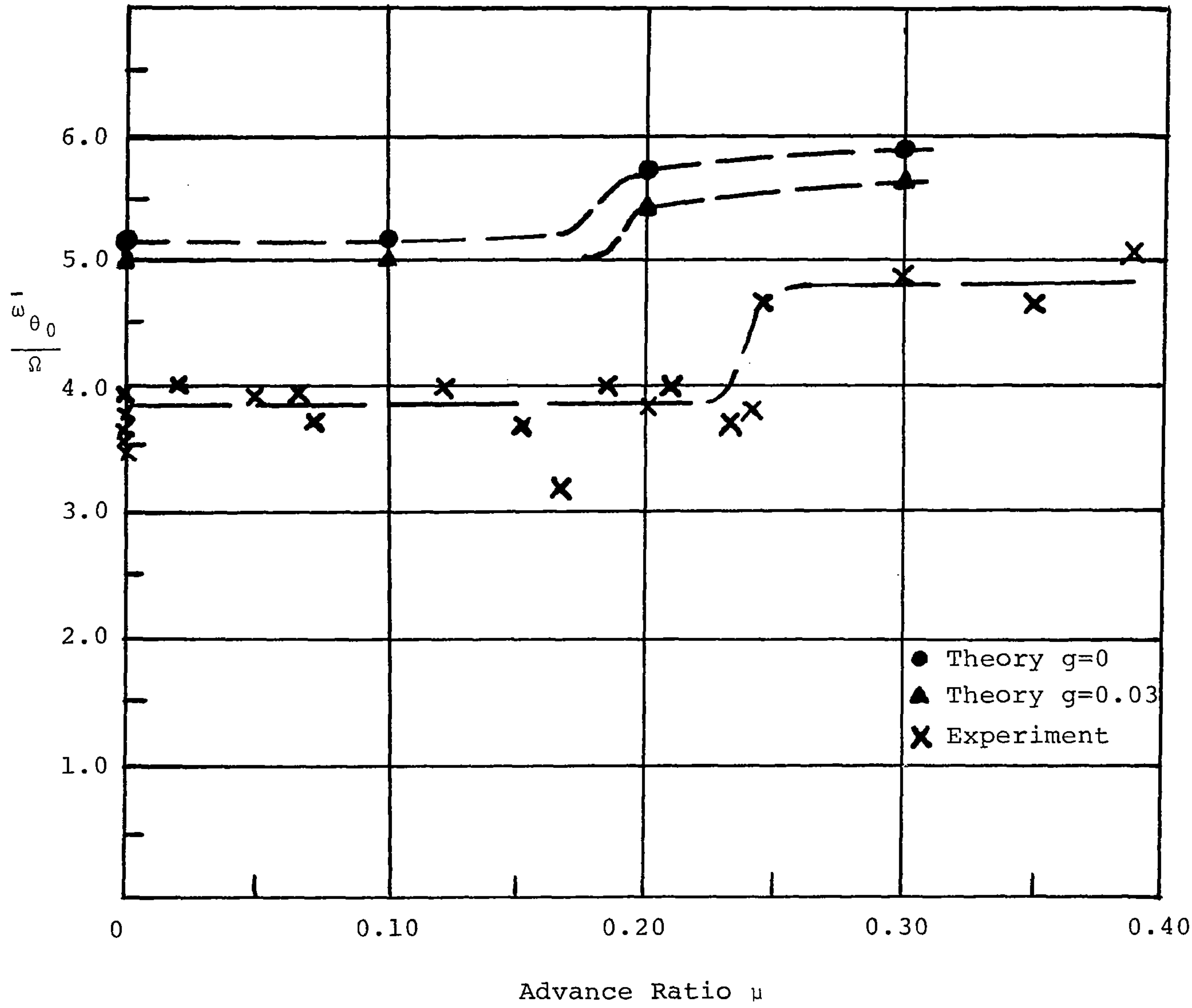

Figure 3. Comparison of predicted and experimental characteristic values

not be undertaken. For the obvious benefits that could be derived, it is suggested that it might be invaluable to measure, during all scale model dynamic tests, the performance characteristics of the rotor so that an estimate of the effective lift curve slope can be made for use in theoretical analyses correlating the experimental results.

CONCLUDING REMARKS

The results of the research program conducted herein indicate that a reliable method of predicting the characteristic functions associated with rotor instabilities has been developed once the characteristic values of the instability have been determined by means of the analysis procedure previously developed.

During the performance of the effort associated with this research investigation, it again became apparent that there is a definite need for a reliable and well-documented set of experimental flutter data. It is believed that the need for this data is urgent as, due to the rapid growth in rotor technology, more instances of unexpected and unexplained cases of rotor instability will probably occur more frequently. In order to investigate the reason for the instabilities and determine means for corrective action, there is a need for a proven method of analysis. While it is believed that a reliable analysis has been developed, it or any other analysis procedure cannot be assumed to be quantitatively reliable until it has been proven to be so by comparison with a well-documented set of experimental data.

APPENDIX A

Listing of Computer Program for Determining Characteristic Eigenvalues

(Update of Program Listed in Reference 1)

```
      PROGRAM NASA4(INPUT,OUTPUT,ABS,TAPE5=INPUT,TAPE6=OUTPUT,
     1 TAPE8=ABS)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION AR(125),AI(125),BR(125),BI(125),CR(250),CI(250),DR(250)
      DIMENSION DI(250),FR(250),FI(250),ER(250),EI(250)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      READ (5,9876) NN,NF,LI,LJ,NW1,NSTOP
 9876 FORMAT (10I5)
      LK=NF
      DO 11 IJK=1,125
      BR(IJK)=0.
      BI(IJK)=0.
      AI(IJK) = 0.
   11   AR(IJK)=0.0
      DO 111 II=1,NN
      AZ=1.
      IF (NW1.EQ.1) GO TO 150
      READ (8) I,J,K,ARR,AIR,BRR,BIR
      GO TO 160
  150 READ (5,9875) I,J,K,ARR,AIR,BRR,BIR
 9875 FORMAT (3I3,4E14.7)
  160 IJK= (I*LI-4+J)*NF+K
      IF (K.GT.5) AZ=0.
      AR(IJK)=ARR*AZ
      AI(IJK) = AIR*AZ
      BI(IJK)=BIR*AZ
       BR(IJK)= BRR*AZ
      WRITE (6,100) I,J,K,AR(IJK),AI(IJK),BR(IJK),BI(IJK)
  100 FORMAT (3(2X,I2),4X,4(3X,E14.7))
  111 CONTINUE
      CALL        Q (IOUT,NF,LK,AR,AI,BR,BI,NR,NA,PDQ)
      STOP
      END
      SUBROUTINE Q (IOUT,NF,LK,AR,AI,BR,BI,NR,NA,PDQ)
C     IMPLICIT REAL*8(A-H,O-Z)
      INTEGER UT
      REAL MU
      DIMENSION L2(5)
      DIMENSION ER(350),EI(350),FR(350),FI(350),FFR(150),FFI(150)
      DIMENSION DELT(150),ALPHA(150),BETA(150),GAMMA(150)
      DIMENSION GR(5000),GI(5000)
      DIMENSION CI(250),DR(250),DI(250),DUM(810)
      DIMENSION YR9(9),YI9(9),YR12(12),YI12(12),BIGA(9),BIGAH(13),YY(13)
      DIMENSION AK9(9),AK12(13),DYR9(11),DYI9(11),DYR12(12),DYI12(12)
      DIMENSION SIGMA(7)
      DIMENSION XI9(10),ET9(10),XI12(13),ET12(13),XI6(7),ETA6(7)
      DIMENSION R12TR(13),R12TI(13)
      DIMENSION AUM(1,1),AY(15),AR(125),AI(125),BR(125),BI(125),CR(250)
      DIMENSION PSI9(9),R06TR(6),R06TI(6),NDR9(9),NDI9(9),NDR12(12)
      DIMENSION NDI12(12),DETR(12),DETI(12),PIRNEW(12),PIROLD(12)
      DIMENSION PIINEW(12),PIIOLD(12),AUMR9(8,14),AUMI9(8,14)
      DIMENSION AUMR12(11,14),AUMI12(11,14)
```

```
      DIMENSION ROLY(9),PSI12(13),ETA9(10),ETA12(13)
      DIMENSION RR(20),RRI(20),ROOTI(20),ROOTR( 20),Z(20),Y(20),POLY(20)
      DIMENSION XR(12),XI(12)
      DIMENSION CCH(150),CC(150),PSI6(7)
      DIMENSION RR12(13),RRI12(13)
      DIMENSION A9(9),B9(9),C9(9),AH(13),BH(13),CH(13)
      DIMENSION ICHG(15)
      DIMENSION U(12)
      EQUIVALENCE(PSI9(1),DUM(50)),(R06TR(1),DUM(60)),(R06TI(1),DUM(70))
      EQUIVALENCE(NDR9(1),DUM(80)),(NDI9(1),DUM(90)),(NDR12(1),DUM(100))
      EQUIVALENCE (DYR9(1),DUM(115)),(DYI9(1),DUM(130)),(RR(1),DUM(145))
      EQUIVALENCE(DYR12(1),DUM(166)),(NDI12(1),DUM(180))
      EQUIVALENCE(DETR(1),DUM(193)),(DETI(1),DUM(206))
      EQUIVALENCE(PIRNEW(1),DUM(220)),(PIROLD(1),DUM(233))
      EQUIVALENCE(PIINEW(1),DUM(246)),(PIIOLD(1),DUM(260))
      EQUIVALENCE  (AUMR9(1,1),DUM(273)),(AUMI9(1,1),DUM(386))
      EQUIVALENCE(AUMR12(1,1),DUM(500)),(AUMI12(1,1),DUM(655))
      EQUIVALENCE(AK9(1),GR(1)),(AK12(1),GR(10)),(SIGMA(1),GR(25))
      EQUIVALENCE(GR(35),XI9(1)),(ET9(1),GR(45)),(XI12(1),GR(60))
      DSQRT(V)=SQRT(V)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(V)=ABS(V)
      DATAN(V)=ATAN(V)
      DCOS(V)=COS(V)
      DEXP(V)=EXP(V)
      DLOG(V)=ALOG(V)
      DLOG10(V)=ALOG10(V)
      DSIN(V)=SIN(V)
      DCOSH(V)=COSH(V)
      DSINH(V)=SINH(V)
C    FORMULA FOR GR,GI,ALPHA,BETA,GAMMA,DELT    IS  3(ND+1)**2
C     MAINFRAME PROGRAM FOR THE NASA FLUTTER
      DO 1 I=1,5000
      GR(I)=0.
    1 GI(I)=0.
      DO 2 I=1,9
      BIGA(I)=0.
      YR9(I)=0.
      YI9(I)=0.
      DYR9(I)=0.
      DYI9(I)=0.
      A9(I)=0.
      B9(I)=0.
    2 C9(I)=0.
      DO 3 I=1,15
      BIGAH(I)=0.
      AH(I)=0.
      BH(I)=0.
      CH(I)=0.
      RR12(I)=0.
    3 RRI12(I)=0.
      DO 4 I=1,12
      YR12(I)=0.
      YI12(I)=0.
      DYR12(I)=0.
      DYI12(I)=0.
      XR(I)=0.
      XI(I)=0.
    4 U(I)=0.
      DO 5 I=1,15
      AY(I)=0.
    5 ICHG(I)=0.
```

```
      DO 6 I=1,150
      CCH(I)=0.
    6 CC(I)=0.
      DO 7 I=1,7
    7 PSI6(I)=0.
      NUMMA=1
      INP = 0
      NO= 9
C     LK = NF   A SPECIIFIED INTEGER    USUALLY LESS THAN 3
      UT=6
      IOUT=UT
      NIN=5
      NP=6
      NOUT=NP
      LI=3
      LJ=3
      LK=NF
      N=NF
      K=1
      LM = 3
      LJ1 = LJ+1
      DO 1575 MM=1,810
      DUM(MM) = 0
 1575 CONTINUE
      OUT=-1
      PI=3.1415926
      DELTA=.0001/PI
      EPS=1.D-10
      DELTAX=1.D-5
      CALL S1B(OUTPUT,CR,CI,INPUT,AR,AR,AI,AI,BR,BI,LI,LJ,LK,NF,NP)
      CALL S1B(OUT,DR,DI,IN,AR,BR,AI,BI,DUM,DUM,LI,LJ,LK,NF,NP)
      CALL S1C(AR,AI,CR,CI,CR,CI,DR,DI,ER,EI,LI,LJ,LK,NF)
      CALL S1C(BR,BI,CR,CI,DR,DI,DUM,DUM,FR,FI,LI,LJ,LK,NF)
      CALL S3(1,PSI9,ETA9,YR9,YI9,K9,IN,DELTA,CR,DR,NF,9,NP,NIMAG)
      L=NIMAG
      CALL S3(1,PSI12,ETA12,YR12,YI12,K12,IN,DELTA,ER,FR,NF,12,NP,NIMAG)
      CALL D(FFR,FFI,GR,GI,ALPHA,BETA,GAMMA,DELT,CR,CI,DR,DI,AUMR9,AUMI9
     1,AUMR12,AUMI12,DYR9,DYI9,PIROLD,PIRNEW,YR9,YI9,NDR9,NDI9,9,K9,L,N,
     2LI,LJ,NF,EPS,DELTAX)
      CALL D(FFR,FFI,GR,GI,ALPHA,BETA,GAMMA,DELT,ER,EI,FR,FI,AUMR9,AUMI9
     1,AUMR12,AUMI12,DYR12,DYI12,PIIOLD,PIINEW,YR12,YI12,NDR12,NDI12,
     212,K12,NIMAG,N,LI,LJ,NF,EPS,DELTAX)
      WRITE (NP,1180) DYR9,DYI9,DYR12,DYI12
      CALL S4(CC,BIGA,A9,B9,C9,XR,XI,U,PSI9,ETA9,9,K9,PI,DYR9,DYI9,YR9,
     1 YI9)
      WRITE (NP,1180) BIGA
      CALL SIMQ(CC,9,BIGA)
      WRITE (NP,1180) BIGA
      OUT=1
      CALL S4(CCH,BIGAH,AH,BH,CH,XR ,XI ,U,PSI12,ETA12,12,K12,PI,
     1 DYR12,DYI12,YR12,YI12)
      WRITE (NP,1180) BIGAH
      CALL SIMQ(CCH,12,BIGAH)
      WRITE (NP,1180) BIGAH
      WRITE (NP,1180) AH,BH,CH
      WRITE (NP,1180) A9,B9,C9
 1180 FORMAT (*1*/(G15,7))
      CALL S5A(OUT,AK12,IN,K12, 6,BIGAH,BH,AH,CH,IERR)
 1171 AY(1) = AK12(12)
      AY(2) = AK12(11)
      AY(3) = AK12(10)
      AY(4) = AK12(9)
```

```
      AY(5) = AK12(8)
      AY(6) = AK12(7)
      AY(7) = AK12(6)
      AY(8) = AK12(5)
      AY(9) = AK12(4)
      AY(10) = AK12(3)
      AY(11) = AK12(2)
      AY(12) = AK12(1)
      AY(13) = 1,
      NUM=12
      CALL DPRQD (AY,13,R12TR,R12TI,YY,NUM,IERR)
      WRITE (6,1180) NUM,IERR
 1169 CALL TEA(12,6,R12TR,R12TI,XI12,ET12)
      CALL S5A(OUT,AK9,IN,K9,4,BIGA,B9,A9,C9,IERR)
 1177 AY(1) = AK9(9)
      AY(2) = AK9(8)
      AY(3) = AK9(7)
      AY(4) = AK9(6)
      AY(5) = AK9(5)
      AY(6) = AK9(4)
      AY(7) = AK9(3)
      AY(8) = AK9(2)
         AY(9) = AK9(1)
      AY(10) = 1,
      NUM=9
      CALL DPRQD (AY,10,RR,RRI,YY,NUM,IERR)
      WRITE (6,1180) NUM,IERR
 1175 CALL TEA(9,6,RR,RRI,XI9,ET9)
      CALL S6(OUT,SIGMA,IN,AK9,AK12,NP)
 1167  Z(7) = 1
      Z(6) = SIGMA(1)
      Z(5) = SIGMA(2)
      Z(4) = SIGMA(3)
      Z(3) = SIGMA(4)
      Z(2) = SIGMA(5)
      Z(1) = SIGMA(6)
      WRITE (NP,1180) SIGMA
      NUM=6
      CALL DPRQD (Z,7,R06TR,R06TI,YY,NUM,IERR)
      WRITE (6,1180) NUM,IERR
      CALL TEA(6,6,R06TR,R06TI,XI6,ETA6)
      CALL PAT(AUMR9,AUMR12,PSI9,PSI12,XI9,XI12,XI6,ETA9,ETA12,ET9,ET12
     1,ETA6,   GA,MU,NF,RR,RRI,R12TR,R12TI,R06TR,R06TI,NDR9,NDR12,YR9,YR
     212,PIROLD,PIRNEW,PIIOLD,PIINEW,AUMI9,AUMI12,NDI9,NDI12,YI9,YI12)
      STOP
      END
      SUBROUTINE D(FFR,FFI,GR,GI,AL,BE,GA,DE,CR,CI,DR,DI,AUMR9,AUMI9
     1,AUMR12,AUMI12,DYR9,DYI9,PIROLD,PIRNEW,YR9,YI9,NDR9,NDI9,NO,L,K,N,
     2LI,LJ,NF,EPS,DELTAX)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION AY(150)
      DIMENSION DETR(20),DETI(20),NDI(20),PIRNEW(20),PIROLD(20)
      DIMENSION FFR(1),FFI(1),GR(1),GI(1),AL(1),BE(1),GA(1),DE(1),DR(1),
     1   DI(1),CR(1),CI(1),AUMR9(8,14),AUMI9(8,14),AUMR12(11,14),
     2 AUMI12(11,14),DYR9(1),DYI9(1),YR9(1),YI9(1),NDR9(1),NDI9(1)
      DSQRT(V)=SQRT(V)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(V)=ABS(V)
      DATAN(V)=ATAN(V)
      DCOS(V)=COS(V)
      DEXP(V)=EXP(V)
      DLOG(V)=ALOG(V)
```

```
      DLOG10(V)=ALOG10(V)
      DSIN(V)=SIN(V)
      DCOSH(V)=COSH(V)
      DSINH(V)=SINH(V)
C     SUBROUTINE TO FETCH DETERMINANTS AND CONVERGENCE MONITOR
      OUT=6.
        NO=6
      NIMAG= K
      KEU=K
      K9=L
      IF(NO-9) 2,3,2
    3 NUM=1
      GO TO 4
    2 IF(NO-12) 5,6,5
    5 KEU=3
      RETURN
    6 NUM=2
    4 NO1=NO-1
C       DO COMPLEX ROOTS FIRST   USING COMPLEX CONJUGATE OF THE DETERMINAN
C     DO THE REMAINING REAL ROOTS  SECOND       PART 2
       DO 1145 NP=1,2
        GO TO (1041,1042),NP
 1041  NP1=1
      NP2=NIMAG
      NSTEP= 2
       GO TO 1143
 1042  NP1=NIMAG+1
      NP2= NO-1
      IF (NP2.EQ.NIMAG) GO TO 1145
      NSTEP= 1
 1143 DO 145 MM=NP1,NP2,NSTEP
      IN=0
      H=YR9(MM)
      GH=YI9(MM)
  168 DO 55 ND=5,6
      NO3=6*ND+3
      NN=NO3*NO3
      DO 77 LL=1,NN
      GR(LL)=0.
   77 GI(LL)=0.
      MD=ND
C    NOGO = 4 INDICATES THAT BOTH HAVE CONVERGED
      CALL S2A(OUT,FFR,FFI,GR,GI,AL,BE,GA,DE,PIR,PII,
     1  IN,CR,CI,DR,DI,H,GH,N,ND,LI,LJ,NF,M,NQ,NO,EPS,DELTAX)
 4444 IF(KEU) 11,11,12
   12 WRITE(NQ,1037) ND,H,GH,PIR,PII
 1037  FORMAT(1H0,4H Y( ,I3,4H)=  E20.8,*I*,14H  D REAL            E20.8,
     112H   D IMAG = ,G20.8 /*0*,8G15.7)
   11 DETR(ND)=PIR
      DETI(ND)=PII
      GO TO (919,912),NUM
  919 AUMR9(MM,ND)=PIR
      AUMI9(MM,ND)=PII
      IF(NP.EQ.1)  GO TO 55
      AUMR9(MM+1,ND)= PIR
      AUMI9(MM+1,ND)=-PII
       GO TO 55
  912 AUMR12(MM,ND)=PIR
      AUMI12(MM,ND)=PII
      IF(NP.EQ.1)  GO TO 55
      AUMR12(MM+1,ND)=PIR
      AUMI12(MM+1,ND)=-PII
```

```
   55 CONTINUE
      NOMO=1
      NOCO=1
      DETI(3)=0.
      DETI(4)=0.
      DO 155 ND=7,11
      IF (NSTEP.EQ.1) NOMO=2
      NOGO= NOCO+NOMO
      IF (NOGO.EQ.4) GO TO 1677
      NO3=6*ND+3
       NN=NO3*NO3
      DO 78 LL=1,NN
      GR(LL)=0.
   78 GI(LL)=0
      MD=ND
   91 CALL S2A(OUT,FFR,FFI,GR,GI,AL,BE,GA,DE,PIR,PII,
     1 IN,CR,CI,DR,DI,H,GH,N,ND,LI,LJ,NF,M,NQ,NO,EPS,DELTAX)
      IF(KEU)  13,13,14
   14 WRITE(NQ,1037) ND,H,GH,PIR,PII,NOCO,NOMO
   13 DETR(ND)=PIR
      GO TO(29,32),NUM
   29 AUMR9(MM,ND)=PIR
      GO TO 35
   32 AUMR12(MM,ND)=PIR
  35  NSTART=5
      GO TO (92,93),NOCO
   92 CALL UP(DETR,MD,NSTART,NQ)
      NDR9(MM)=ND
      DETR(4)=PIR
   83 IF (NSTART) 93,141,93
  141 IF(DETR(2)) 93,167,93
  167 DETR(4)=DETR(1)
C    REAL PART HAS CONVERGED
      NOCO=2
   93 NSTART=5
      GO TO (95,155),NOMO
   95 DETI(ND)=PII
      CALL UP(DETI,MD,NSTART,NQ)
      DETI(4)=PII
      NDI9(MM)=ND
      IF(NSTART) 155,1441,155
 1441 IF(DETI(2)) 155,97,155
   97 NOMO=2
      DETI(4)=DETI(1)
  155 CONTINUE
      IF (NOMO.EQ.1) DETI(1)=PII
      IF (NOCO.EQ.1) DETR(1)=PIR
      IF (NSTEP.EQ.1) DETI(1)=0.
      ND=MD
 1677 DYI9(MM)= DETI(4)
      DYR9(MM)=DETR(4)
      PIRNEW(MM)=DETR(3)
      PIROLD(MM)=DETI(3)
      GO TO(49,52),NUM
   49 AUMR9(MM,ND+1)=DYR9(MM)
      AUMI9(MM,ND+1)=DYI9(MM)
      IF(NP.EQ.2)  GO TO 51
      AUMR9(MM+1,ND+1)=DYR9(MM)
      AUMI9(MM+1,ND+1)=-DYI9(MM)
      NDI9(MM+1)=0
      GO TO 51
   52 AUMR12(MM,ND+1)=DYR9(MM)
```

```
      AUMI12(MM,ND+1)=DYI9(MM)
      IF(NP,EQ,1)  GO TO 51
      AUMR12(MM+1,ND+1)=DYR9(MM)
      AUMI12(MM+1,ND+1)=-DYI9(MM)
      NDI9(MM+1)=0
   51 NDI9(MM)=ND
      IF(KEU)145,145,57
   57 WRITE(NQ,6656)DETR(1),DETI(1),H
 6656 FORMAT(1H0,22HPREDICTED CONVERGENCE G20,8,7H REAL    ,G20,8,
     1 7H IMAG,    15H EVALUATED AT      G20,8)
      DETI(4)=0,
  145 CONTINUE
 1145 CONTINUE
      RETURN
      END
      SUBROUTINE PRE(ND,PC,NP)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PC(18)
C        PREDICT THE CONVERGEED VALUES BASED ON THE LAST THREE
C     DETERMINANTS
C     REAL*8 K2K1,K2K3,MU,K1,K2,K3
      REAL K1,K2,K3,K2K1,K2K3,MU
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      DELTA=,00001
      C1 = PC(ND-2)
      C2 = PC(ND-1)
      C3 = PC(ND)
      K1=ND-2
      K2 = ND-1
      K3 = ND
      K2K3= K2/K3
      K2K1=K2/K1
C   WRITTEN FOR THE INFINITE DETERMINANT  SUBROUTINE WITH ASYMPTOTIC
      MU = (C1-C2)/(C2-C3)
      P=,00001
      FP = MU*(1,-K2K3**P)-K2K1**P+1
      A=DABS(FP)/FP
C   THIS GETS NEGATIVE OR POSITIVE ONE
      AM = 1
      AP = 1
      M20=1
      M100=100
      M1 = 1
   13 KKKK=0
      DO 1 M = M20,M100,M1
      AP = K2K1*AP
      AM = K2K3*AM
      FP = MU-MU*AM -AP+1
      B=DABS(FP)/FP
      IF(A+B) 1,10,1
    1 CONTINUE
      PC(2) = 3
```

```
      RETURN
   10 M50 = M+50
   18 PN=M
   19 FA = MU*(1,-K2K3**PN)-K2K1**PN+1
      DO 15 L = 1,50
      PNK2K3 = K2K3**PN
      PNK2K1= K2K1**PN
      PN1 = PN +(MU*(1,-PNK2K3)-PNK2K1 +1)/(PNK2K1*DLOG(K2K1)+MU*PNK2K3*
     1DLOG(K2K3))
      PN1FP=(PN1-PN)/PN1
      PN = PN1
      B = MU*(1,-K2K3**PN1)-K2K1**PN1 +1
      PC(3)=PN1
      IF (DABS(PN1FP)-DELTA) 16,16,15
   15 FA = B
   14 PC(2)=2
      RETURN
   16 PC(1) =(C2*K2K1**PN1-C1)/( K2K1**PN1-1, )
      PC(2) = 0
      RETURN
      END
      SUBROUTINE S2A(OUT,FFR,FFI,GR,GI,ALPH,BET,     GAMM,DELT,PIR,PII,
     1  IN,CR,CI,DR,DI,YR,YI,N,ND,LI,LJ,NF,M,NP,NO,EPS,DELTAX)
C     IMPLICIT REAL*8(A-H,O-Z)
C   TO GET THE REAL AND IMAGINARY PART OF A COMPLEX DETERMINANT
C     SUBROUTINE SECI2B(OUT,FFR,FFI,GR,GI,ALPH,BET,GAMM,DELT,PIR,PII,
      DIMENSION CR(250),CI(250),DR(250),DI(250),ER(350),EI(350)
      DIMENSION ALPH(150),BET(150),GAMM(150),DELT(150)
      DIMENSION FFR(350),FFI(350)
      DIMENSION FR(350),FI(350),GR(350),GI(350)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      ALF(FFRI,GRIJ,FFII,GIIJ,AB)      =(FFRI*GRIJ +FFII*GIIJ)*AB
      BETF(FFRI,GIIJ,FFII,GRIJ,AB)=(FFRI*GIIJ-FFII*GRIJ)*AB
      FRF(YR,YI,R,CRMM1,DRMM)=YR**3-3,*YR*((2,*R+YI)**2)+YR*CRMM1+DRMM
      FIF(YR,YI,R,CRMM1) =(2,*R+YI)*(3,*YR*YR-((2*R+YI)**2)+CRMM1   )
      GRF(YR,YI,S,CRM,CIM,DRM)=YR*CRM-(2,*S+YI)*CIM+DRM
      GIF(YR,YI,S,CRM,CIM,DIM)=YR*CIM+(2,*S+YI)*CRM+DIM
      FHR(YR,YI,R,CRMM1,DRMM1)=((YR*YR-(2,*R+YI)**2)**2)+YR*CRMM1-
     1(4,*R*YR+2,*YR*YI)**2 +DRMM1
      FHI(YR,YI,R,CRMM1)=(2,*R+YI)*(4,*YR*(YR*YR-((2,*R+YI)**2))+CRMM1)
      NIN=5
      NP=6
      ORDER=NO
      NDP1=ND+1
      ND2=2*ND+1
      IF(ORDER-9,)1,2,1
    1  IF(ORDER-12,)4,3,4
 1005 FORMAT(41H1IN DELY NEITHER A 9TH OR 12TH ORDER CALL)
    3 NOR=2
      LK=3*NF-2
      GO TO 8
    2 NOR=1
```

```
      LK = 2*NF-1
    8 LJ1 =3*(2*ND+1)
C   LJ1 IS USED FOR 2 DIMENSION INDICES
      LM=LI
      LN=LJ
      LJ=LJ+1
      LK33 = LI*LJ*LK
      MD=-ND
      MZ=0
C
      DO 89 M=1,LM
      MM1=(LI*M-LJ*M)*LK+1
      CRMM1 = CR(MM1)
      DRMM1 = DR(MM1)
C
C         NEGATIVE SUBSCRIPTING STARTS
C
      DO 89 MX=1,ND2
      MR=MX-NDP1
      R=MR
      I = 3*(MR+ND)+M
      NIRM=3*(ND-MR)+M
   23 GO TO (66,67),NOR
C    DO 9 TH ORDER EQUATIONS
   66 FFII=FIF(YR,YI,R,CRMM1)
      FFRI=FRF(YR,YI,R,CRMM1,DRMM1)
      ALPH(I)=1./(FFRI*FFRI+FFII*FFII   )
      FRN= FRF(YR,-YI,R,CRMM1,DRMM1)
      FIN=FIF(YR,-YI,R,CRMM1)
      BET(NIRM)=1,/(FRN*FRN+FIN*FIN)
      GAMM(NIRM)=FRN
      DELT(NIRM)=FIN
      FFI(I)=FFII
      FFR(I)=FFRI
      GO TO 89
   67   FFRI=FHR(YR,YI,R,CRMM1,DRMM1   )
      FFII=FHI(YR,YI,R,CRMM1)
      FFI(I)=FFII
      FFR(I)=FFRI
      ALPH(I)=1,/(FFII*FFII+FFRI*FFRI)
      FRN=FHR(YR,-YI,R,CRMM1,DRMM1)
      FIN=FHI(YR,-YI,R,CRMM1)
      BET(NIRM)=1,/(FRN*FRN+FIN*FIN)
      GAMM(NIRM)=FRN
      DELT(NIRM)=FIN
   89 CONTINUE
      DO 5 MY=1,ND2
      MS=MY-NDP1
      S=MS
      S2 = S+S
      MSND = ND-MS
      MSND3 = MSND+MSND+MSND
      NDMS=ND+MS
      NDMS3 = NDMS+NDMS+NDMS
      DO 5 MX=MY,ND2
      MR=MX-NDP1
      R=MR
      NDMR = ND-MR
      NDMR3 = NDMR+NDMR+NDMR
      MRND= MR+ND
      MRND3 = MRND+MRND+MRND
      MRMS1 = MR-MS+1
```

```
      DO 5 M=1,LM
      I = MRND3+M
C    INDICES ARE IN THREE DIMENSIONS
      NIRM = NDMR3+M
      BA=BET(NIRM)
      FRNEG  =GAMM(NIRM)
      FINEG  =DELT(NIRM)
      NIRM1=(NIRM-1)*LJ1
C     IRM= 3*(MR+ND)+M
      FFRI=FFR(I)
      FFII = FFI(I)
      AB=ALPH(I)
      IRM = I
        IRM1 = (IRM-1)*LJ1
      DO 5 N=1,LN
      MN=(LI*M-LJ+N)*LK
      MNRS1= MN+MRMS1
      NJSN = MSND3+N
      NEGIJ = NIRM1+NJSN
      JSN = NDMS3+N
      IJ= IRM1+JSN
      IF(MRMS1-LK) 6177,6177,6117
 6117 GR(IJ) = 0.
      GI(IJ) = 0
      GR(NEGIJ)=0.
      GI(NEGIJ)=0.
      GO TO 5
 6177 IF(M-N)2111,617,2111
  617 IF(MR-MS)  2111,2112,2111
 2112 GR(IJ) = 1.
      GI(IJ) = 0
      GR(NEGIJ)=1.
      GI(NEGIJ)=0.
      GO TO 5
 2111 CRMNRS= CR(MNRS1)
      CIMNRS= CI(MNRS1)
      DRMNRS= DR(MNRS1)
      DIMNRS= DI(MNRS1)
      GRIJ=GRF(YR,YI,S,CRMNRS,CIMNRS,DRMNRS)
      GIIJ=GIF(YR,YI,S,CRMNRS,CIMNRS,DIMNRS)
      TEMPA =(FFRI*GRIJ + FFII*GIIJ)*AB
      GI(IJ)= (FFRI*GIIJ-FFII*GRIJ)*AB
      GR(IJ) = TEMPA
 3111 GRNEG  =GRF(YR,-YI,S,CRMNRS,CIMNRS,DRMNRS)
      GINEG  =GIF(YR,-YI,S,CRMNRS,CIMNRS,DIMNRS)
      GI(NEGIJ)=-BETF(FRNEG  ,GINEG  ,FINEG  ,GRNEG  ,BA)
      GR(NEGIJ)=ALF(FRNEG  ,GRNEG  ,FINEG  ,GINEG  ,BA)
    5 CONTINUE
 8887 KKKKK=0
 9999 KKKKK=0
      M=6*ND+3
C
      M1 = M-1
      DO 200 KM = 1,M1
      NST = 3
      M2J = KM
      MKM=M-KM
      M2=MKM+1
      M21M = (M2-1)*M
      MM=(M2-1)*M+M2
C
      GRMM= GR(MM)
```

```
      GIMM= GI(MM)
      D =  GRMM*GRMM + GIMM*GIMM
      DII=1./D
      IF(D-EPS)71,72,72
C
C
C     UNLIKELY EVENT     A(MM)**2    + B(MM)**2    TO SMALL
   71 NA=M-KM
      DO 73 LL=1,NA
           LUVM=(LL-1)*M+J
      IF(GR(LUVM) -DELTAX) 91,91,92
   91 IF(GI(LUVM)-DELTAX) 733,733,92
   73 CONTINUE
  733 WRITE(NP,1002)DELTAX,D,M2J
 1002 FORMAT(54H ****UNABLE TO FIND AN ALPHA OR BETA LARGER THAN DELTA /
     1 21H IN SUBROUTINE DELY       /
     2 10H DELTAX=     , G20.8,17H  OLD VALUE USED= G20.8,10H  COLUMN
     1  I5)
      IF(D)    72,999,72
   92 DO 77 J=1,NA
      NAJ=(NA-1)*M+J
      KJ=(LL-1)*M+J
      GR(NAJ) = GR(NAJ) • GR(KJ)
      GI(NAJ) = GI(NAJ) + GI(KJ)
      NANA = (NA-1)*M + NA
      D = GR(NANA)*GR(NANA)+GI(NANA)*GI(NANA)
   77 CONTINUE
 4007 FORMAT(21H ADJUSTMENT ON COLUMN    I5)
      WRITE(NP,4007) M
      GO TO  72
  999 WRITE(NP,1009)
 1009 FORMAT(1H050HNECESSARY TO ABORT DUE TO SINGULARITY                )
      STOP
C
C
   72 MONEY=0
      DO 2200 I=1,MKM
      I1M= (I-1)*M
      IM = I1M+M2
      IM=(I-1)*M+M2
      GRIM = GR(IM)
      GIIM = GI(IM)
      GAM = GRIM*GRMM + GIMM*GIIM
      GA = DII*GAM
      DE = DII*(GIIM*GRMM-GRIM*GIMM)
      DO2200 J = 1, MKM
      IJ = I1M+J
      MJ= M21M+J
      BE = GI(MJ)
      AL = GR(MJ)
      GR(IJ) = GR(IJ)-GA*AL+DE*BE
      GI(IJ) = GI(IJ) - GA*BE-DE*AL
 2200 CONTINUE
  200 CONTINUE
      PIIN = GI(1)
      PIRN = GR(1)
      M1 = M-1
      DO 103 K = 2,M1
      KK=(K-1)*M+K
      GRKK = GR(KK)
      GIKK= GI(KK)
      PIR = GRKK*PIRN-GIKK*PIIN
```

```
      PII = PIIN*GRKK + GIKK*PIRN
      PIRN=PIR
  103 PIIN=PII
      LJ=LJ-1
      RETURN
    4 WRITE(NP,1005)
      RETURN
      END
      SUBROUTINE UP(PC,ND,NSTART,NP)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PC(18)
      COMMON /AERR/     PD,PA,RATIO,IRRCD,MRRCD,NRRCD,IKX
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
C    SUBROUTINE TO ESTABLISH CRITERION FOR CONVERGENCE
C         THE LAST THREE DETERMINANTS MUST BE MONOTONIC, CONVERGING
      IKX=0
      IRRCD=0
      NRRCD=0
      MRRCD=0
      EPSIL = .075
      EP2 = 2.*EPSIL
   51 ND1 = ND-1
      ND2 = ND-2
      PA   = PC(ND1)-PC(ND2)
       PD   = PC(ND)-PC(ND1)
      PA65=PA
      PA76=PD
      PA1=DABS(PA)
      PD1=DABS(PD)
      IF(PD1- PA1)8,55,55
    8 KKKK=0
      IF(PA) 1,2,2
    1 PA=-1.
      GO TO 3
    2 PA=1.
    3 IF(PD) 4,5,5
    4 PD=-1.
      GO TO 6
    5 PD=1.
    6 IF(PA+PD)7,55,7
    7 IRRCD=1
      PD=PD1/PC(ND1)
      IRRCD=0
      MRRCD=3
      PA=PA1/PC(ND2)
      MRRCD=0
      PA=DABS(PA)
      PD=DABS(PD)
      IF(PA-EPSIL) 17,17,55
   17 IF(PD-EPSIL) 18,18,55
   18 KKKK=0
      NRRCD=5
```

```
      RATIO=PA76/PA65
      NRRCD=0
      IF (IKX,GT,0)
     1 WRITE (6,9876) IRRCD,MRRCD,NRRCD
 9876 FORMAT (*0*,*DIVISION BY ZERO ERROR *,I5,I5,I5)
      IF(RATIO)55,55,19
   19 IF(RATIO-,8)54,55,55
   54 CALL PRE(ND,PC,NP)
      NSTART = 0
      RETURN
   55 CONTINUE
      NSTART = 1
      RETURN
      END
      SUBROUTINE S3(OUT,PSI,ETA,YR,YI,N,IN,DD,CR,DR,NF,K,NP,NIMAG9)
C     IMPLICIT REAL*8(A-H,O-Z)
           INTEGER OUT
      DIMENSION ROLY(6),POLY(6)
      DIMENSION PSI(1),ETA(1),YR(1),YI(1),CR(1),DR(1)
      DIMENSION ROOTR(12),ROOTI(12),RR(12),RRI(12)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
C   K IS THE ORDER
C    N  IS THE NUMBER OF COMPLEX PAIRS
      NOUT=NP
C    SUBROUTINE TO GET THE POLYNOMIAL ROOTS OF THE 9TH AN 12TH ORDER
      DELTA= DD
      LI=3
      L=0
            LJ1=4
      NF3=3*NF-2
      IF(K-9) 1,2,1
    1 IF(K-12)  99,3,99
    2 LK=2*NF-1
      NORDER=1
      NK=3
      GO TO 4
    3  LK=3*NF-2
      NORDER=2
      NK=4
      KNF3 = 3*NF-2
    4 DO  20 M= 1,3
      L=(M-1)*NK
      NN1=(LI*M-LJ1+M)*LK+1
      C = CR(NN1)
      D= DR(NN1)
      GO TO ( 9,12),NORDER
    9 ROLY(4)= 1,
      ROLY(3)=0,
      ROLY(2)= C
      ROLY(1)= D
      IK=4
      GO TO  10
```

```
   12 ROLY(5)= 1
       ROLY(4)=0
      ROLY(3)=0
      ROLY(2)=C
      ROLY(1)= D
      IK=5
 4242 FORMAT(1H ,6G20.6)
   10 CALL PROD(ROLY,IK,ROOTR,ROOTI,POLY,NUM,IER)
      WRITE(6,4242)ROOTI
      WRITE(6,4242)ROOTR
   14 IF(OUT) 21,21,16
   16 GO TO  (919,912),NORDER
41112  FORMAT(*0*// * POLYNOMIAL     *5HX**3 * + *F16.6,*X + *F16.6/
     1 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/
     1 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/
     1 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/)
 1002 FORMAT(3X,25HERROR IN PROD SUBROUTINE , I5/)
41113  FORMAT(*0*// * POLYNOMIAL     *5HX**4 * + *F16.6,*X + *F16.6/
     1 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/
     2 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/
     3 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/
     4 * REAL PART OF ROOT * F16.6,*   IMAGINARY PART OF ROOT * F16.6/)
  919 WRITE(NP,41112)C,D,(ROOTR(J),ROOTI(J),J=1,NK)
      GO TO 21
  912 WRITE(NP,41113)C,D,(ROOTR(J),ROOTI(J),J=1,NK)
   21 DO 26 LM=1,NK
      LML=LM+L
      RR(LML) = ROOTR(LM)
   26 RRI(LML) = ROOTI(LM)
      MML=(M-1)*4
   20 CONTINUE
      WRITE (6,4242) RR,RRI
C     ALL TWELVW ROOTS OF THE 12TH ORDER SYSTEM HAVE BEEN FOUND
C     ALL NINE ROOTS HAVE BEEN DETERMINED
C     SEARCH FOR THE ZERO IMAGINARIES
      KTOP=K
      KLOW = 1
      DO 30 M=1,K
      IF(RRI(M))31,32,31
   32 PSI(KTOP)=RR(M)
      YR(KTOP)=RR(M)-DELTA
      YI(KTOP)=0.
      ETA(KTOP)=0.
      KTOP = KTOP-1
        GO TO 30
   31 YR(KLOW)=RR(M)-DELTA
      PSI(KLOW)=RR(M)
      ETA(KLOW)=RRI(M)
      YI(KLOW)=RRI(M)
      KLOW = KLOW+1
   30 CONTINUE
      NIMAG9=KLOW-1
   47 N= KLOW/2
C     ALL TWELVW ROOTS OF THE 12TH ORDER SYSTEM HAVE BEEN FOUNC
      IF(OUT) 100,100,98
   98 DO 140 M=1,K
      WRITE(NP,1015) M,PSI(M)
      WRITE(NP,1016) M,ETA(M)
      WRITE(NP,1017)M,YR(M),YI(M)
 1016 FORMAT(1H ,6HETA  ( I5,4H) =  E20.8)
 1017 FORMAT(1H 6HYR   (,I5,4H) =  E20.8,7H IMAG,  E20.8)
 1015 FORMAT(1H0,6HPSI  (,I5,4H) =  E20.8         )
```

```
  140 CONTINUE
      GO TO 100
   99 IN=1
  100 RETURN
      END
      SUBROUTINE S1B(OUT,CR,CI,IN,AR,BAR,AI,BAI,BR,BI,LI,LJ,LK,NF,NP)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION CI(250),CR(250),AR(250),BAR(250),BAI(250),AI(125)
      DIMENSION BR(250),BI(25)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
C   FOR THE CALCULATION OF THE CR(I,J,K) AND THE CI(I,J,K) USE,..
C   CALL  SECT1(CR,CI,AR,AR,AI,AI,BR,BI,LI,LJ,LK,NF)
C   FOR THE CALCULATION OF THE DR(I,J,K) AND THE DI(I,J,K) USE,.,
C     CALL  SECT1(DR,DI,AR,BR,AI,BI,DUM,DUM,LI,LJ,LK,NF)
C    WHERE DUM IS A   VECTOR AS LONG AS BI AND BR BUT = TO ZERO EVERYWHE
C
      LJ=LJ+1
      LJ1 = LJ-1
      LK=NF
      N2F1 = 2 * NF -1
      NF1 = NF + 1
      KL = 2*NF-1
C
      DO 5 I=1, LI
C
      DO 5 J = 1, LJ1
C
      IKL = (LI*I-LJ*J)*KL
            IJ=(LI * I -LJ+J)*LK
C     FOR K = 1, 2, .,, , NF
C
                    DO  55  K = 1 , NF
      AK = K-1
      IJKL = IKL+K
      IJK = IJ + K
      CI(IJKL) = 2.* AK * BAR(IJK) + BI(IJK)
      CR(IJKL)= -2.*AK*BAI(IJK) + BR(IJK)
C   PERFORM THE SUMMATIONS
      CONSTR =0
      CONSTI = 0
C
      DO  7 L= 1, 3
      IL = (LI*I- LJ + L)*LK
       JL= (LI*L -LJ+J)*LK
      CONST1 =0
      CONST4 =0.
C
      DO 15 N = 1 , K
      ILN = IL+N
      LJK1N = JL + K +1 -N
      ARILN = AR(ILN)
      BALJK1=BAR(LJK1N)
```

```
      AIILN=AI(ILN)
      BILJK1=BAI(LJK1N)
       APPLE= BALJK1*ARILN
      PEAR = AIILN*BILJK1
      CONST1 = CONST1+APPLE-PEAR
C     CONST1=CONST1+ARILN*BALJK1-AIILN*BILJK1
      APPLE=ARILN*BILJK1
C     CONST1 = CONST1 + AR(ILN) * BAR(LJK1N) - AI(ILN) * BAI(LJK1N)
C     CONST4 = CONST4 * AR(ILN) * BAI(LJK1N) + AI(ILN) * BAR(LJK1N)
      PEAR = AIILN*BALJK1
      CONST4=CONST4+APPLE+PEAR
   15 CONTINUE
      CONST2 = 0
      CONST5 = 0
      IF(NF-K) 17, 17, 16
   16 NFK=NF-K
C
      DO 20 N=1, NFK
      ILN1 = IL + N + 1
      LJNK=JL+N+K
      ILNK = IL + N + K
      LJN1 = JL + N +1
      ARILN1=AR(ILN1)
      BRLJNK=BAR(LJNK)
      AIILN1=AI(ILN1)
      BILJNK=BAI(LJNK)
      ARILNK=AR(ILNK)
      BRLJN1=BAR(LJN1)
      AIILNK=AI(ILNK)
      BILJN1=BAI(LJN1)
      APPLE=ARILN1*BRLJNK
      PEAR = AIILN1*BILJNK
      PEACH=ARILNK*BRLJN1
      PLUM = AIILNK*BILJN1
      CONST2 = CONST2+APPLE+PEAR+PEACH+PLUM
      APPLE=ARILN1*BILJNK
      PEAR  = AIILN1*BRLJNK
      PEACH = ARILNK*BILJN1
      PLUM = AIILNK*BRLJN1
       CONST5 = CONST5 + APPLE-PEAR-PEACH+PLUM
C     CONST5 = CONST5+ARILN1*BILJNK-AIILN1*BRLJNK-ARILNK*BILJN1+AIILNK*
   20 CONTINUE
   17 CONSTR= CONSTR + CONST1 + CONST2
      CONSTI = CONSTI + CONST4 + CONST5
    7 CONTINUE
      CR(IJKL)= CR(IJKL)- CONSTR
      CI(IJKL)= CI(IJKL)- CONSTI
   55 CONTINUE
      DO 9 K = NF1,N2F1
      CONSTR=0,0
      CONSTI = 0,0
      IJKL = IKL + K
         DO 8 L= 1,3
      IL= (LI*I -LJ+L)*LK
      JL = ( LI*L -LJ+J)*LK
      CONST3 = 0,0
      CONST6 = 0,0
      K1N = K + 1 - NF
C
   22 DO 30 N = K1N ,NF
      ILN = IL+N
      LJK1N = JL + K + 1 -N
```

```
      ARILN=AR(ILN)
      BRLJK1=BAR(LJK1N)
      AIILN=AI(ILN)
      BILJK1=BAI(LJK1N)
      CONST3 = CONST3+ARILN*BRLJK1-AIILN*BILJK1
      CONST6 = CONST6+ARILN*BILJK1+AIILN*BRLJK1
   30 CONTINUE
   21 CONSTR = CONSTR + CONST3
    8 CONSTI=CONSTI+CONST6
      CI(IJKL)=-CONSTI
      CR(IJKL)=     - CONSTR
    9 CONTINUE
    5 CONTINUE
      LJ= LJ-1
       RETURN
      END
      SUBROUTINE  S1C(AR,AI,CR,CI,ACR,ACI,DR,DI,ER,EI,LI,LJ,LK,NF)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION CR(250),CI(250),DR(250),DI(250),ER(350),EI(350)
      DIMENSION AR(125),AI(125),ACR(250),ACI(250)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
C     CALL SECT1C(BR,BI,CR,CI,DR,DI,DUM,DUM,FR,FI,LI,LJ,LK,NF)
C    LK IS THE LENGTH OF K
C - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      LK=3*NF-2
      LK2NF = 2*NF-1
      LKNF = NF
      LJ=LJ+1
      LJ3=LJ+1
      LJ1= LJ-1
      DO 5 I=1,LI
        DO 5 J= 1,LJ1
      IJ = (LI*I-LJ+J)*LK
      IJ2 = (LI*I-LJ+J)*LK2NF
      DO 45 K=1,NF
      AK= K+K-2
      IJK=IJ+K
      IJK2 = IJ2+K
      ER(IJK)  = -AK*ACI(IJK2)+DR(IJK2)
      EI(IJK) = AK*ACR(IJK2)+DI(IJK2)
C     ER(IJK)=-AK*ACI(IJK)+DR(IJK)
C     EI(IJK) = AK*ACR(IJK)+DI(IJK)
      CONSTR=0
      CONSTI=0
      DO 55 L=1,3
      IL = (LI*I-LJ+L)*LK2NF
      JL = (LI*L-LJ+J)*LKNF
C      JL =(LI*L-LJ+J)*LK
C      IL = (LI*I-LJ+L)*LK
      CONST1=0
      CONST7=0
      NFK=NF-K
```

```
      IF(NFK) 3,3,4
    4 DO 15 N = 1,NFK
      ILN1=IL+N+1
      LJKN=JL+K+N
      CONST1=CONST1+CR(ILN1)*AR(LJKN) +CI(ILN1)*AI(LJKN)
      CONST7= CR(ILN1) * AI(LJKN) - CI(ILN1)*AR(LJKN) + CONST7
   15 CONTINUE
    3 CONST2=0.0
      CONST8=0
      NF1=NF-1
      DO 20 N= 1, NF1
      ILKN=IL+K+N
      LJN1=JL+N+1
      CONST2=CR(ILKN) * AR(LJN1) + CI(ILKN)* AI(LJN1) + CONST2
      CONST8 = CONST8 - CR(ILKN)* AI(LJN1) + CI(ILKN)*AR(LJN1)
   20 CONTINUE
      CONST3    =0
      CONST9 =0
      DO 25 N=1,K
      ILK1N = IL+K+1-N
      LJN= JL+N
      CONST3 = CONST3 + CR(ILK1N) * AR(LJN) - CI(ILK1N) * AI(LJN)
      CONST9= CONST9 + CR(ILK1N) * AI(LJN) + CI(ILK1N) * AR(LJN)
   25 CONTINUE
      CONSTR=CONSTR+ CONST1+CONST2+CONST3
      CONSTI=CONST7 + CONST8 + CONST9  +CONSTI
   55 CONTINUE
      ER(IJK) = ER(IJK)- CONSTR
      EI(IJK)= EI(IJK)- CONSTI
   45 CONTINUE
C
      NF1= NF+1
      NF21 = 2*NF-1
      DO 60 K = NF1,NF21
      IJK=IJ+K
      IJK2 = IJ2+K
      AK = K + K -2
      EI(IJK) = AK* ACR(IJK2)+DI(IJK2)
      ER(IJK) = -AK*ACI(IJK2) + DR(IJK2)
C     EI(IJK) = AK*ACR(IJK)+DI(IJK)
C     ER(IJK) = -AK*ACI(IJK)+DR(IJK)
      CONSTR=0
      CONSTI=0
      DO 65 L=1,3
      NN=2*NF-1-K
      IL = (LI*I-LJ+L)*LK2NF
      JL = (LI*L-LJ+J)*LKNF
C     IL=(LI*I-LJ+L)*LK
C     JL = (LI*L-LJ+J)*LK
      CONST4 =0.0
      ONST10=0.
      IF(NN)66,66,67
   67 DO 70 N = 1,NN
       ILKN=IL+K+N
      LJN1 = JL+N+1
       ONST10=-CR(ILKN)* AI(LJN1) + CI(ILKN) * AR(LJN1) +  ONST10
      CONST4 = CR(ILKN) * AR(LJN1) + CI(ILKN) * AI(LJN1)  + CONST4
   70 CONTINUE
   66 CONST5=0.0
       ONST11 = 0
C
      DO 80 N=1,NF
```

```
      ILK1N=IL+K+1-N
      LJN=JL+N
       ONST11 =  ONST11 + CR(ILK1N)*AI(LJN)+CI(ILK1N)*AR(LJN)
      CONST5=CONST5+CR(ILK1N)*AR(LJN)-CI(ILK1N)*AI(LJN)
   80 CONTINUE
C
      CONSTR=CONSTR+CONST4+CONST5
      CONSTI=CONSTI+ ONST10+  ONST11
   65 CONTINUE
      ER(IJK) = ER(IJK) • CONSTR
      EI(IJK) = EI(IJK)  + CONSTI
   60 CONTINUE
C
      NF2 = 2*NF
      NF32= 3*NF-2
      DO 100 K=NF2,NF32
      IJK=IJ+K
      IJK2 = IJ2+K
      CONSTR = 0
      CONSTI=0
C
      DO 105 L=1,3
C     IL= (I+LI-LJ+L)*LK
      IL = (I+LI-LJ+L)*LK2NF
C     JL =(LI+L-LJ+J)*LK
      JL = (LI+L-LJ+J)*LKNF
      CONST6= 0
       ONST12 = 0
      NN = K-2*NF • 2
      IF(NF- NN) 101, 102, 102
C
  102 DO 110 N = NN , NF
      ILK1N= IL+K+1-N
      LJN=JL+N
      CONST6 = CR(ILK1N) * AR(LJN) - CI(ILK1N) * AI(LJN)
       ONST12 = CR(ILK1N) • AI(LJN) • CI(ILK1N) • AR( LJN)
  110 CONTINUE
C
  101 CONSTR= CONSTR + CONST6
      CONSTI = CONSTI +  ONST12
  105 CONTINUE
      ER(IJK)=  • CONSTR
      EI(IJK) = ■ CONSTI
  100 CONTINUE
C
    5 CONTINUE
C
      LJ=LJ-1
      RETURN
      END
      SUBROUTINE  PROD(XCOF,M,ROOTR,ROOTI,COF, NUM,IER)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION XCOF(13),COF(13),ROOTR(13),ROOTI(13)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
```

```
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
C     COMPUTES THE REAL AND COMPLEX ROOTS OF A POLYNOMIAL
      IFIT=0
      N=M-1
      IER=0
      IF(XCOF(N+1))10,25,10
   10 IF(N)15,15,32
   15  IER=1
   20 RETURN
   25   IER=4
      GO TO 20
   30  IER=2
      GO TO 20
   32  IF(N-36)  35,35,30
   35 NX=N
      NXX=N+1
      N2=1
      KJ1= N+1
      DO 40 L=1,KJ1
      MT= KJ1-L+1
   40  COF(MT)=XCOF(L)
   45  XO= ,00500101
      YO= ,01000101
      IN=0
   50 X=XO
      XO=-10,*YO
      YO=-10,*X
      X=XO
      Y=YO
      IN=IN+1
      GO TO 59
   55 IFIT= 1
      XPR=X
      YPR=Y
   59  ICT= 0
   60 UX= 0.
      UY=0,
      V=0.
      YT=0,
      XT=1,
      U=COF(N+1)
      IF(U) 65,130,65
   65 DO 70 I=1,N
      L=N-I+1
      TEMP= COF(L)
      XT2= X*XT-Y*YT
      YT2= X*YT+Y*XT
      V=V+TEMP*YT2
      U=U+TEMP*XT2
      FI=I
      UX=UX+FI*XT*TEMP
      UY=UY-FI*YT*TEMP
      XT=XT2
   70 YT=YT2
      SUMSQ= UX*UX+UY*UY
      IF(SUMSQ) 75,110,75
   75 DX= (V*UY-U*UX)/SUMSQ
      X=X+DX
      DY= -(U*UY+V*UX)/SUMSQ
      Y=Y+DY
   78  IF(DABS(DY)+DABS(DX)-1,D-07)  100,80,80
```

```
 80 ICT=ICT+1
    IF(ICT-500) 60,85,85
 85 IF(IFIT) 100,90,100
 90  IF(IN-5) 50,95,95
 95 IER=3
    RETURN
100 DO 105 L=1,NXX
    MT=KJ1-L+1
    TEMP=XCOF(MT)
    XCOF(MT)=COF(L)
105 COF(L)= TEMP
    ITEMP=N
    N=NX
    NX=ITEMP
    IF(IFIT) 120,55,120
110 IF(IFIT) 115,50,115
115 X=XPR
    Y=YPR
120 IFIT= 0
    IF(X) 122,125,122
122 IF (DABS(Y/X)-1,E-05) 135,125,125
125 ALPHA= X+X
    SUMSQ= X*X+Y*Y
    N=N-2
    GO TO 140
130 X=0.
    NX= NX-1
    NXX= NXX-1
135  Y=0,
    SUMSQ=0.
    ALPHA=X
    N=N-1
140 COF(2)= COF(2)+ALPHA*COF(1)
    IF (N,LT,2) GO TO 155
145  DO 150 L=2,N
150 COF(L+1)=COF(L+1)+ALPHA*COF(L)-SUMSQ*COF(L-1)
155 ROOTI(N2)=Y
    ROOTR(N2)= X
    N2=N2+1
    IF(SUMSQ) 160,165,160
165 IF (N) 20,20,45
160 Y=-Y
    SUMSQ=0
    GO TO 155
    END
    SUBROUTINE SIMQ(A,N,Y)
C   IMPLICIT REAL*8(A-H,O-Z)
    DIMENSION A(150),Y(15),ICHG(15),SV(15)
    DSQRT(D)=SQRT(D)
    DMAX1(A,B,C)=AMAX1(A,B,C)
    DABS(D)=ABS(D)
    DATAN(D)=ATAN(D)
    DCOS(D)=COS(D)
    DEXP(D)=EXP(D)
    DLOG(D)=ALOG(D)
    DLOG10(D)=ALOG10(D)
    DSIN(D)=SIN(D)
    DCOSH(D)=COSH(D)
    DSINH(D)=SINH(D)
C   SUBROUTINE FOR SOLVING SIMULTANEOUS EQUATIONS USING KROUTS METHOD
    DO 1000 I=1,N
    II = (I-1)*N+I
```

```
      SV(I) = A(II)
      IF(SV(I))5,46,5
    5 Y(I)=Y(I)/SV(I)
      DO 1000 J=1,N
       IJ = (I-1)*N+J
      A(IJ) = A(IJ)/SV(I)
 1000 CONTINUE
      DO 101 K=1,N
      KK = (K-1)*N+K
      AMX=DABS(A(KK))
      IMX=K
      DO 15 I=K,N
      IK = (I -1)*N+K
      IF(DABS(A(IK))-AMX) 15,15,14
   14 AMX=DABS(A(IK))
      IMX=I
   15 CONTINUE
      IF(AMX)27,46,27
   27 IF(IMX-K)8,9,8
    8 DO 22 J=1,N
      KJ = (K-1)*N+J
      TEMP=A(KJ)
      IMXJ =(IMX-1)*N+J
      A(KJ) = A(IMXJ)
   22 A(IMXJ )=TEMP
      ICHG(K)=IMX
      TEMP=Y(K)
      Y(K)= Y(IMX)
      Y(IMX)= TEMP
      GO TO 10
    9 ICHG(K)=K
   10 A(KK) = 1./A(KK)
      DO 33 J=1,N
      IF (J-K) 6,33,6
    6 KJ = (K-1)*N+J
      A(KJ) = A(KJ)*A(KK)
   33 CONTINUE
      Y(K) = Y(K)*A(KK)
      DO 44 I=1,N
      IK = (I-1)*N+K
    7 DO 45 J=1,N
      IF(I-K)17,44,17
   17 IF(K-J)18,45,18
   18 IJ = (I-1)*N+J
      KJ = (K-1)*N+J
      A(IJ) = A(IJ) -(A(IK)*A(KJ))
C  18 A(I,J)=A(I,J)-(A(I,K))*(A(K,J))
   45 CONTINUE
C     Y(I) = Y(I)-A(I,K)*Y(K)
      Y(I) = Y(I)-A(IK)*Y(K)
   44 CONTINUE
      DO 99 I=1,N
      IF(I-K)26,99,26
   26 IK = (I-1)*N+K
      A(IK) = -A(IK)*A(KK)
   99 CONTINUE
  101 CONTINUE
      DO 70 K=1,N
      L=N+1-K
      KI=ICHG(L)
      IF (L-KI) 68,70,68
   68 DO 69 I=1,N
```

```
      IL = (I-1)*N+L
      TEMP = A(IL)
      IKI =(I-1)*N+KI
      A(IL) = A(IKI)
      A(IKI) = TEMP
   69 CONTINUE
   70 CONTINUE
      DO 1001 I=1,N
      DO 1001 J=1,N
      IJ = (I-1)*N+J
      A(IJ) = A(IJ)/SV(J)
 1001 CONTINUE
      RETURN
   46 N=-N
      RETURN
      END
      SUBROUTINE S4(CC,BIGA,A,B,C,XR,XI,U,SI,E,NO,K,PI,DYR,DYI,YR,YI)
C     IMPLICIT REAL*8(A-H,O-Z)
      INTEGER STEP
      DIMENSION XR(12),XI(12)
      DIMENSION A(13),B(13),C(13),CC(150),YR(13),YI(13),DYR(13)
      DIMENSION DYI(13),X(13),E(13),BIGA(13),U(13),SI(13)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
C  PROGRAM FOR SETTING UP THE SIMULTANEOUS EQUATIONS
C        INPUT PARAMETERS
C     A(I), B(I),C(I) ARE DUMMY STORAGE
C      MAXIMUM LENGTH IS 12 FOR  A,B,C
C      DELTY IS RHE PARAMETER TO BE USED IF DENOMINATOR BECOMES SMALL
C    CC(I) IS THE RETURNED SET OF SUMULTANEOUS EQUATIONS
C    Y  ARE THE INPUT DETERMINANT VALUES
C   DY  ARE THE DELTA Y VALUES
C    X   IS DUMMY STORAGE OF SIZE 12
C    E(I) IS ETA(J)   MAX SIZE IS 12
C    U (I)  IS DUMMY OF SIZE 12
      XMX=150.
      NO1=NO-1
      NK=K*2
      DO5 N=2,NK,2
      A(N)=PI*SI(N)
      B(N)=PI*E(N)
    5  CONTINUE
      N1=NK+1
      NK1=NK-1
      DO 10  N=N1,NO
   10  C(N) = PI*SI(N)
      DO 25  MM=1,NK1,2
      XR(MM)=-PI*YR(MM)
      XI(MM)=PI*(-YI(MM))
      XXR=XR(MM)
      XXI=XI(MM)
      DO 15 N=1,NK1,2
      NN=N+1
```

```
      BNN=B(NN)+XXI
         XIB=B(NN)-XXI
      AA=A(NN)
      XAA=XXR+AA
      XAA=DABS(XAA)
      IF(XMX.LT.XAA) GO TO 200
      COSHY=DCOSH(XAA)
      U(N)=((DSIN(BNN)/(COSHY-DCOS(BNN)))+(DSIN(XIB)/(COSHY-DCOS(XIB))))
     1*.5
      GO TO 15
  200 U(N)=0.
   15  CONTINUE
      DO 20 N=2,NK,2
      AA=A(N)
      XAA=XXR+AA
      BNN=B(N)
      XIB=XXI+BNN
      XIBN=XXI+BNN
      BNXI=BNN-XXI
      XXX=DABS(XAA)
      IF (XMX.LT.XXX) GO TO 201
      SINHY=DSINH(XAA)
      COSHY=DCOSH(XAA)
      U(N)=((SINHY/(COSHY-DCOS(XIBN)))+(SINHY/(COSHY-DCOS(BNXI))))*.5
      GO TO 20
  201 U(N)=XAA/XXX
   20   CONTINUE
      DO 21  N=N1,NO
      XAA=XXR+C(N)
      XXX=DABS(XAA)
      IF (XMX.LT.XXX) GO TO 202
      SINHY=DSINH(XAA)
      COSHY=DCOSH(XAA)
      GO TO 203
  202 U(N)=XAA/XXX
      GO TO 21
  203 XIBN=XXI
       U(N)= SINHY/(COSHY-DCOS(XIBN))
   21 CONTINUE
      DO 30 N=1,NO
      MN=(MM-1)*NO+N
   30  CC(MN)=U(N)
   25  CONTINUE
      DO 125 MM=2,NK,2
      XXR=-PI*YR(MM)
      XXI=-PI*YI(MM)
      DO 130 N=1,NK1,2
      XAA=XXR+A(N+1)
      XAA=DABS(XAA)
      IF (XMX.LT.XAA) GO TO 204
      SINHY=DSINH(XAA)
      COSHY=DCOSH(XAA)
      BNXI=B(N+1)-XXI
      XIBN= B(N+1)+XXI
      U(N)=(SINHY/(COSHY-DCOS(BNXI))-(SINHY/(COSHY-DCOS(XIBN))))*.5
      GO TO 130
  204 U(N)=0.
      BNXI=B(N+1)-XXI
      XIBN= B(N+1)+XXI
  130  CONTINUE
      DO 135  N=2,NK,2
      XAA=XXR+A(N)
```

```
      XAA=DABS(XAA)
      IF (XMX.LT.XAA) GO TO 205
      BNXI=B(N)-XXI
      XIBN = B(N)+XXI
      COSHY=DCOSH(XAA)
      U(N)=(-DSIN(BNXI)/(COSHY-DCOS(BNXI))+DSIN(XIBN)/(COSHY-DCOS(XIBN))
     1)*.5
      GO TO 135
  205 U(N)=0.
  135  CONTINUE
      DO 140 N=N1,NO,1
      XAA=XXR+C(N)
      XAA=DABS(XAA)
      IF (XMX.LT.XAA) GO TO 206
      COSHY=DCOSH(XAA)
      U(N)=DSIN(XXI)/(COSHY-DCOS(XXI))
      GO TO 140
  206 U(N)=0.
  140 CONTINUE
C 140 U(N)=DSIN(XXI)/(COSHY-DCOS(XXI))
      DO  145  N=1,NO
      MN=(MM-1)*NO+N
  145  CC(MN) = U(N)
  125  CONTINUE
      DO 150 MM=N1,NO1
      M=(MM-1)*NO
      XXR= -PI*YR(MM)
      DO 155 N= 1,NK1,2
      XAA=XXR+A(N+1)
      XAA=DABS(XAA)
      IF (XMX.LT.XAA) GO TO 207
      COSHY=DCOSH(XAA)
      MN=M+N
      XIBN=B(N+1)
      CC(MN)=DSIN(XIBN)/(COSHY-DCOS(XIBN))
      GO TO 155
  207 MN=M+N
      XIBN=B(N+1)
      CC(MN)=0.
  155 CONTINUE
      DO 160  N=2,NK,2
      MN=M+N
      XAA=XXR+A(N)
      XXX=DABS(XAA)
      IF (XMX.LT.XXX) GO TO 208
      COSHY=DCOSH(XAA)
      SINHY=DSINH(XAA)
      XIBN=B(N)
      CC(MN)=SINHY/(COSHY-DCOS(XIBN))
      GO TO 160
  208 CC(MN)=XAA/XXX
  160 CONTINUE
      DO 165  N=N1,NO
      MN=M+N
      XAA=(XXR+C(N))*.5
      XXX=DABS(XAA)
      IF (XMX.LT.XXX) GO TO 209
       CC(MN)=DCOSH(XAA)/DSINH(XAA)
      GO TO 165
  209 CC(MN)=XAA/XXX
  165 CONTINUE
  150  CONTINUE
```

```
      DO 40 N=1,NK1,2
      MN=NO*(NO-1)+N
      MN1= MN+1
      CC(MN1)=1.
      CC(MN) = 0.
   40 CONTINUE
      DO 70  N=NK,NO
      MN= NO*NO1+N
   70 CC(MN) = 1.
      DO 50 M=1,NK1,2
      BIGA(M)= DYR(M)-1.
   50  BIGA(M+1)= DYI(M)
      DO 60  M=N1,NO1
   60  BIGA(M)= DYR(M)-1.
      BIGA(NO) = 0
      RETURN
      END
      SUBROUTINE S5A(OUT,AK,IN,K,MORDER,F,PB,PA,PC,IERR)
      IMPLICIT REAL*8(A-H,O-Z)
      INTEGER R21
      INTEGER R1
      INTEGER R,R2,R22
      DIMENSION BIGA(13,13),A(13,13),B(13,13),C(13,13)
      DIMENSION BIGAP(13,13,6),BP(13,13,6),CP(13,13,6),AP(13,13,6)
      DIMENSION F(13),AC(13)
      DIMENSION AR(13),AS(13),P(13),Q(13)
      DIMENSION AK(13),PC(13),PA(13),PB(13)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      NP=6
      DO 45 M=1,K
      M2= 2*M
      M21=M2-1
      PBM2 = PB(M2)
      COSB2M =DCOS(PBM2)
      PAM2 = PA(M2)
      EA2M =DEXP(-PAM2)
      F2M = F(M2)
      E2A2M =DEXP(-2.*PAM2)
      F2M1=F(M21)
      AR(M) = 2.*EA2M*(F2M1*DSIN(PBM2)+F2M*COSB2M)
      AS(M) = -2.*F2M*E2A2M
      P(M) = -2.*EA2M*COSB2M
      Q(M) = E2A2M
      WRITE(NP,2000)P(M),Q(M),AR(M),AS(M)
   45 CONTINUE
      K1=K+1
      DO 10 M=K1,MORDER
      M2 = 2*M
      M21 = M2-1
      PCM2 = -PC(M2)
      EC2M =DEXP(PCM2)
      PCM21 =-PC(M21)
```

```
      EC2M1 =DEXP(PCM21)
      F2M = F(M2)
      F2M1 = F(M21)
      AR(M) = 2.*(F2M1*EC2M1+F2M*EC2M)
      CC9 =  PCM21-PCM2
C     CC9 = -PCM21-PCM2
      ECC9 =DEXP(CC9)
      AS(M) = -2.*ECC9*(F2M1+F2M)
        P(M) = -(EC2M1+EC2M)
      Q(M) = ECC9
      WRITE(NP,20000)M
20000 FORMAT(5H M =  ,I5/)
      WRITE(NP,2000)P(M),Q(M),AR(M),AS(M)
 2000 FORMAT(1H0,/15X,4HP(M),15X,4HQ(M),15X,4HR(M),15X,4HS(M),/
     1 /3X,4G19.6)
   10 CONTINUE
      NORDER=MORDER*2 +1
      NL1= NORDER +1
      DO 25 I = 1,13
      DO 25 R  = 1,13
   25 A(I,R) = 0.
C     ADJUSTED TO AVOID ZERO INDICES   A,1,2) AND A(2,2)
      A(2,2) = Q(1)
      A(1,2) = P(1)
      R=MORDER
      MR2 = 2.*MORDER
      MR21= MR2+1
      DO 5 I = 1,MR2
      KI=I
      KI1 = KI-1
      KI2= KI-2
      DO 5 R = 1,MORDER
      R2 = 2*R
C     R22 = 2*R-2+1
C     ADJUSTED TO AVOID ZERO INDICES
      R22=R2-1
      IF(KI2) 1,2,3
    1 BIGA(I,R)= 0.
      GO TO 7
    2 BIGA(I,R)= 1.
    7 IF(KI1) 4,6,3
    4 B(I,R)= 0.
      GO TO  3
    6 B(I,R)= 1.
    3  IF(KI2.GT.0)  BIGA(I,R)=A(KI2,R22)
      IF(KI1.GT.0)  B(I,R)=A(KI1,R22)
   33 C(I,R)= A(KI,R22)
      R21 = R2+1
      A(KI,R21) = BIGA(I,R)*Q(R) + B(I,R)*P(R) + C(I,R)
    5 CONTINUE
      A(3,3) = 0.0
      A(2,3) = 0.0
      A(1,3) = 0
      DO 21 R = 4,R2,2
      R1 = R+1
      DO 21 I = 1,R2
      A(I,R) = A(I,R1)
   21 A(I,R1) = 0
      WRITE (NP,300) ((I,J,A(I,J),I=1,13),J=1,13)
  300 FORMAT(1X,2HA(,I5,1H,I5,2H)=G20.6)
C    NEXT, DEFINE THE SUCCESSION OF MODIFIED TRAINGULAR ARRAYS
C     AP(I,2R,M)    I= 1,2,...,2R   R=1,2,..,MORDER-1 M = 1,2,..,MORDER
```

```
C     NOTE THE SUCCESSION HALTS AT R=MORDER-1
      NORDER=13
      DO 30 I = 1, NORDER
      DO 30 R = 1,NORDER
      DO 30 M = 1,MORDER
   30 AP(I,R,M) = 0,0
      DO 58 M = 1,MORDER
      TMPP=P(M)
      TMPQ=Q(M)
      Q(M)=Q(MORDER)
      P(M) = P(MORDER)
      AP(2,2,M) = Q(1)
      AP(1,2,M) = P(1)
      ML1 = MORDER -1
      MR2 = 2*MORDER
      MR21 = MR2+1
      DO 50 I = 1,MR2
      KI=I
      KI1=KI-1
      KI2=KI-2
      DO 50 R = 1,ML1
      R2 = 2*R
      R22 = R2-1
      IF(KI2)  51,52,53
   51 BIGAP(I,R,M)= 0,
      GO TO 57
   52 BIGAP(I,R,M)= 1,
   57 IF(KI1) 54,56,53
   54  BP(I,R,M)= 0,
      GO TO 53
   56  BP(I,R,M)= 1,
   53  IF(KI2,GT,0)  BIGAP(I,R,M)=AP(KI2,R22,M)
      IF(KI1,GT,0)  BP(I,R,M)=AP(KI1,R22,M)
   63   CP(I,R,M)= AP(KI,R22,M)
      R21 = R2+1
      AP(I,R21,M) = BIGAP(I,R,M)*Q(R)+ BP(I,R,M)*P(R) + CP(I,R,M)
   50 CONTINUE
      Q(M) = TMPQ
      P(M) = TMPP
   58 CONTINUE
C     NOW BACK SHIFT
      R= MORDER-1
      R2 = 2*R
      DO 121 I = 1,NORDER
      DO 121 M=1,MORDER
      DO 121 R = 4,R2,2
      R1 = R+1
      AP(I,R,M) = AP(I,R1,M)
  121 AP(I,R1,M) = 0
        DO 122 M=1,MORDER
      AP(1,3,M)=0
      AP(2,3,M)=0
  122 CONTINUE
  400 FORMAT(2X,3HAP(,I5,1H,I5,1H,I5,3H)= G20,6)
      CONST1 = 0
         IF(MORDER-4) 199,200,201
  201 IF(MORDER-6)199,202,199
  200 DO 100 M = 1,MORDER
  100 CONST1= CONST1+AR(M)
      AC(1) = A(1,8) + CONST1
      CONST1 = 0
      DO 110 M = 1,4
```

```
  110 CONST1 = CONST1 + AR(M) * AP(1,6,M) + AS(M)
      AC(2) = A(2,8) + CONST1
      DO 1200 MS=3,7
      CONST1 = 0
      DO 120 M=1,4
       MS1 = MS-1
      MS2 = MS-2
      CONST1 = CONST1 + AR(M)*AP(MS1,6,M)+AS(M)*AP(MS2,6,M)
  120 CONTINUE
      AC(MS)    = A(MS,8) + CONST1
 1200 CONTINUE
      WRITE (NP,500) (I,AC(I),I=1,8)
  500 FORMAT(1H0, 1X,3HAC(,I5, 3H)= ,G20.6)
      CONST1 = 0
      DO 130 M = 1,4
      CONST1 = CONST1 + AS(M)*AP(6,6,M)
  130 CONTINUE
      AC(8) = A(8,8) + CONST1
      AK1 =DEXP(-PC(9))
      AK0 = 2.*F(9)*AK1
      AK(1) = AC(1) -AK1+AK0
      DO 150 MS= 2,8
      MS1 = MS-1
      AK(MS) =AC(MS) -AK1*AC(MS1) + AK0*A(MS1,8)
  150 CONTINUE
      AK(9) = AK0 * A(8,8) - AK1 * AC(8)
  550 FORMAT(3X,//,3X4HAK0=,G20.6,5X,4HAK1=,G20.6/)
  650 FORMAT(1H ,1X,/,2X,4HAK( ,I5,3H)= G20.6)
       WRITE(NP,650)( I,AK(I) ,I=1,9)
      IERR=0
      RETURN
  202 CONST1 = 0
      DO 205 M=1,MORDER
      CONST1 = CONST1 + AR(M)
  205 CONTINUE
      AK(1) = A(1,12) + CONST1
      CONST1 = 0
      DO 210 M = 1, MORDER
      CONST1 = CONST1 + AR(M)*AP(1,10,M)+AS(M)
  210 CONTINUE
      AK(2) = A(2,12) + CONST1
      DO 215 MS=3,11
      MS1 = MS-1
      MS2 = MS-2
      CONST1 = 0
      DO 212 M= 1,MORDER
      CONST1 = CONST1 + AR(M)*AP(MS1,10,M)+AS(M)*AP(MS2,10,M)
  212 CONTINUE
      AK(MS)    = A(MS,12) + CONST1
  215 CONTINUE
      CONST1 = 0
      DO 220 M=1,MORDER
      CONST1 = CONST1 + AS(M) * AP(10,10,M)
  220 CONTINUE
      AK(12) = A(12,12) + CONST1
      WRITE(NP,650) (I,AK(I) ,I=1,12)
      IERR= 0
      RETURN
  199 IERR=MORDER
      RETURN
      END
      SUBROUTINE TEA(NO,NP,ROOTR,ROOTI,XI,ETA)
```

```
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION ROOTR(13),ROOTI(13),XI(13),ETA(13)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      PI = 3.1415926
      PITWO = PI/2.
      PI2 = 2.*PI
      PIONE = 1./PI
      PINV=1./PI2
      DO 15 I=1,NO
      SQS = ROOTR(I)*ROOTR(I)+ROOTI(I)*ROOTI(I)
      XI(I) = -PINV*DLOG(SQS)
      IF(ROOTR(I))14,50,14
   14 GAMMA=DATAN(ROOTI(I)/ROOTR(I))
      ETA(I) = GAMMA
      IF(ROOTR(I)) 10,50,15
   50 IF(ROOTI(I)) 51,52,53
   51 ETA(I) =-PITWO
      GO TO 15
   52 K=K
   53 ETA(I) = PITWO
      GO TO 15
   10 IF(ROOTI(I)) 30,40,35
   40 ETA(I) = 3.1415926
      GO TO 15
   35 ETA(I) = ETA(I) + PI
      GO TO 15
   30 ETA(I)=ETA(I)-PI
   15 ETA(I) = -PIONE*ETA(I)
      RETURN
      END
      SUBROUTINE S6(OUT,SS,IN,K,KHAT,NP)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION K(12),KHAT(12),S(12),SS(12)
C    SUBROUTINE FOR DETERMINING THE COMMON POLYNOMIAL COEFFICIENTS
      REAL KHAT,K
      REAL KHAT1,KHAT2,KHAT3,KHAT4,KHAT5,KHAT6,KHAT7,KHAT8
     1, KHAT9,KHAT10,KHAT11,KHAT12,
     1 K1,K2,K3,K4,K5,K6,K7,K8,K9,K10,K11,MU1,MU1SQ,MUMU,
     2 L4,L3,L5,MU3,MU2,MUU,MUUU,K12,MU2SQ
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      NP=6
      KHAT1= KHAT(1)
```

```
      KHAT2=KHAT(2)
      KHAT3 = KHAT(3)
      KHAT4 = KHAT(4)
      KHAT5=KHAT(5)
      KHAT6=KHAT(6)
      KHAT7=KHAT(7)
      KHAT8=KHAT(8)
      KHAT9=KHAT(9)
      KHAT10=KHAT(10)
      KHAT11= KHAT(11)
      KHAT12=KHAT(12)
      K1= K(1)
      K2= K(2)
      K3 = K(3)
      K4 = K(4)
      K5 = K(5)
       K6 = K(6)
      K7 = K(7)
      K8=K(8)
      K9 = K(9)
      MU1 = K9 / KHAT12
      MU2 = (K8-KHAT11*MU1)/KHAT12
      MU2SQ=MU2*MU2
      MU3 = (K7-KHAT10*MU1-KHAT11*MU2)/KHAT12
        MU1SQ = MU1 * MU1
      MUMU = MU2/MU1SQ
      MUUU = MU2SQ/MU1SQ - MU3/MU1
      H11 = KHAT1 - K1
      H10 = KHAT2 - K2
       L4 = H10 -K1*H11
      A11    = K7 +K4/MU1 -K5*MUMU +(K6/MU1)*MUUU-KHAT7
      A12          = K7*H11 + K8 + K5/MU1 - K6*MUMU - KHAT8
      L5 = H11
      L3 = KHAT3 - K3 -K1*H10 + H11*(K1*K1 - K2)
      A13          = K7*(H10-K1*H11) + K8*H11 + K9 + K6/MU1 - KHAT9
      A21        =  K5 + K2/MU1 - K3*MUMU     +K4*MUUU/MU1 -KHAT5
      A22          = K5*H11 + K6 + K3/MU1 - K4*MUMU     - KHAT6
      A23 = K5*(H10-K1*H11)+K6*H11+K7+K4/MU1-KHAT7
      A31          = K3 + 1/MU1 - K1*MUMU+(K2*MUUU)/MU1 - KHAT3
      A32          = K3*H11 + K4 + K1/MU1 - K2*MUMU- KHAT4
      A33  =         K3*(H10-K1*H11) + K4*H11 + K5 + K2/MU1 - KHAT5
      B1    = KHAT10  -K7*L3 - K8 * L4 - K9*L5
        B2 = KHAT8-K8-K5*L3-K6*L4-K7*L5
      B3  =  KHAT6 - K6 - K3*L3 - K4*L4 - K5*L5
C     FROM CRAMERS RULE...
C     D1 = DELTA1/D,    D2 = DELTA2/D,    D3 = DELTA3/D
      A2233 = A22*A33 - A32 * A23
      A1233 = A12*A33 - A32*A13
      A1223 = A12*A23 - A22 *A13
      D = A11*A2233 - A21*A1233 + A31*A1223
       IF(D) 15,5,15
 1002 FORMAT(1H1,46H***DENOMINATOR D FOR CRAMERS RULE IS ZERO EOJ       )
    5 WRITE(NP,1002)
   15 CONTINUE
      DELTA1 = B1*A2233 - B2*A1233 + B3 *A1223
      A2133 = A21*A33 - A31*A23
      A1133 = A11*A33 - A31*A13
      A1123 = A11*A23 - A21*A13
      DELTA2 = -B1*A2133 + B2 * A1133 - B3 *A1123
      A2132 = A21*A32 - A31*A22
      A1132 = A11 * A32 - A31*A12
      A1122 = A11*A22 - A21*A12
```

```
      DELTA3 = B1 *A2132  - B2*A1132 + B3 *A1122
      D1 = DELTA1/D
      D2 = DELTA2/D
      D3 = DELTA3/D
C   THE  COEFFICIENTS  THEN ARE GIVEN BY ...
      SS(1)=K1-D3
      SS(2) = K2-D2-D3*SS(1)
      SS(3) = K3-D1-D2*SS(1)-D3*SS(2)
      SS(4)=K4 - D1*SS(1) - D2 *SS(2)-D3*SS(3)
      SS(5) = K5 - D1*SS(2) - D2*SS(3)-D3 *SS(4)
      SS(6) = K6 -D1*SS(3)-D2*SS(4)-D3*SS(5)
      RETURN
      END
      SUBROUTINE PAT(A,B,X,Z,X9,X12,X6,D,E,E9,E12,E6,G,MU,NF,RR9,RI9,
     1 RR12,RI12,RR6,RI6,ND9,ND12,Y9,Y12,PIROLD,PIRNEW,PIIOLD,PIINEW,
     2                 S,T,NDI9,NDI12,YI9,YI12)
C     IMPLICIT REAL*8(A-H,O-Z)
      REAL MU
C   SUBROUTINE FOR THE GENERATION OF THE SUMMARY TABLES FOR THE NASA FLU
      DIMENSION                              S(8,14),      T(11,14)
      DIMENSION                        A(8,14),B(11,14)
      DIMENSION NDI12(13),PIIOLD(13),PIINEW(13),NDI9(10),YI9(10)
      DIMENSION YI12(13),PIROLD(13),PIRNEW(13),X9(13),X12(13),X6(7)
      DIMENSION D(013),E9(013),E12(013),E6(013),RR9(13),RI9(13)
        DIMENSION RR12(13),RI12(13),RR6(7),RI6(7),Y9(10),E(013),X(013)
      DIMENSION Z(013),ND9(10),ND12(13),Y12(13)
      DSQRT(D)=SQRT(D)
      DMAX1(A,B,C)=AMAX1(A,B,C)
      DABS(D)=ABS(D)
      DATAN(D)=ATAN(D)
      DCOS(D)=COS(D)
      DEXP(D)=EXP(D)
      DLOG(D)=ALOG(D)
      DLOG10(D)=ALOG10(D)
      DSIN(D)=SIN(D)
      DCOSH(D)=COSH(D)
      DSINH(D)=SINH(D)
      ND=6
      NP=6
      WRITE(NP,50)           NF, (X(J),D(J),Y9(J),YI9(J) ,J=1,9), (Z(J),
     1 E(J),Y12(J),YI12(J ),J=1,12)
      WRITE(NP,54)
      DO 25 J = 1,8
      L = ND9(J)
      L=3
      WRITE(ND,51)   Y9(J),A(J,L+1),A(J,L-2),A(J,L-1),A(J,L),ND9(J) ,
     1 PIRNEW(J)
      WRITE(ND,551)  YI9(J),S(J,L+1),S(J,L-2),S(J,L-1),S(J,L),ND9(J) ,
     1 PIINEW(J)
   25 CONTINUE
   54 FORMAT(55X,24HNINTH ORDER SYSTEM             /)
   55 FORMAT(55X,24HTWELFTH ORDER SYSTEM            /)
      WRITE(NP,55)
      DO 20 M=1,11
      L=ND12(M)
      L=3
      JM8= M
      WRITE(ND,51)Y12(JM8),B(JM8,L+1),B(JM8,L-2),B(JM8,L-1),
     1 B(JM8,L),ND12(M) ,PIROLD(JM8)
      WRITE(ND,551)YI12(JM8),T(JM8,L+1),T(JM8,L-2),T(JM8,L-1),
     1T(JM8,L),ND12(M) ,PIIOLD(JM8)
   20 CONTINUE
```

```
      WRITE(NP,53) (X9(J),E9(J),RR9(J),RI9(J ),J=1,9),( X12(J),E12(J),
     1RR12(J),RI12(J) ,J=1,12), (X6(J),E6(J),RR6(J),RI6(J) ,J=1,6)
   50 FORMAT(1H1,38X,55HSTABILITY OF DYNAMIC SYSTEMS WITH PERIODIC PARAM
     1METERS                    /
     2  /49X,40H
     3/55X,7H              /55X,7H              /55X,7H    NF = I5/
     452X,40HSINGULARITIES AND EVALUATION POINTS           /
     5 30X, *          XI                   ETA                  YR
     2                 YI*                        /             55X*NI
     6NTH ORDER SYSTEM* /9(30X,4G20.8/)/55X,*TWELFTH ORDER SYSTEM *
     4/12(30X,4G20.8/)/60X,18HDETERMINANT VALUES
     5/15X ,1HY,18X,7HD, EST,14X,7HDELTA 1,14X,7HDELTA 2
     6 14X,7HDELTA 3,10X,7HND  MAX 10H      P             /)
   51 FORMAT(1X,*REAL*5G20.8,I12,F10.5)
  551 FORMAT(1X,*IMAG*5G20.8,2X,I10,F10.5)
   52 FORMAT(1X,5G20.8,2X,I15,F10.5)
   53 FORMAT(*1*/55X,21HCHARACTERISTIC VALUES//55X,24HNINTH ORDER SYSTEM
     2               //19X,2HXI,19X,3HETA,9X,18HREAL PART OF ROOT  ,3X,
     4 19HIMAG, PART OF ROOT          /, 9(10X,4G20.8/),
     3/,55X,24HTWELFTH ORDER SYSTEM
     4//19X,2HXI,19X,3HETA9X,18HREAL PART OF ROOT ,3X,
     519HIMAG, PART OF ROOT              /12(10X,4G20.8/),
     6/55X,24HSIXTH ORDER SYSTEM
     7//19X,2HXI,19X,3HETA9X,18HREAL PART OF ROOT ,3X,
     819HIMAG, PART OF ROOT              /6(10X,4G20.8/),/1H1)
      RETURN
      END
```

These programs also use the standard IBM subroutine DPRQD, as given in IBM System /360 Scientific Subroutine Package (360A-CM-03X) Version III Programmer's Manual, IBM publication H20-0205-3, Fourth Edition.

```
C     PROGRAM TO COMPUTE A'S AND B'S
      REAL    MU,MB,KM,MT,MODFR,MBAR
C     REAL*8 MU,MB,KM,MT,MODFR,MBAR
      INTEGER OUT
      DIMENSION AR(3,3,12),AI(3,3,12),BR(3,3,12),BI(3,3,12)
      DIMENSION XIBARO(17)
      DIMENSION                ZBARA(17),CBAR(17),PSI(17)
      DIMENSION MODFR(3)
      DIMENSION PHI(17),XLBARZ(17),MBAR(17),EBAR(17)
      COMMON COSAS,COSQAS,RMUSQ,MU,AS,CT,CTSQ,SINAS,SINETA,RMUMWB,
     1 XMUCSE,SINZA,COSZA,RMUCSA
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
      COMMON /A2/ CIJR,CIJI,DIJR,DIJI,FAC,NFTW
      COMMON /A4/ IN,OUT,MB,R,KM,MT,MODFR,NR,NF,    XIBARO,CBAR,ZBARA,
     1 PHI,XLBARZ,MBAR,EBAR,PI,ETA,ETA1
      COMMON /A5/ XBAR(17)
      COMMON /A6/ TX(17,24,16),NSW
 9876 FORMAT ('1'/(' ',09G12.4))
      IN=5
      IN1=5
      IN2=5
      OUT=6
      NSW=0
    1 FORMAT (10(D10.7/),2(I5/))
      READ (IN,1) MU,AS,MB,CT,R,KM,MT,MODFR,NR,NF
      AS=AS*.0174532
      READ (IN1,9874)     (I,XBAR(I),MBAR(I),XIBARO(I),CBAR(I),EBAR(I),
     1 XLBARZ(I),ZBARA(I),PHI(I),J=1,NR)
      READ (IN2,9874)     (I,ANU(I,1),ANU(I,2),ANU(I,3),AW(I,1),
     1 AW(I,2),AW(I,3),APHI(I,1),APHI(I,2),J=1,NR)
      READ (IN2,9874)     (I,APHI(I,3),ATH(I,1),ATH(I,2),ATH(I,3),
     1 APSI(I,1),APSI(I,2),APSI(I,3),X,J=1,NR)
      READ (IN2,9873) ((APHI(I,J),ANU(I,J),ATH(I,J),I=1,NR),J=1,3)
 9873 FORMAT (9X,E9.3,1X,E9.3,1X,E9.3)
 9874 FORMAT (6X,I2,8E8.7)
      DO 5 I=1,NR
      ANU(I,1)=ANU(I,1)/R
      ANU(I,2)=ANU(I,2)/R
      ANU(I,3)=ANU(I,3)/R
    5 PHI(I)=PHI(I)*.0174532
      PI=3.141592
      COSAS= COS(AS)
      COSQAS=COSAS*COSAS
      SINAS= SIN(AS)
      CTSQ=CT*CT
      RMUCSA=MU*COSAS
      RMUSQ=MU*MU
      RMUSNA=MU*SINAS
      RMUCA4=RMUCSA*RMUCSA
      RMUCA4=RMUCA4*RMUCA4
      IF (CT) 6,7,6
    6 WBARI= SQRT(.5*( SQRT(RMUCA4+CTSQ)-RMUSQ*COSQAS))
      GO TO 8
    7 WBARI=0.
    8 RMUMWB=RMUSNA-WBARI
      ETA=PI/NF
      RSQ=R*R
      NFTW=2*NF
      XMMBRQ=-MB*RSQ/NF
```

```
      TWMBRQ= 2.*(-XMMBRQ)
      WRITE (OUT,9876) MU,AS,MB,CT,R,KM,MT,MODFR,NR,NF
      WRITE (OUT,9876)   (I,XBAR(I),MBAR(I),XIBARO(I),CBAR(I),EBAR(I),
     1 XLBARZ(I),ZBARA(I),PHI(I),I=1,NR)
      WRITE (OUT,9876)   (I,ANU(I,1),ANU(I,2),ANU(I,3),AW(I,1),
     1 AW(I,2),AW(I,3),APHI(I,1),APHI(I,2),I=1,NR)
      WRITE (OUT,9876)   (I,APHI(I,3),ATH(I,1),ATH(I,2),ATH(I,3),
     1 APSI(I,1),APSI(I,2),APSI(I,3),X,I=1,NR)
      DO 100 I=1,3
      FOURWQ=4.*MODFR(I)*MODFR(I)
      DO 100 J=1,3
      DO 100 K=1,NF
      FAC=ETA*(K-1)
      CALL CDIJ (I,J,K)
      AR(I,J,K)=XMMBRQ*DIJR
      AI(I,J,K)=-XMMBRQ*DIJI
      BI(I,J,K)= TWMBRQ*CIJI
      IF (K.EQ.1.AND.J.EQ.I) GO TO 10
      BR(I,J,K)=-TWMBRQ*CIJR
      GO TO 100
   10 BR(I,J,K)=FOURWQ-TWMBRQ*CIJR
  100 CONTINUE
      DO 200 I=1,3
      DO 200 J=1,3
      DO 200 K=1,NF
      ARR=AR(I,J,K)
      AII=AI(I,J,K)
      BRR=BR(I,J,K)
      BII=BI(I,J,K)
      WRITE (8) I,J,K,ARR,AII,BRR,BII
      WRITE (7,9779) ARR,AII,BRR,BII
 9779 FORMAT(4E14.7)
  200 WRITE (6,9875)   I,J,K,AR(I,J,K),AI(I,J,K),BR(I,J,K),BI(I,J,K)
 9875 FORMAT ('0'/(' ',           I5,1X,I5,1X,I5,4X,G12.5,1X,G12.5,
     1 1X,G12.5,1X,G12.5))
      END FILE 8
      STOP
      END
      SUBROUTINE CDIJ (L,K,M)
C     IMPLICIT REAL*8(A-H,O-Z)
C     REAL*8 MU,MB,KM,MT,MODFR,MBAR
      REAL   MU,MB,KM,MT,MODFR,MBAR
      INTEGER OUT
      DIMENSION XIBARO(17)
      DIMENSION                 ZBARA(17),CBAR(17),PSI(17)
      DIMENSION MODFR(3)
      DIMENSION PHI(17),XLBARZ(17),MBAR(17),EBAR(17)
      DIMENSION XINTER(17),XINTR1(17),Z(17)
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
      COMMON /A2/ CIJR,CIJI,DIJR,DIJI,FAC,NFTW
      COMMON /A4/ IN,OUT,MB,R,KM,MT,MODFR,NR,NF,   XIBARO,CBAR,ZBARA,
     1 PHI,XLBARZ,MBAR,EBAR,PI,ETA,ETA1
      COMMON /A5/ XBAR(17)
      COMMON /A6/ TX(17,24,16),NSW
      CIJR=0.0
      CIJI=0.0
      DIJR=0.0
      DIJI=0.0
```

```
      DO 200 J=1,NFTW
      XAC=J-1
      ETA1=ETA*XAC
      XAC=FAC*XAC
      SINFAC= SIN(XAC)
      COSFAC= COS(XAC)
      DO 100 I=1,NR
      IF (NSW.NE.0) GO TO 50
      CALL VECTOR (I)
      TX (I,J,1) =TERM(2)
      TX (I,J,2) =TERM(3)
      TX (I,J,3) =TERM(4)
      TX (I,J,4) =TERM(5)
      TX (I,J,5) =TERM(6)
      TX (I,J,6) =TERM(8)
      TX (I,J,7) =TERM(9)
      TX (I,J,8) =TERM(10)
      TX (I,J,9) =TERM(11)
      TX (I,J,10)=TERM(12)
      TX (I,J,11)=TERM(29)
      TX (I,J,12)=TERM(39)
      TX (I,J,13)=TERM(40)
      TX (I,J,14)=TERM(42)
      TX (I,J,15)=TERM(43)
      TX (I,J,16)=TERM(44)
 9876 FORMAT (' '/(' ',10G12.4))
C     XINTER(I)=ANU(I,L)*XMUNU(I,K)+AW(I,L)*XMUWJ(I,K)+APHI(I,L)*
C    1 XMUPHJ(I,K)+APSI(I,L)*XMUPSI(I,K)
   50 XINTER(I)=ANU(I,L)*XMUNU(I,J,K)+APHI(I,L)*XMUPHJ(I,J,K)
C 100 XINTR1(I)=ANU(I,L)*XLMBNU(I,K)+AW(I,L)*XLAMWJ(I,K)+APHI(I,L)*
C    1 XLMPHJ(I,K)+APSI(I,L)*XLMPSJ(I,K)
  100 XINTR1(I)=ANU(I,L)*XLMBNU(I,J,K)+APHI(I,L)*XLMPHJ(I,J,K)
      NRPTS=NR
      CALL QTFG (XINTER,Z,NRPTS)
      CIJI=SINFAC*Z(NR)+CIJI
      IF (M.EQ.1.AND.L.EQ.K) GO TO 120
      CIJR=CIJR+COSFAC*Z(NR)
      GO TO 150
  120 CIJR=CIJR+Z(NR)
  150 CALL QTFG(XINTR1,Z,NRPTS)
      DIJI=SINFAC*Z(NR)+DIJI
  200 DIJR=COSFAC*Z(NR)+DIJR
      NSW=1
      RETURN
      END
      SUBROUTINE VECTOR (I)
C     IMPLICIT REAL*8(A-H,O-Z)
C     REAL*8 MU,MB,KM,MT,MODFR,MBAR
      REAL   MU,MB,KM,MT,MODFR,MBAR
      INTEGER OUT
      DIMENSION XIBARO(17)
      DIMENSION                 ZBARA(17),CBAR(17),PSI(17)
      DIMENSION MODFR(3)
      DIMENSION PHI(17),XLBARZ(17),MBAR(17),EBAR(17)
      COMMON COSAS,COSQAS,RMUSQ,MU,AS,CT,CTSQ,SINAS,SINETA,RMUMWB,
     1 XMUCSE,SINZA,COSZA,RMUCSA
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
      COMMON /A4/ IN,OUT,MB,R,KM,MT,MODFR,NR,NF,   XIBARO,CBAR,ZBARA,
```

```
     1 PHI,XLBARZ,MBAR,EBAR,PI,ETA1,ETA
      COMMON /A5/ XBAR(17)
 9876 FORMAT ('0',G15.7)
      COSETA= COS(ETA)
      SINETA= SIN(ETA)
      AO=PHI(I)+ZA(ETA,I)
      XBMSEA=XBAR(I)+MU*SINETA
      CALL SERIES (I,J,NCODE,MT,XBMSEA,V,AO,CL,ASLOP,CM,CD,CMA,CDA)
      SINAO= SIN(AO)
      COSAO= COS(AO)
      CLCDA=CL-CDA
      CDCLA=CD+ASLOP
      SIGD=CLCDA*SINAO-CDCLA*COSAO
      SIGL=CL*COSAO+CD*SINAO
      VO= SQRT(RMUMWB*RMUMWB+XMUCSE*XMUCSE)
      GAML=CLCDA*COSAO+CDCLA*SINAO
      GAMD=CL*SINAO-CD*COSAO
      DELTA=CBAR(I)*CM-ZBARA(I)*CL
      DELTAP=CBAR(I)*CMA-ZBARA(I)*ASLOP
      PICBAR=PI*CBAR(I)
      SIGL2=2.*SIGL
      ZBAMLZ=ZBARA(I)-XLBARZ(I)
      COSASE=COSAS*COSETA
      VOKM=KM*VO
      OCKV=CBAR(I)*VOKM
      TERM(10)=.5*OCKV
      TERM(1)=TERM(10)*VO
      TERM(2)=TERM(1)*SIGD
      TERM(3)= COS(PHI(I))
      TERM(4)= SIN(PHI(I))
      TERM(5)=SIGD*(MU*COSZA*COSASE-ZBAMLZ*COSAO)+SIGL2*(ZBAMLZ*
     1 SINAO-MU*SINZA*COSASE)+PICBAR/2.*COSAO
      TERM(6)=-2.*MBAR(I)*EBAR(I)*TERM(4)
      TERM(7)=PI/4.*OCKV
      TERM(8)=TERM(7)*COSAO*CBAR(I)
      TERM(9)=TERM(10)*(SIGD*COSAO-SIGL2*SINAO)
      TERM(11)=ZBARA(I)
      TERM(12)=TERM(10)*(SIGD*SINAO+SIGL2*COSAO)
      TERM(13)=TERM(1)*GAML
      TERM(14)=MU*COSASE
      TERM(15)=TERM(14)*COSZA
      TERM(16)=ZBAMLZ*COSAO
      TERM(17)=TERM(15)-TERM(16)
      TERM(18)=GAML*TERM(17)
      TERM(19)=TERM(14)*SINZA
      TERM(20)=ZBAMLZ*SINAO
      TERM(21)=2.0*GAMD*(TERM(19)-TERM(20))
      TERM(22)=PI/2.0*CBAR(I)
      TERM(23)=TERM(22)*SINAO
      TERM(24)=TERM(18)+TERM(21)-TERM(23)
      TERM(25)=2.0*MBAR(I)*EBAR(I)
      TERM(26)=-TERM(25)*TERM(3)
      TERM(27)=OCKV*CBAR(I)*PI/4.0
      TERM(28)=TERM(27)*SINAO
      TERM(29)=TERM(27)*ZBARA(I)
      TERM(30)=TERM(10)*(GAML*COSAO+2.*GAMD*SINAO)
      TERM(31)=TERM(10)*(-GAML*SINAO+2.*GAMD*COSAO)
      TERM(32)=2.0*DELTA
      TERM(33)=DELTAP*COSZA
```

```
      TERM(34)=TERM(32)*SINZA
      TERM(35)=DELTAP*COSAO
      TERM(36)=TERM(32)*SINAO
      TERM(37)=TERM(33)+TERM(34)
      TERM(38)=TERM(35)+TERM(36)
      TERM(39)=TERM(1)*DELTAP
      TERM(40)=TERM(10)*(TERM(14)*TERM(37)-ZBAMLZ*TERM(38) +TERM(22)*
     1 ZBARA(I))
      TERM(41)=2.0*XIBARO(I)
      TERM(42)=TERM(41)*TERM(4)
      TERM(43)=TERM(10)*TERM(38)
      TERM(44)=TERM(10)*(-DELTAP*SINAO+TERM(32)*COSAO)
      RETURN
      END
      SUBROUTINE QTFG (Y,Z,NDIM)
C     IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION Y(1),Z(1)
      COMMON /A5/ X(17)
C
C        INTEGRATION BY GENERAL TRAPEZOIDAL RULE
C
      SUM2=0.
      IF (NDIM-1) 4,3,1
    1 DO 2 I=2,NDIM
      SUM1=SUM2
      SUM2=SUM2+0.5*(X(I)-X(I-1))*(Y(I)+Y(I-1))
    2 Z(I-1)=SUM1
    3 Z(NDIM)=SUM2
    4 RETURN
      END
      FUNCTION ZA(ETA,I)
C     IMPLICIT REAL*8(A-H,O-Z)
C     REAL*8 MU,MB,KM,MT,MODFR,MBAR
      REAL   MU,MB,KM,MT,MODFR,MBAR
      COMMON COSAS,COSQAS,RMUSQ,MU,AS,CT,CTSQ,SINAS,SINETA,RMUMWB,
     1 XMUCSE,SINZA,COSZA,RMUCSA
      COMMON /A5/ XBAR(17)
      XMUCSE=XBAR(I)+RMUCSA*SINETA
      IF (RMUMWB.EQ.0..AND.XMUCSE.EQ.0.) GO TO 2
      ZA= ATAN2(RMUMWB,XMUCSE)
      GO TO 3
    2 ZA =0.
    3 SINZA= SIN(ZA)
      COSZA= COS(ZA)
      RETURN
      END
      FUNCTION XMUNU (I,J,L)
C     IMPLICIT REAL*8(A-H,O-Z)
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
      COMMON /A6/ TX(17,24,16),NSW
      XMUNU=TX(I,J,1)*APHI(I,L)+TX(I,J,8)*(TX(I,J,2)*ATH(I,L)+
     1 TX(I,J,3)*APSI(I,L))*TX(I,J,4)
      RETURN
      END
      FUNCTION XLMBNU (I,J,L)
C     IMPLICIT REAL*8(A-H,O-Z)
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
```

```
      COMMON /A6/ TX(17,24,16),NSW
      XLMBNU=TX(I,J,5)*APSI(I,L)-TX(I,J,6)*APHI(I,L)+TX(I,J,7)*
     1 (TX(I,J,9)*APHI(I,L)+ANU(I,L))-TX(I,J,10)*AW(I,L)
      RETURN
      END
      FUNCTION XMUPHJ (I,J,L)
C     IMPLICIT REAL*8(A-H,O-Z)
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
      COMMON /A6/ TX(17,24,16),NSW
      XMUPHJ=TX(I,J,12)*APHI(I,L)+TX(I,J,13)*(ATH(I,L)*TX(I,J,2)+
     1APSI(I,L)*TX(I,J,3))
      RETURN
      END
      FUNCTION XLMPHJ (I,J,L)
C     IMPLICIT REAL*8(A-H,O-Z)
      COMMON /A1/ TERM(50),ANU(17,3),AW(17,3),APHI(17,3),ATH(17,3),
     1 APSI(17,3)
      COMMON /A6/ TX(17,24,16),NSW
      XLMPHJ=TX(I,J,14)*APSI(I,L)-TX(I,J,11)*APHI(I,L)+TX(I,J,15)*
     1 (TX(I,J,9)*APHI(I,L)+ANU(I,L))+TX(I,J,16)*AW(I,L)
      RETURN
      END
      SUBROUTINE SERIES(I,J,NCODE,EMT,U,V,APHIJ,CLIFT,ASLOP,CMOME,CDRAG,
     1 CMA,CDA)
C     IMPLICIT REAL*8(A-H,O-Z)
C
C
C     SUBROUTINE TO COMPUTE CLIFT=LIFT COEFFICIENT
C
C
C                           ASLOP=LIFT CURVE SLOPE
C                           CMOME=MOMENT COEFFICIENT
C                           CDRAG=DRAG COEFFICIENT
C
      CDA=0.
      CMA=0.
      CLIFT=0.
      ASLOP=0.
      CMOME=0.
      CDRAG=0.
C
C
      TWPI=2.*3.1415926
      IF (APHIJ.LT.-TWPI) STOP 100
      IF (APHIJ.GT.2.*3.1415926) STOP 101
  180 NEG=1
      EMIJ=EMT* ABS(U)
      SQT= SQRT(1.-EMIJ*EMIJ)
      C1=1.-EMIJ
      C2=.22689*C1
  172 IF (APHIJ+3.1415926) 173,181,182
  173 APHIJ=APHIJ+3.1415926*2.
      GO TO 186
  182 IF(APHIJ-3.1415926) 184,186,183
  183 APHIJ=APHIJ-3.1415926*2.
      GO TO 181
  184 IF(APHIJ) 181,186,186
  181 APHIJ=-APHIJ
```

```
      NEG=-1
  186 IF(APHIJ-C2) 185,187,187
  185 ASLOP=5.7296/SQT
      CLIFT=ASLOP*APHIJ
      CDRAG=.006+.13131*APHIJ*APHIJ
      CMOME=1.4324*APHIJ/SQT
      CDA=.26262*APHIJ
      CMA=1.4324/SQT
      GO TO 250
  187 IF(APHIJ-.34906) 189,191,191
  189 CLIFT=.29269*C1+(1.3*EMIJ-.59)*APHIJ
      CMOME=CLIFT/(SQT*(.48868+.90756*EMIJ))
      C2=(.12217+.22689*EMIJ)*SQT
      CLIFT=CLIFT/C2
      ASLOP=(1.3*EMIJ-.59)/C2
      CMA=(1.3*EMIJ-.59)/((.48868+.90756*EMIJ)*SQT)
      GO TO 210
  191 IF(APHIJ-2.7402) 193,195,195
  193 S= SIN(APHIJ)
      S2= SIN(2.*APHIJ)
      S3= SIN(3.*APHIJ)
      S4= SIN(4.*APHIJ)
      CLIFT=(.080373*S+1.04308*S2-.011059*S3+.023127*S4)/SQT
      CMOME=(-.02827*S+.14022*S2-.00622*S3+.01012*S4)/SQT
      C= COS(APHIJ)
      C2= COS(2.*APHIJ)
      C3= COS(3.*APHIJ)
      C4= COS(4.*APHIJ)
      ASLOP=(.080373*C+2.08616*C2-.033177*C3+.092508*C4)/SQT
      CDRAG=(1.1233-.C29894*C-1.00603*C2+.003115*C3-.091487*C4)/SQT
      CMA=(-.02827*C+.28044*C2-.01866*C3+.04048*C4)/SQT
      GO TO 240
  195 IF(APHIJ-3.0020) 197,199,199
  197 CLIFT=-(.4704+.10313*APHIJ)/SQT
      ASLOP=-.10313/SQT
      CMOME=-(.4786+.02578*APHIJ)/SQT
      CMA=-.02578/SQT
      GO TO 210
  199 IF(APHIJ-3.1415926) 200,200,260
  200 CLIFT=(-17.550+5.5864*APHIJ)/SQT
      ASLOP=5.5864/SQT
      CMOME=(-12.5109+3.9824*APHIJ)/SQT
      CMA=3.9824/SQT
  210 C= COS(APHIJ)
      C2= COS(2.*APHIJ)
      C3= COS(3.*APHIJ)
      C4= COS(4.*APHIJ)
      CDRAG=(1.1233-.C29894*C          -1.00603*C2
     1       +.003115*C3           -.091487*C4           )/SQT
  240 S= SIN(APHIJ)
      S2= SIN(2.*APHIJ)
      S3= SIN(3.*APHIJ)
      S4= SIN(4.*APHIJ)
      CDA=(.029894*S          +2.01206*S2           -.009345*S3
     1       +.36595*S4          )/SQT
  250 IF(NEG) 255,255,260
  255 CLIFT=-CLIFT
      CMOME=-CMOME
      APHIJ=-APHIJ
```

```
      CDA=-CDA
  260 CONTINUE
C
300   CONTINUE
      RETURN
      END
```

APPENDIX B

Listing of Computer Program for Determining Characteristic Modal Functions

```
C
      COMPLEX T,TS,C,CMPLX,TSRET,DET,XLOG,A,B,DUM,ZERO,STLAM,EIGENC
      DIMENSION A(3,3,12),B(3,3,12)
      DIMENSION DUM(216),EVALUE(41)
      DIMENSION NTIT(20)
      DIMENSION T(41,41),TS(41,41),C(41),TSRET(41,41)
      EQUIVALENCE (A(1,1,1),DUM(1)),(B(1,1,1),DUM(109))
      COMMON DUM,EIGENC,N,NP
      IROW(N,NR)=3*N+NR
      ICOL(N,NC)=3*N+NC
C
C        READ PROGRAM CONTROL CONSTANTS
C
      READ (5,9991) NTIT
      WRITE (6,9992) NTIT
      READ(5,2) N,NS,NQ,NLAMB
C
C        PROGRAM CONSTANTS
C
      ZERO=CMPLX(0.,0.)
      ONE = CMPLX(1.,0.)
      DO 199 I=1,216
  199 DUM(I)=ZERO
      NP=2*N+1
      NAUSEC = 3*(2*N+1)
      NAUSER = 3*(2*N+1)
      NBUSEC=NAUSEC-1
      NBUSER=NAUSER-1
      NADIC = 41
      NADIR = 41
      NBDIC = 41
      NBDIR = 41
      THREEN=3*N
      KTH=ICOL(N,NQ)
      LTH=IROW(N,NS)
      LTHM1=LTH-1
      LTHP1=LTH+1
C
C        INPUT PROGRAM VARIABLES
C
      READ (5,9871) (((A(I,J,K),B(I,J,K),K=1,NP),J=1,3),I=1,3)
      WRITE(6,9874)  (((I,J,K,A(I,J,K),B(I;J,K),K=1,NP),J=1,3),I=1,3)
C
C        ITERATE ON THE NUMBER OF ROOTS, NLAMB
C
      DO 1000 IZZ=1,NLAMB
      EVALUE(IZZ)=ZERO
      READ (5,9877) EIGENC
      WRITE (6,9882) EIGENC
C
C        DEFINE A NEW T ARRAY AND C VECTOR
C
```

```
      CALL MAKET (T)
      WRITE (6,9886) ((T(M,II),II=1,NAUSEC),M=1,NAUSER)
 9886 FORMAT (/'OT MATRIX'/('0',2E16.7,5X,2E16.7,5X,2E16.7))
      WRITE (6,9881) LTH
      CALL REMV (T,TS,LTH,KTH,NAUSEC,NAUSER,NADIC,NADIR,NBDIC,NBDIR)
      IF (LTH.EQ.1) GO TO 12
      DO 10 II =1,LTHM1
   10 C(II)=-T(II,KTH)
      IF (LTH.EQ.NAUSER) GO TO 16
   12 DO 14 II=LTHP1,NAUSER
      M=II-1
   14 C(M)=-T(II,KTH)
   16 WRITE (6,9875) (C(M),M=1,NBUSER)
C
C         SOLVE THE RESULTING EQUATIONS FOR EPS
C
      CALL CMAT(TS,NBUSER,C,DET)
      WRITE (6,9875) (C(M),M=1,NBUSER)
      WORK1=REAL(DET)
      WORK2=AIMAG(DET)
      IF (WORK1.EQ.0..AND.WORK2.EQ.0.) WRITE (6,9876)
 1000 CONTINUE
      STOP
C
C
C
    2 FORMAT(8I2,2E14.7)
 9871 FORMAT (4E14.7)
 9872 FORMAT (2E14.7)
 9873 FORMAT (1H1/(1H0,I5,2E16.7))
 9874 FORMAT ('0',3I5,4E16.7)
 9875 FORMAT ('1'/('0',2E16.7))
 9876 FORMAT ('0','VALUE OF DET IS ZERO')
 9877 FORMAT (2E13.7)
 9881 FORMAT (/'OEXTRACTED ROW ',I3)
 9882 FORMAT (/'OTHE STARTING EIGENVALUE IS ',2E16.7)
 9991 FORMAT (20A4)
 9992 FORMAT ('1',20X,20A4)
      END
      SUBROUTINE MAKET (T)
      COMPLEX A,B,C,CLAM,CMPLX,CONJG,T,CZ,CY
      DIMENSION T(41,41),A(3,3,12),B(3,3,12)
      COMMON A,B,CLAM,N,NP
C
C         CLAM IS THE EIGENVALUE
C
      NADD1=N+1
      C=CMPLX(0.,2.)
      DO 20 M=1,NP
      C2=M-NADD1
      CY=CLAM+C*CMPLX(C2,0.)
      IM=(M-1)*3 +1
      IMAD2=IM+3 -1
      DO 20 K=1,NP
```

```
      C1=K-NADD1
      CZ=CLAM+C*CMPLX(C1,0.)
      IK=(K-1)*3 +1
      IKAD2=IK+3 -1
      IF(K-M)25,30,35
   25 LZ=M-K+1
      DO 27 II = IM,IMAD2
      IMM=MOD(II,3 )
      IF(IMM.EQ.0)IMM=3
      DO 27 JJ= IK,IKAD2
      IKK=MOD(JJ,3 )
      IF(IKK.EQ.0)IKK=3
   27 T(II,JJ)=CZ*A(IMM,IKK,LZ)+B(IMM,IKK,LZ)
      GO TO 20
   30 DO 32 I=IM,IMAD2
      IMM=MOD(I,3 )
      IF(IMM.EQ.0)IMM=3
      DO 32 J=IK,IKAD2
      IKK=MOD(J,3 )
      IF(IKK.EQ.0)IKK=3
   32 T(I,J)=CY*A(IMM,IKK,1)+B(IMM,IKK,1)
      DO 34 NZ=IM,IMAD2
   34 T(NZ,NZ)=T(NZ,NZ)+CY*CY
      GO TO 20
   35 LS=K-M+1
      DO 38 IX=IM,IMAD2
      IMM=MOD(IX,3 )
      IF(IMM.EQ.0)IMM=3
      DO 38 JX=IK,IKAD2
      IKK=MOD(JX,3 )
      IF(IKK.EQ.0)IKK=3
   38 T(IX,JX)=CZ*CONJG(A(IMM,IKK,LS))+CONJG(B(IMM,IKK,LS))
   20 CONTINUE
      RETURN
      END
      SUBROUTINE REMV (A,B,I,J,NAUSEC,NAUSER,NADIC,NADIR,NBDIC,NBDIR)
      COMPLEX*8   A(NADIR,NADIC),B(NBDIR,NBDIC)
      I1=0
      DO 2 II=1,NAUSER
      IF(II.EQ.I) GO TO 2
      I1=I1+1
      J1=0
      DO 1 JJ=1,NAUSEC
      IF (JJ.EQ.J) GO TO 1
      J1=J1+1
      B(I1,J1)=A(II,JJ)
    1 CONTINUE
    2 CONTINUE
      RETURN
      END
      SUBROUTINE CMAT(A,N,Y,DET)
C     CMAT SUBROUTINE FILE NUMBER 310.4.505
C     UNIVERSITY OF ROCHESTER COMPUTING CENTER
C     A BECOMES AINVERSE, YOFAX=Y BECOMES X, DET IS DET A
C     MATEQ SOLVES AX=Y FOR X, COMPUTES A INVERSE, AND CALCULATES THE
C     DETERMINANT USING A VARIATION OF GAUSSIAN ELIMINATION.
C     P J EBERLEIN, REVISED BY C TEAGUE FOR /360
C     USER SHOULD CHECK DET IMMEDIATELY FOR SINGULARITY
C     KEYPUNCH IS 029
C     CMAT  COMPLEX*8 VERSION OF MATEQ /360 FILE NO. 310.4.500
C     A = COMPLEX*8, MATRIX TO BE INVERTED
C     N = INTEGER*4, ACTUAL SIZE OF MATRIX A
```

```
C      Y = COMPLEX*8, VECTCR TO SOLV AX=Y
C      DET = COMPLEX*8, VARIABLE FOR DETERMINANT OF A
       IMPLICIT COMPLEX (A-H,O-Z)
       REAL*4 CABS
       DIMENSION A(41,41),Y(41)
       DIMENSION ICHG(41)
       DET=1.0
       DO 118 K=1,N
       AMX = A(K,K)
       IMX=K
       DO 100 I=K,N
       IF(CABS(A(I,K)).LE.CABS(AMX)) GO TO 100
       AMX = A(I,K)
       IMX=I
   100 CONTINUE
       IF(CABS(AMX).GT.0.1E-70) GO TO 102
       DET=0.0
       GO TC 124
   102 IF (IMX.EQ.K) GO TO 106
       DO 104 J=1,N
       TEMP=A(K,J)
       A(K,J)=A(IMX,J)
   104 A(IMX,J)=TEMP
       ICHG(K)=IMX
       TEMP=Y(K)
       Y(K)= Y(IMX)
       Y(IMX)= TEMP
       DET=-DET
       GO TO 108
   106 ICHG(K)=K
   108 DET=DET*A(K,K)
       A(K,K)=1.0/A(K,K)
       DO 110 J=1,N
       IF (J.NE.K) A(K,J)=A(K,J)*A(K,K)
   110 CONTINUE
       Y(K) = Y(K)*A(K,K)
       DO 114 I=1,N
       DO 112 J=1,N
       IF (I.EQ.K) GO TO 114
       IF (K.NE.J) A(I,J)=A(I,J)-A(I,K)*A(K,J)
   112 CONTINUE
       Y(I) = Y(I)-A(I,K)*Y(K)
   114 CONTINUE
       DO 116 I=1,N
       IF (I.NE.K) A(I,K)=-A(I,K)*A(K,K)
   116 CONTINUE
   118 CONTINUE
       DO 122 K=1,N
       L=N+1-K
       KI=ICHG(L)
       IF (L.EQ.KI) GO TO 122
       DO 120 I=1,N
       TEMP = A(I,L)
       A(I,L) = A(I,KI)
   120 A(I,KI) = TEMP
   122 CONTINUE
   124 RETURN
       END
```

REFERENCES

1. Crimi, P.: A Method for Analyzing The Aeroelastic Stability of a Helicopter Rotor in Forward Flight. NASA CR-1332, August 1969.
2. Whittaker, E.T., and Watson, G.N.: Modern Analysis, Cambridge University Press, New York, 1962.
3. Gates, C. A. and DuWaldt, F.A.: Experimental and Theoretical Investigation of the Flutter Characteristics of a Model Helicopter Rotor Blade in Forward Flight. ASD TR 61-712, February 1962.
4. Sadler, S. G.: Rotating Blade Natural Frequency and Mode Shape Program for Coupled and Uncoupled Modes. Rochester Applied Science Assoc., Inc., RASA Report 69-3, April 1969.
5. DuWaldt, F.A.; Gates, C.A.; and Piziali, R.A.: Investigation of Helicopter Rotor Blade Flutter and Flapwise Bending Response in Hovering. WADC TR 59-403, August 1959.

NASA-Langley, 1970 —— 32 CR-1577