Penetration of Electromagnetic Pulses through Larger Apertures in Shielded Enclosures

Dikewood Industries, Inc.

May 1976
PENETRATION OF ELECTROMAGNETIC PULSES THROUGH LARGER APERTURES IN SHIELDED ENCLOSURES

University of Illinois
Urbana, IL 61801

for Dikewood Corporation
Albuquerque, NM 87106

May 1976

Final Report

Approved for public release; distribution unlimited.

This research was sponsored by the Defense Nuclear Agency under Subtask EB088, Work Unit 33, Work Unit Title: Coupling Characteristics of Apertures.

Prepared for
Director
DEFENSE NUCLEAR AGENCY
Washington, DC 20305

AIR FORCE WEAPONS LABORATORY
Air Force Systems Command
Kirtland Air Force Base, NM 87117

REPRODUCED BY
NATIONAL TECHNICAL INFORMATION SERVICE
U.S. DEPARTMENT OF COMMERCE
SPRINGFIELD, VA 22161

DDC
SEP 20 1976

5239
This final report was prepared by the University of Illinois, Urbana, IL, for Dikewood Corporation, Albuquerque, NM, under Contract F29601-74-C-0010, Job Order WDNE 2705, with the Air Force Weapons Laboratory, Kirtland AFB, NM. Mr. William D. Prather (ELP) was the Laboratory Project Officer-in-Charge.

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related Government procurement operation, the Government thereby incurs no responsibility nor any obligation whatsoever, and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

This report has been reviewed by the Information Office (OI) and is releasable to the National Technical Information Service (NTIS). At NTIS, it will be available to the general public, including foreign nations.

This technical report has been reviewed and is approved for publication.

WILLIAM D. PRATHER
Project Officer

LARRY W. WOOD
Lt Colonel, USAF
Chief, Phenomenology and Technology Branch

DO NOT RETURN THIS COPY. RETAIN OR DESTROY.
Penetration of Electromagnetic Pulses Through Larger Apertures in Shielded Enclosures

**Abstract**

The results of an initial investigation of the Singularity Expansion Method representation of the electromagnetic coupling through a rectangular aperture in a perfectly conducting sheet are reported. The problem is formulated in terms of the coupled Hallen-type integral equations for the dual problem of a rectangular plate. The integral equations are converted to a system of linear algebraic equations by way of the method of moments with subsectionally constant expansion functions and collocation testing. Several techniques used in

**Keywords**

Electromagnetic Fields and Waves
Interaction and Coupling
Aperture Penetration
minimizing the execution time of the computations are described. Some difficulties in accurately approximating the singularities of the system of integral equations by the singularities of the algebraic system are discussed. These difficulties arise because the subsectionally constant representation for the current cannot adequately represent the correct edge singularities in the currents on the plate. A set of pole trajectories indicative of the trends in pole location for the plate is reported. A listing of the pertinent computer code is provided.
## CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I</td>
<td>INTRODUCTION</td>
<td>5</td>
</tr>
<tr>
<td>II</td>
<td>THIN-PLATE INTEGRAL EQUATION FORMULATION FOR COMPLEX WAVE NUMBER</td>
<td>7</td>
</tr>
<tr>
<td>III</td>
<td>SYMMETRY CONDITIONS FOR THE NATURAL MODE CURRENTS</td>
<td>10</td>
</tr>
<tr>
<td>IV</td>
<td>THE NUMERICAL MODEL</td>
<td>16</td>
</tr>
<tr>
<td>V</td>
<td>ALGORITHMIC CONSIDERATIONS IN EVALUATING THE SYSTEM DETERMINANT</td>
<td>21</td>
</tr>
<tr>
<td>VI</td>
<td>NUMERICAL CHECKS ON THE ACCURACY OF THE POLES</td>
<td>25</td>
</tr>
<tr>
<td>VII</td>
<td>POLE TRAJECTORIES AS A FUNCTION OF SHAPE RATIO</td>
<td>31</td>
</tr>
<tr>
<td>VIII</td>
<td>CONCLUSIONS</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>APPENDIX A: THE SELF-PATCH INTEGRATION</td>
<td>35</td>
</tr>
<tr>
<td></td>
<td>APPENDIX B: THE SPARSE MATRIX ALGORITHMS</td>
<td>37</td>
</tr>
<tr>
<td></td>
<td>APPENDIX C: PROGRAM LISTINGS</td>
<td>43</td>
</tr>
<tr>
<td></td>
<td>REFERENCES</td>
<td>74</td>
</tr>
</tbody>
</table>
ILLUSTRATIONS

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Geometry of the Rectangular Plate</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Lowest Order Natural Mode Current Pairs for Each of the Symmetry Cases, a) J Symmetric w.r.t. x-Axis and Symmetric w.r.t. y-Axis, b) Symmetric-Antisymmetric, c) Antisymmetric-Symmetric, and d) Antisymmetric-Antisymmetric</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>Subsectioning for the Discretization of the Integral Equations</td>
<td>17</td>
</tr>
<tr>
<td>4</td>
<td>Organization of the System of Linear Equations</td>
<td>19</td>
</tr>
<tr>
<td>5</td>
<td>a) Conceptual Zoning for Calculation of the Interaction Matrix, b) Example of the Four Interaction Contributions to a Single Source Term</td>
<td>22</td>
</tr>
<tr>
<td>6</td>
<td>Calculated Pole Locations for Thin-Strip for Varying Numbers of Zones in the x-Direction and Different Edge Treatments (Cylinder Results from Ref. 6)</td>
<td>26</td>
</tr>
<tr>
<td>7</td>
<td>Computer Pole Trajectory Under Varying w/L with Zoning Changes</td>
<td>28</td>
</tr>
<tr>
<td>8</td>
<td>Behavior of Singular Component of Current at the Edge Under Change in Transverse Zoning</td>
<td>30</td>
</tr>
<tr>
<td>9</td>
<td>Pole Trajectories as Computed with 4x3 zoning</td>
<td>32</td>
</tr>
<tr>
<td>B1</td>
<td>Submatrix Organization for the Sparse Matrix Algorithms, a) the Original Matrix, and b) Triangularized Form with the Generated CMAT4</td>
<td>38</td>
</tr>
</tbody>
</table>
### TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Compatible Current Symmetry Features</td>
<td>14</td>
</tr>
<tr>
<td>2</td>
<td>Matrix Formulation Parameters</td>
<td>20</td>
</tr>
<tr>
<td>B1</td>
<td>Primary Indexing Quantities in the Algorithm</td>
<td>42</td>
</tr>
</tbody>
</table>
SECTION I

INTRODUCTION

This report presents the results of an investigation for representing the transient electromagnetic coupling through a rectangular aperture by means of the singularity expansion method.

The singularity expansion method, which was introduced by Baum in 1971 (ref. 1), has been subsequently applied to many scatterer geometries. The essence of the singularity expansion method is the representing of the temporal response of a body in terms of the complex natural frequencies for the body.

Taylor et al. point out that the singularity expansion for an aperture in an infinite perfectly conducting screen can be determined in terms of that for the complementary perfectly conducting plate by way of Babinet's principle (ref. 2). This approach was taken in the work reported here. The remaining discussion is directed to the equivalent problem of determining the current distributions on the complementary plate geometry.

Rahmat-Samii and Mittra have derived a coupled pair of Hallen-type integral equations governing the current behavior on the rectangular plate (ref. 3). The work reported here builds on their work by generalizing the integral equations and solution method to the complex frequency plane for the


SEM application. The same method-of-moments formulation, as described in (ref. 3), is used, i.e., two-dimensional pulse expansion functions with collocation testing.

In order that the computation time be practical in a problem of this complexity, a great deal of care was given to algorithmic streamlining in the conduct of this work. The streamlining includes maximum exploitation of geometric symmetry, organization of calculations to make use of redundant terms and partial terms occurring in the calculation, and direct algorithmic exploitation of matrix sparseness. The end result is a highly efficient computer code. Key features of the algorithms are discussed in this report.

The pulse expansion appears to be inadequate in accurately modeling the rectangular plate. The difficulty, which relates to representing the actual size of the plate, is demonstrated and discussed herein. Remedies for the problem are suggested, but they are outside the scope of the present work.

By holding the zoning scheme invariant while the aspect ratio of the plate was changed, self-consistent pole trajectories for the four fundamental modes were determined. For the reasons cited above, these poles cannot claim to be the exact poles for the body. They are, however, indicative of the trends in the pole behavior for the plate under change in aspect ratio. These results are reported and discussed in this context.
SECTION II
THIN-PLATE INTEGRAL EQUATION FORMULATION
FOR COMPLEX WAVE NUMBER

Rahmat-Samii and Mittra (ref. 3) give an integral equation formulation for the rectangular plate subject to time-harmonic excitation. Their results may be directly extended to the complex wavenumber case. That is, for the geometry in Figure 1 with \( \exp[st] \) time dependence, \( s = \sigma + j\omega \) complex, and an incident plane-wave magnetic field component

\[
\vec{H} = [H_{ox} \hat{u}_x + H_{oy} \hat{u}_y + H_{oz} \hat{u}_z] \exp[j(k_x x + k_y y + k_z z)]
\]

the following coupled integral equations result:

\[
\int_{-L/2}^{L/2} \int_{-w/2}^{w/2} \left\{ \begin{array}{l}
J_x(x,y) \\
J_y(x,y)
\end{array} \right\} K(x,y | x',y') \, dx' \, dy' = \frac{j}{k_z} \left( \begin{array}{l}
H_{ox} \\
H_{oy}
\end{array} \right) \exp[j(k_x x + k_y y)]
\]

\[
+ \frac{n}{k} \sum_{n=\infty}^{\infty} \left\{ \begin{array}{l}
1 \\
-1
\end{array} \right\} C_n J_{n+1}(-\sigma\rho/c) \, J_{n+1}(-\sigma\rho/c)
\]

The kernel is given by

\[
K(x,y | x',y') = \exp[-sR/c]/R
\]

with

\[
R = [(x - x')^2 + (y - y')^2]^{1/2}
\]

The \( J_x(x,y) \) and \( J_y(x,y) \) denote the respective \( x \) and \( y \) components of current on the plate; \( J_n(\xi) \) denotes the Bessel function of the first kind; the \( C_n \) are unknown constants; \( c \) is the velocity of light; and \( (\rho,\phi) \) are the polar coordinates for the point \( (x,y) \) on the plate. Equation (1) holds for \( x \in (-L/2,L/2) \) and \( y \in (-w/2,w/2) \), and \( z = 0 \).

It is pointed out that the two integral equations represented by (1) are
Figure 1. Geometry of the Rectangular Plate
coupled through the $C_n$ in the summation in the right-hand side. This summation is simply a Bessel function expansion of the homogeneous solution to the wave equation which occurs in the derivation of (1). Details of arriving at this expansion are found in (ref. 3). The pair of integral equations (1) is complete in the sense that no additional constraints are needed to correctly specify the currents. It is noteworthy, however, that current solutions to (1) satisfy the Meixner's edge condition (ref. 4). Namely,

$$
J_x[\pm(L/2 - d), y] + d^{1/2} \\
J_y[\pm(L/2 - d), y] + d^{-1/2} \\
J_x[x, \pm(w/2 - d)] + d^{-1/2} \\
J_y[x, \pm(w/2 - d)] + d^{1/2}
$$

$d \to 0$ (3)

The ability of a numerical solution to approximate the behavior of eqn. (3) is a key point in a subsequent discussion.

SECTION III
SYMMETRY CONDITIONS FOR THE NATURAL MODE CURRENTS

The natural frequencies of (1) occur when the complex frequency $s$ is such that there are non-trivial $J_x$ and $J_y$ and the accompanying $C_n$ satisfying (1) for $H = 0$. Such $J_x$ and $J_y$ solutions are natural mode current solutions for the rectangular plate, and the concomitant value of $s$ is a pole of the plate. The vanishing of incident wave dependence gives rise to symmetry in the integral equations. By discerning the symmetry relations \textit{a priori} and bringing them to bear upon solution procedures, one gains significant computational savings in the numerical solution for poles and natural modes. These symmetry relations are explored in this section.

The excitation-free form of (1) is

$$
\begin{align*}
\int_{-L/2}^{L/2} \int_{-w/2}^{w/2} J_x K(x,y|x',y') \, dx' \, dy' &= \frac{j \pi c}{s} \sum_{n=-\infty}^{\infty} C_n \left\{ J_{n+1} \exp[j(n+1)\phi] J_{n+1} (-sp/c) \\
&\quad + j^{n-1} \exp[j(n-1)\phi] J_{n-1} (-sp/c) \right\} \quad (4a)
\end{align*}
$$

and

$$
\begin{align*}
\int_{-L/2}^{L/2} \int_{-w/2}^{w/2} J_y K(x,y|x',y') \, dx' \, dy' &= \frac{j \pi c}{s} \sum_{n=-\infty}^{\infty} C_n \left\{ J_{n+1} \exp[j(n+1)\phi] J_{n+1} (-sp/c) \\
&\quad - j^{n-1} \exp[j(n-1)\phi] J_{n-1} (-sp/c) \right\} \quad (4b)
\end{align*}
$$

By using the symmetry of the Bessel function with respect to order, expanding the exponentials by way of Euler's identity, and appropriately adjusting the indices, one arrives at the following equation after some manipulation.
\[
\begin{align*}
\frac{L}{2} & \quad \frac{w}{2} \\
\int_{-L/2}^{L/2} \int_{-w/2}^{w/2} & {J}_x K \, dx \, dy' \\
& \quad \int_{-L/2}^{L/2} \int_{-w/2}^{w/2} {J}_y K \, dx' \, dy \\
& = \frac{4\pi c}{s} \sum_{n=0}^{\infty} \left\{ j^{n+1} d^+_{n} \left[ \cos(n+1) \phi \, {J}_{n-1}(-\omega \varphi/c) - u_{n-1} \cos(n-1) \phi \, {J}_{n-1}(-\omega \varphi/c) \right] \\
& \quad - j^{n} d^-_{n} \left[ \sin(n+1) \phi \, {J}_{n+1}(-\omega \varphi/c) - \sin(n-1) \phi \, {J}_{n-1}(-\omega \varphi/c) \right] \right\} \quad (5a) \\
\text{and} \\
& \quad \int_{-L/2}^{L/2} \int_{-w/2}^{w/2} \frac{4\pi c}{s} \sum_{n=0}^{\infty} \left\{ j^{n+1} d^+_{n} \left[ \sin(n+1) \phi \, {J}_{n+1}(-\omega \varphi/c) + \sin(n-1) \phi \, {J}_{n-1}(-\omega \varphi/c) \right] \\
& \quad + j^{n} d^-_{n} \left[ \cos(n+1) \phi \, {J}_{n+1}(-\omega \varphi/c) + u_{n-1} \cos(n-1) \phi \, {J}_{n-1}(-\omega \varphi/c) \right] \right\} \quad (5b) \\
\text{where} \\
& d^\pm_n = c_n \pm c_{-n} \\
& \text{and} \\
& u_n = \begin{cases} 1, & n \geq 0 \\
0, & n < 0 \end{cases}
\end{align*}
\]

It is noted that the \( d^+_n \) multiply terms containing cosine functions in the \( J_x \) equation, while they multiply terms containing sine functions in the \( J_y \) equation. The situation is reversed for the \( d^-_n \).

Because of the symmetry properties of the kernel, the integral operator on the left-hand sides of (5) produces a function whose symmetry character is identical to that of the current on which it operates. Then, for a given current symmetry, only part of the \( d^\pm_n \) on the right-hand side may be non-zero because of the symmetries possessed by the trigonometric terms. Thus, the respective symmetries for \( J_x \) and \( J_y \), which are compatible, and the
surviving terms in the right-side series may be discerned by 1) postulating a
symmetry for \( J_x \), 2) determining from (5a) which right-hand side terms survive so
as to be compatible with the \( J_x \) symmetry, 3) observing in (5b) the variation
which terms have non-zero coefficients, and 4) determining the \( J_y \) symmetry
conditions compatible with the postulated \( J_x \) symmetry conditions.

For example, if \( J_x \) is symmetric with respect to the \( y \) axis and antii-
symmetric with respect to the \( x \) axis, only \( \sin(n + 1) \phi \) terms with \( n \) even are
compatible in (5a). Thus, only \( d_n^- \), \( n \) even, may be non-zero. In the right-
hand side of (5b), the coefficients multiply \( \cos(n + 1) \phi \) terms with \( n \) even.
These cosines sum to functions which are antisymmetric with respect to the
\( y \) axis and symmetric with respect to the \( x \) axis. Stated mathematically, if
\[
J_x(x,y) = J_x(-x,y) \quad (6a)
\]
and
\[
J_x(x,y) = -J_x(x,-y) \quad (6b)
\]
then
\[
d_n^+ = 0, \quad \text{for all } n, \quad (6c)
\]
\[
d_n^- = 0, \quad \text{if } n \text{ odd}, \quad (6d)
\]
and
\[
J_y(x,y) = -J_y(-x,y) \quad (6e)
\]
\[
J_y(x,y) = J_y(x,-y) \quad (6f)
\]
These vector symmetries are in accord with the general symmetry relations
given by Baum (ref. 5). The information in (6) may be used to reduce the
complexity of the integral equations (4), viz., by (6a,b,e,f) the range of
each integration may be halved while by (6c,d) the zero terms of the right-
hand side are known \textit{a priori}:
\[
\begin{align*}
L/2 & \quad w/2 \\
\int_0^{L/2} \int_0^{w/2} J_x K^{+}(x,y|x',y') \, dx' \, dy' \\
& = \frac{\pi c}{s} \sum_{n=0}^{\infty} d_n^- j^{n-1} [\sin(n + 1) \phi \, J_{n-1}(\text{-}\omega \rho/c) - \sin(n - 1) \phi \, J_{n-1}(\text{-}\omega \rho/c)] \quad (7a)
\end{align*}
\]

5. Baum, C. E., "Interaction of Electromagnetic Fields with any Object
which has an Electromagnetic Symmetry Plane," Interaction Note 63,
Air Force Weapons Laboratory, Kirtland AFB, NM, March 1971.
and

\[
\frac{L/2}{v/2} \int \int_{y} K(x,y|x',y') \, dx \, dy' \quad \int_{y} \int_{x} K(x,y|x',y') \, dx' \, dy' = \frac{\pi c}{s} \sum_{n=0}^{\infty} j^{n+1} d_x [\cos(n+1)\phi \, j_{n+1}(-\omega/c) + u_{n-1} \cos(n-1)\phi \, j_{n-1}(-\omega/c)] \tag{7b}
\]

where

\[
K(x,y|x',y') = K(x,y|x',y') - K(x,y|-x',y') + K(x,y|x',-y') - K(x,y|-x',-y') \tag{8a}
\]

and

\[
K(x,y|x',y') = K(x,y|x',y') + K(x,y|-x',y') - K(x,y|x',-y') - K(x,y|-x',-y') \tag{8b}
\]

For subsequent reference

\[
K^{++}(x,y|x',y') = K(x,y|x',y') + K(x,y|-x',y') + K(x,y|x',-y') + K(x,y|-x',-y') \tag{8c}
\]

and

\[
K^{--}(x,y|x',y') = K(x,y|x',y') - K(x,y|-x',y') - K(x,y|x',-y') + K(x,y|-x',-y') \tag{8d}
\]

are defined as well. Equations (7) are enforced for \( z = 0, x \in (0,L/2) \)
and \( y \in (0,w/2) \).

Table 1 summarizes the four symmetry cases which are derived as in the
foregoing discussion. By means of this table, four integral equation pairs
can be constructed in the spirit of (7) by replacing the kernels in (7) with
the appropriate kernels from the table and retaining only the non-vanishing
terms in the series expansion.

Figure 2 depicts qualitatively the respective modal current distributions
for the lowest frequency natural resonance exhibiting each symmetry.
Table 1

COMPATIBLE CURRENT SYMMETRY FEATURES

<table>
<thead>
<tr>
<th>J_x</th>
<th>J_y</th>
</tr>
</thead>
</table>
| Sym. w.r.t.
  x axis | Sym. w.r.t.
  y axis | Kernel | Coefs. ≠ 0 | Compatible Trig. Fns. | Kernel | Coefs. ≠ 0 | Compatible Trig. Fns. | Sym. w.r.t.
  x axis | Sym. w.r.t.
  y axis |
| sym       | sym       | k^{++}  | cos 2n\phi | d_{2n+1}^{+} | sin 2n\phi | k^{-} | anti       | anti       |
| sym       | anti      | k^{+-}  | cos (2n + 1)\phi | d_{2n}^{+} | sin (2n + 1)\phi | k^{+} | anti       | sym       |
| anti      | sym       | k^{-+}  | sin (2n + 1)\phi | d_{2n}^{-} | cos (2n + 1)\phi | k^{++} | sym       | anti       |
| anti      | anti      | k^{-}   | sin 2n\phi | d_{2n+1}^{-} | cos 2n\phi | k^{--} | sym       | sym       |
Figure 2. Lowest Order Natural Mode Current Pairs for Each of the Symmetry Cases, a) \( J_x \) Symmetric w.r.t. x-Axis and Symmetric w.r.t. y-Axis, b) Symmetric-Antisymmetric, c) Antisymmetric-Symmetric, and d) Antisymmetric-Antisymmetric
SECTION IV
THE NUMERICAL MODEL

The integral equation pair of the form (7) for each of the four symmetry cases can be discretized by the method of moments. In the work reported here, two-dimensional, subsectionally constant expansion functions were used with collocation testing. The zoning scheme is represented in Figure 3.

The unknown currents \( J_x \) and \( J_y \) were expanded in piecewise constant functions as in (ref. 3) with subsectioning of the form given in Figure 3. Notice that half-width patches are used at the edges of the plate so that match points lie precisely on the edge of the plate. The half-width pulse has proved useful in realizing the actual electrical size of a body in one-dimensional problems (ref. 6). Some numerical experimentation was also done with full-sized pulses on the edges and comparative results are reported in a later section. Some difficulties occur in definition of the edge of the plate in the present formulation because of the presence of two current components which have the asymptotic behavior given in (3). This difficulty is discussed in a later section.

The boundary condition \( J_{\text{norm}}^x = 0 \) must be enforced on selected patches at the edge of the place as discussed in (ref. 3). Concomitantly, only as many \( d_n^x \)'s are retained in the right-hand side summation in (7) as there are current values preassigned to zero. The shaded patches in Figure 3 indicate the selection of patches where a current component is preassigned a zero value. At the corner patch, both components are preassigned zero values.

Figure 3. Subsectioning for the Discretization of the Integral Equations
By assigning one match point per expansion patch and by retaining one series expansion term for each current value preassigned in each of the two integral equations, a consistent (i.e. square) system of linear equations results. The truncated summation is taken to the left-hand side so that a homogeneous system results. The matrix organization used to represent these equations is given in Figure 4. Table 2 defines the computer variables noted in Figure 4, primarily for reference purposes in the next section.

The matrix that results is a function of the complex frequency $s$. A natural resonance occurs when $s$ has a value such that the matrix has a zero determinant; hence, the homogeneous system of equations has a non-trivial solution. The next section explores some algorithmic considerations in the efficient generation and manipulation of the matrix.
Figure 4. Organization of the System of Linear Equations
Table 2

MATRIX FORMULATION PARAMETERS

$N_{I1}$  
Number of match points on the zoned quadrant of the plate.

$N_{I2} = N_{I1}$

$N_{PREJ}$  
Number of patches along the $|x| = L/2$ edge where $J_x$ is preassigned to zero.

$N_{PREI}$  
Number of patches along the $|y| = w/2$ edge where $J_y$ is preassigned to zero.

$N_{J1} = N_{I1} - N_{PREJ}$  
Number of unknown current values in $J_x$ expansion.

$N_{J2} = N_{I2} - N_{PREI}$  
Number of unknown current values in $J_y$ expansion.

$N_{J3} = N_{PREI} - N_{PREJ}$  
Number of preassigned current values (Also the number of coefficients retained in summation).
SECTION V

ALGORITHMIC CONSIDERATIONS IN EVALUATING THE SYSTEM DETERMINANT

Some considerations taken into account in generating the system matrix and evaluating its determinant efficiently are discussed in this section. Since these two operations must be repeatedly carried out for many values of s in the course of determining the natural frequencies of the plate, it is essential that clean, efficient computer programming and coding be used so that execution of the program will be affordable. The volume of code in the algorithms is consistently compromised toward a larger size in order to meet the following two time-efficient objectives:

1. Avoidance of calculating the same quantity twice; and

2. Avoidance of logical decisions, particularly those which might be imbedded in loops.

The program is discussed in the context of the following major segments:

1. Computation of an "interaction matrix";
2. Construction of the non-zero submatrices of the system matrix from the interaction matrix;
3. Computation of the series terms' submatrix; and
4. Determinant evaluation.

The major contribution to the elimination of redundant calculations is the one-time computation of an "interaction matrix" which is made up of the individual kernel integral terms from (2) for all argument combinations which occur in the computation. The subsequent program step then picks, by subscript, entries from this matrix and constructs the appropriate kernel from one of equations (8) according to the symmetry conditions being solved. This procedure can be viewed in terms of the layout given in Figure 5a. The terms in the interaction matrix are those evaluated for the match-point as
Figure 5.  a) Conceptual Zoning for Calculation of the Interaction Matrix.  b) Example of the Four Interaction Contributions to a Single Source Term
shown in the lower left with the source patches indexed over the entire plate to generate the matrix. Thus, all geometric relationships which occur in the kernel terms are encompassed in the calculation. Note that all source patches are full patches for this calculation. The effect of half patches at the edges is accounted for by weighting by a factor of 1/2 the edge contributions. The kernel integral appropriate to the symmetry is constructed by summing with correct signs the appropriate elements from the matrix. Figure 5b gives an example of the four source patches entering into one kernel integral.

Differing degrees of sophistication are required in the calculation of the interaction terms depending on the spacing of the patches for which an interaction is being calculated. For the self patch, i.e., the patch in which the match-point resides, the integration of the kernel must be performed analytically because of the integrable singularity in the kernel there. Appendix A gives a series approximation to this integral. The result in Appendix A is evaluated directly in the program. For the patches adjacent to the patch containing the match point, the kernel is a rapidly varying but well-behaved function. The integration over these patches is evaluated numerically by a polynomial approximation. For patches further separated, the kernel is slowly varying and the integral is evaluated approximately as the product of the value of the kernel at the center of the patch and the area of the patch.

Some minor time economy is achieved in the matrix filling algorithm, which is a four-dimensional loop. The economy is found in the form of decision-free indexing, that is, the source contributions from interior patches, from \(|x| = L/2\) edge patches, from \(|y| = w/2\) edge patches, and from corners take on different forms. Rather than index over all source patches
with logical decisions implemented to discriminate among the four cases above, four different loops are used.

The computation of the series term submatrix is relatively straightforward. Because the Bessel-trigonometric products appear in two terms each, they are all precalculated and stored in a vector. The individual terms are then constructed from them.

The determinant evaluation profits significantly from an exploitation of the sparceness of the matrix. Either of two approaches may be taken to the sparse matrix manipulations. One is to separate the matrix algebraically and calculate an inverse as a composite of inverses of terms involving the submatrices. The alternative approach is to attack the matrix directly with Gaussian elimination, and to exploit the sparceness directly in the algorithm. The latter approach was chosen for the present purpose because it is judged to be slightly faster computationally and because in order to determine natural mode solutions for the SEM formulation, the homogeneous system of equations occurring at a pole must be backsolved. The algorithm resulting from the second approach is described in Appendix B.

The determinant evaluation routine itself appears in Appendix C as the function routine CPLATE.
SECTION VI
NUMERICAL CHECKS ON THE ACCURACY OF THE POLES

The results of some numerical checks on the accuracy of the pole location as determined from the numerical model described in Sections II through V are reported. It is shown that the model predicts well the poles for narrow strips possessing essentially thin scatterer resonance properties. Difficulties occur, however, in obtaining self-consistent results under different zone sizes for plates with larger aspect ratios. It is conjectured that the difficulty occurs because the subsectionally constant current representation is inadequate to represent the correct edge behavior for the currents—particularly the singular behavior for the parallel component. The rationale behind this conjecture is discussed.

Initial tests on the accuracy of the model were made for a rectangular strip with a shape ratio $w/L = 1/10$. Such a strip has an approximate equivalent dipole whose diameter-to-length ratio is $1/10\pi$.

Figure 6 gives the results of pole determinations for the first two poles for various numbers of pulses in the expansion of the current and for two different treatments of the edge pulse. The strip was zoned with one pulse across a quadrant. The numbers indicated in the figure are the numbers of pulses along the longitudinal direction of a quadrant. The "half-pulse" results are those obtained by the zone scheme described in Section IV where half-width pulses are placed at the edge so that match points fall at the edge. The "full-pulse" results are those obtained by zoning the plate with equal-sized pulses over the entire quadrant. In the latter case an \textit{a posteriori} adjustment is made in the data correcting the length of the strip such that the end of the corrected strip lies at the end match-point.
Figure 6. Calculated Pole Locations for Thin-Strip for Varying Numbers of Zones in the x-Direction and Different Edge Treatments
(Cylinder Results from Ref. 6)
It is seen that the differences are small both for varying order and increasing pulse density. The NX = 6 results for the second pole show some departure from the trend established by the results for NX = 4 and NX = 5. This is attributable to the fact that the matrix is on the brink of numerical instability for NX = 6. The results for NX = 7, which are not shown, are observed to be meaningless because of the instability manifested.

For comparison purposes, the first two poles for an equivalent cylinder (one whose circumference equals the strip width) are given as found in ref. 7. These results are judged reliable inasmuch as they have been cross-checked among several integral equation formulations and for several method-of-moments schemes. The equivalent radius taken is, of course, an approximation. It is seen that the half-pulse solutions compare slightly more favorably with the cylinder results. Because of this, and moreover, because the a posteriori data adjustment is avoided with the half-pulse scheme, it was used consistently in the remaining solutions.

The stable convergence properties of the numerical model exhibited for the thin-strip solution are not manifested for higher aspect ratios. The reason for the difference is that the strip is essentially a one-dimensional problem and the transverse (y-directed) component of current has little effect on the dominant longitudinal current. For wider structures the coupling is significant and inadequacies in the modeling of the singularities of the currents produce inaccuracies which are highly sensitive to zoning.

Figure 7 shows the results obtained for a pole trajectory as a function of the shape factor w/L where the zoning in the y-direction was adjusted.

Figure 7. Computer Pole Trajectory Under Varying w/L with Zoning Changes
according to the value of \( w \). It is evident that the solutions are unstable with respect to the zoning on the plate. Attempts to increase the number of zones significantly to improve upon the situation resulted in numerical instabilities in the matrix which cause the pole search iteration to fail.

The reason for the difficulty manifested in Figure 6 is believed to lie in the way that the edge of the plate is defined with the piecewise constant current expansion. Consider the characteristics of the two current components along a line traversing the plate in the \( y \)-direction as shown in Figure 8. The correct edge behavior at \( |y| = w/2 \) is that given in equations (3). The zoning scheme, however, forces \( J_x(x, \pm w/2) \) to take a finite value. The current extrapolates to a singular point for some \( y > w/2 \), i.e., the numerical model represents a plate whose width is greater than \( w \).

If the number of transverse zones is increased as indicated by the dashed curve in Figure 8, the steepness of the edge behavior of \( J_x \) is increased, and the extrapolation is characteristic of a narrower plate as compared to the first case. This narrowing of the effective width in the model is reflected in an increased \( Q \) (resonance quality factor) as the jumps in Figure 7 indicate.

One is tempted to conclude that a full-width pulse at the edge is a cure for the problem since the point at which the pulse current is defined is shifted relative to the edge as zoning is changed with full-width pulses. The effect of this procedure is to transfer the problem from component of current whose behavior is singular at the edge to the component which goes to zero. With full pulses at the edges, the normal component of current would go to zero interior to the plate rather than at the edge as it properly should.

An approach which is potentially a remedy for this difficulty is discussed in the conclusions.
Figure 8. Behavior of Singular Component of Current at the Edge Under Change in Transverse Zoning
SECTION VII
POLE TRAJECTORIES AS A FUNCTION OF SHAPE RATIO

Figure 9 gives the results obtained for pole location for the lowest order pole of each of the symmetries as a function of w/L. Clearly, as the previous section indicates, the results cannot be taken as the correct results for the plate. However, the zoning was fixed for all w/L and the results are expected to reflect the proper trends for these trajectories.

The ++ and +- modes are in essence dipole modes, and their poles show the general lowering of Q as w/L increases. (The ++ indicates that the J_x component is symmetric both with respect to the x and y axes - see Table I.) The -- mode can be thought of as a dipole mode in the transverse direction. As a result it shows a strong frequency dependency on the transverse dimension w. When w/L = 1, the -- mode is identical to the ++ mode rotated spatially 90 degrees. Consequently, the two trajectories coalesce as w/L \rightarrow 1, when the zoning is 5x5 so as to preserve symmetry in the numerical mode. The failure of the 5x3 zone case is due to the reasons outlined in the previous section.
Figure 9. Pole Trajectories as Computed with Zoning Fixed.
SECTION VIII
CONCLUSIONS

The application of SEM to the equivalent problems of the perfectly conducting rectangular plate and the rectangular aperture in a perfectly conducting screen has been conducted with partial success. The applicability of SEM and the computational feasibility of determining SEM quantities are demonstrated. It is further demonstrated that by careful program construction, the computational costs of numerical treatment of two-dimensional problems can be made quite reasonable. The cost of generating a matrix and calculating its determinant by the methods discussed herein is less than 10 cents for each value of $s$.

Difficulties arise in the subsectionally constant current zoning because of a failure to properly model the edge conditions. Whereas Rahmat-Samii and Mittra (ref. 3) obtained good radar cross-section results with such a zoning scheme, the pole locations are highly sensitive to the zoning. Radar cross-section is a far-field quantity, and the integrated effects of the errors are small there. The pole locations, on the other hand, depend strongly on the dimensions of the structure, and it is evident that the plate size must be brought to bear in a precise fashion for the pole locations to be correct.

The edge problem can be remedied by imposing the edge conditions (3) directly in the basis set elements for edge zones. Davis has done this successfully for the circumferential component of current on a thick cylindrical scatterer (ref. 8). The circumferential current

component is singular at the ends of the cylinder. The introduction of the
singular basis element will produce a significant complication to the
problem in that a second singularity, that of the current, must be inte-
grated analytically in order to implement the model with edge conditions
imposed. An independent check on program accuracy is dictated for a
problem of this scope before proceeding with the edge condition approach.
APPENDIX A

THE SELF-PATCH INTEGRATION

The term for the interaction matrix for IDIF = JDIF = 0, i.e., where the match point lies at the center of the source patch, can be written

\[ I_s = 4 \int_{\Delta x/2}^{\Delta x/2} \int_{\Delta y/2}^{\Delta y/2} K(0,0|x',y') \, dx' \, dy' \]  
(A1)

This presumes a unit amplitude expansion pulse over the patch whose dimensions are \( \Delta x \) and \( \Delta y \). The symmetry of the kernel with respect to both \( x' \) and \( y' \) is employed in the forming of (A1). This integral can be transformed to polar coordinates as

\[
I_s = 4 \left\{ \int_{\phi=0}^{\pi/2} \int_{\rho=0}^{\Delta x/2 \cos \phi} \exp[-sp/c] \, dp \, d\phi \right. \\
+ \left. \int_{\phi=\tan^{-1} \frac{\Delta y}{\Delta x}}^{\pi/2} \int_{\rho=0}^{\Delta x/2 \cos \phi} \exp[-sp/c] \, dp \, d\phi \right\}
\]

\[
= -\frac{4c}{s} \left\{ \int_{\phi=0}^{\pi/2} \left[ \exp(-s\Delta x \sec \phi/2c) - 1 \right] \, d\phi \\
+ \int_{\phi=\tan^{-1} \frac{\Delta y}{\Delta x}}^{\pi/2} \left[ \exp(-s\Delta y \csc \phi/2c) - 1 \right] \, d\phi \right\}
\]  
(A2)

If the exponential functions in the integrand are then expanded in McLauren series, the resulting terms of powers of secants and cosecants possess tabulated integrals. The result for three terms retained in the series is
\[ I_s = -\frac{4c}{s} \left\{ -\frac{s\Delta x}{2c} \cdot \frac{1}{2} \ln q_y + \frac{1}{2} \left(\frac{s\Delta x}{2c}\right)^2 \frac{\Delta y}{\Delta x} \right. \\
- \frac{1}{6} \left(\frac{s\Delta x}{2c}\right)^3 \frac{\Delta x (\Delta x^2 + \Delta y^2)^{1/2}}{2\Delta y^2} - \frac{s\Delta y}{2c} \frac{1}{2} \ln q_x \\
+ 1/2 \left(\frac{s\Delta y}{2c}\right)^2 \frac{\Delta x}{\Delta y} - 1/6 \left(\frac{s\Delta y}{2c}\right)^3 \frac{\Delta y (\Delta x^2 + \Delta y^2)^{1/2}}{2\Delta y^2} \right\} \]

where

\[ q_{\left| x \right|} = \frac{\left(\Delta x^2 + \Delta y^2\right)^{1/2}}{\left(\Delta x^2 + \Delta y^2\right)^{1/2}} \left(\frac{\Delta x}{\Delta y}\right) \]

\[ q_{\left| y \right|} = \frac{\left(\Delta x^2 + \Delta y^2\right)^{1/2}}{\left(\Delta x^2 + \Delta y^2\right)^{1/2}} \left(\frac{\Delta y}{\Delta x}\right) \]
1. Introduction

This Appendix describes the algorithmic approach to minimize the computation time involved in Gaussian elimination triangularization of systems of matrix equations which are "sparsely coupled." The term "sparsely coupled" as applied in this Appendix implies the matrix equation form given in (B1).

\[
[M] [X] = \begin{bmatrix}
M_1 & 0 \\
0 & M_2
\end{bmatrix}
\begin{bmatrix}
X
\end{bmatrix}
= 
\begin{bmatrix}
B
\end{bmatrix}
\tag{B1}
\]

It is clear that in this form the sole coupling between the "upper" and "lower" systems of equations is contained in the matrix $M_2$. Generally, the number of columns in $M_2$ is small compared with the order of the overall system.

An algebraic approach to exploiting the sparceness of (B1) results in solutions which are given in terms of the several submatrices and their inverses. (See, for example, ref. 9.) It is well-known, however, that it is sufficient for the purposes of determinant calculation and equation solution to triangularize the composite matrix in (B1). The triangularization process involves fewer operations than the diagonalization necessary for the calculation of an inverse.

\[ \begin{bmatrix}
\text{CMAT1} & \begin{bmatrix} 0 \\ \text{(NI1 x NJ2)} \end{bmatrix} \\
\text{(NI1 x NJ1)} & \begin{bmatrix} \text{CMAT2} \\ \text{(NI2 x NJ2)} \end{bmatrix} \\
\end{bmatrix} \]

\[ \text{NI3} = \text{NI1} + \text{NI2} \]
\[ \text{NJ3} = \text{NI3} - \text{NJ1} - \text{NJ2} \]

(a)

\[ \begin{bmatrix}
\text{#0} & \begin{bmatrix} 0 \\ \text{CMAT4} \end{bmatrix} & \begin{bmatrix} 0 \\ \text{NI4} \end{bmatrix} \\
\text{0} & \text{0} & \begin{bmatrix} \text{CMAT4} \\ \text{(NI4 x NI2)} \end{bmatrix} \\
\end{bmatrix} \]

\[ \text{NI4} = \text{MAX} (\text{NI1} - \text{NJ1}, 0) \]

(b)

Figure B1. Submatrix Organization for the Sparse Matrix Algorithms, a) the Original Matrix, and b) Triangularized Form with the Generated CMAT4.
This Appendix describes an algorithmic exploitation of the sparseness of the composite matrix in (Bl). That is, a numerical process is given whereby only the non-zero subelements are stored and operated on, with the computational operations being those which render the composite matrix M upper triangular. The determinant of the composite matrix results directly from this triangularization. A solution for X in (Bl) requires a backsolving process involving the triangularized form of M and a vector resulting from applying the elimination operations to the vector B. Routines to perform these operations are given as well.

Appendix C gives listings of the routines built on this algorithm. The triangularization routine is named SPRHOM. The backsolving procedure is performed by the entry HOMSLV to the routine SPRSLV. (The name entry SPRSLV backsolves an inhomogeneous system and is not used for present purposes.)

2. The Algorithm

The routine SPRHOM is simply an implementation of the Gaussian elimination procedure with maximum pivot selection applied to the composite matrix M in (Bl) viewed as a whole. The sparseness of M is exploited by storing only the non-zero submatrices in (Bl) and skipping the operations involving zero elements. The result is a substantial saving in both storage and computation time.

The forms of the input and of the end product for the triangularization are given in Figure (Bl) with the sizes of the respective submatrices defined. It is recalled that the fundamental process in the Gaussian elimination procedure is the elimination of all or part of the elements of a column of a matrix with respect to a pivot element, commonly the element of greatest magnitude in the column. That is, for a column under process, the row
containing the main diagonal element of the matrix which falls in that
column. All or part of the elements not on the main diagonal are "eliminated"
or made zero by subtraction of some multiple of the row containing the col-
umn maximum. In the triangularization procedure, the part of the column com-
prising elements lying below the main diagonal after row exchange are elim-
inated. If the matrix is a part of a system of equations with non-zero right-
hand side, the row operations of exchange and subtraction of a constant
multiple of another row must be performed on the corresponding elements of
the right-hand side vector as well.

The algorithm of the routine SPRHOM operates according to the Gaussian
elimination procedure described above. However, the three submatrices CMAT1,
CMAT2, and CMAT3 are stored individually. In addition, the routine generates
a submatrix CMAT4 in the course of selecting pivots for the columns contained
in CMAT2. Further, the "elimination" of elements of submatrices that are
zero a priori is skipped. The result is significant storage and time economy.

In order to minimize logic decisions in the routine, it is organized to
operate sequentially through the partitioned matrix. The steps are as follows
(it is convenient to follow the thinking of these steps by tracing the loca-
tion diagonal of the composite with the aid of Table B1):

a. Perform conventional Gaussian elimination to zero the elements
CMAT1(I,J) for I > J, i.e., the elements below the main diag-

longal of M. Choose maximum pivot elements in conventional
fashion. Carry row operations into CMAT3.

b. Create CMAT4 by row swapping with CMAT2 so as to choose
maximum pivot elements. Perform elimination to zero CMAT4
elements for I > J and the entire column of CMAT2. The
number of rows created in CMAT2 is NI4 = NI1 - NJ1, the
amount by which CMAT1 is over-square. Carry row opera-
tions across into CMAT3.
c. Choose maximum pivot rows in columns of \text{CMAT2} with 
\[ J > N14 \] 
and swap with rows given by \[ I = J - N14 \] 
The rows containing the \( J \)th column diagonal element of 
the composite. Conduct elimination to zero elements 
with \[ I > J + N14 \]. Carry row operations into \text{CMAT3}.
d. Conduct conventional pivot selection and elimination 
on \text{CMAT3} to zero elements \[ \text{CMAT3}(I, J) \] with 
\[ I > J + NJ1 + NJ2 \].

In each column elimination operation, the pivot value is multiplied into a 
product accumulator to produce a value for the determinant of the composite 
matrix. The sign of this product is changed at each row swap in accord with 
the rates of matrix algebra row operations.

The backsolving routine \text{SPRSLV} with its entry \text{HOMSLV} operate in a 
straightforward manner on the submatrices as reduced by \text{SPRHOM}.
Details are omitted here, but the routines may be easily followed in a row-
by-row flow from the bottom of the composite matrix, if one keeps in mind 
the row index relations of column 4 of Table B1. The entry \text{HOMSLV} assumes a 
zero determinant value resulted (approximately) from \text{SPRHOM} and the last 
element of the solution vector is picked as unity. (The zero determinant 
results from a zero falling at the last diagonal location in maximum pivoting 
Gaussian elimination.) The remainder of the homogeneous solution vector is 
backsolved conventionally relative to this last element. The vector is then 
renormalized so that its maximum element is unity.
Table B1  
PRIMARY INDEXING QUANTITIES IN THE ALGORITHM

<table>
<thead>
<tr>
<th>Submatrix</th>
<th>Size of Submatrix</th>
<th>Indices of Main Diag. of Compos.</th>
<th>Relative Row Index of CMAT3 and CRHS</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMAT1</td>
<td>N11 x N12</td>
<td>(J,J)</td>
<td>13 = I</td>
</tr>
<tr>
<td>CMAT4</td>
<td>N11 - N12 x N12</td>
<td>(J,J)</td>
<td>13 = I + N12</td>
</tr>
<tr>
<td></td>
<td>(can be null)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CMAT2</td>
<td>N12 x N12</td>
<td>(J - (N11 - N12), J)</td>
<td>13 = I + N12</td>
</tr>
<tr>
<td>CMAT3</td>
<td>N11 + N12 x N11 + N12 - N11 - N12</td>
<td>(J + N11 + N12, J)</td>
<td>13 = I3</td>
</tr>
</tbody>
</table>

1. Quantities given in terms of input parms. to the routine. Related internal quantities are given in Figure Bl.

2. Relative to I, the row index of the submatrix in question.
APPENDIX C

PROGRAM LISTINGS

All code compilable on IBM OS/360 and OS/370 FORTRAN levels G or H.

The routine ZANLYT and its service routine UERTST is taken from the program library FORTUOI made available by the Computer Services Office, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801. The routines BSLJZ and BSCJZ are taken from the International Mathematical and Statistical Library (IMSL). They may not be reproduced apart from this application program package. The IMSL library is available by subscription from IMSL, Inc., 6100 Hillcroft, Suite 510, Houston, Texas 77036.
Pole Search Program for Thin-Plate Formulation of Thin-Plate Scatterer

By L.W. Pearson 8/74

Implicit Real*8(A,R,P-H,N-Z), Complex*16(C)

Common /GEOM/, XSYM, YSYM, W, NX, NY, IPREAS(10), IPREJ, NPREF, NPREJ

Integer MES(4, 2) /SYM1, ET1, ET2, ET3, ET4, ET5, ET6, ET7, ET8, ET9, ET10,

Data (.13, 008, 0, 0D), /PLUS/-1*1, PI/1, 141592653589793/.

Data X, X*, Y, Y*/

External CPlate

Dimension CX(20), INFFP(20)

Logical Lauto

100 Read(5, 1, END=999) XSYM, YSYM, NX, NY, WS, WM, CSUNCP, Lauto

Format(1A1, 2X, 2I3, 5F10.4, 4T80, L1)

IW=1

IMY=1

TF(XSYM, NP, PLUS) IW=2

TF(YSYM, NP, PLUS) IMY=2

WS(WM-WS)/WS

TF(NW, GT=0) Go to 109

NW=W

WS=WS

105 IF(WN=WLT, WM-WD) NW=NW+1

Do 200 IM=1, NW

W=WN+1*W-1)/WS

IF(.NOT. Lauto) Go to 140

Skip past auto zoning

Routine to determine no of expansion pulses based on electrical size of plate

Test(WV = 1.885D10/DAst(DIMAG=CSUNCP))

//

NPPWVL=6

///// //

WFLNX=1/TESTW

NX=IDINT(WFLNX*NPPWVL)

TFIDFLOAT(NX, LT, WFLNX*NPPWVL) NX=NX+1

WFLNY=W/TESTW

NY=IDINT(WFLNY<NPPWVL)

TFIDFLOAT(NY, LT, WFLNY*NPPWVL) AY=NY+1

NX=MIN(NX, 5)

NY=MIN(NY, 5)

///// //

Begin setup for alternate edge patch preassignment

140 NPREF=(NX*2+1)/3

NPREF=I(NY*2+1)/3

TFNX-2*NPREF+2, LE-1, AMD, NPREF, GT-1) NPREF=NPREF-1

IF(NY>2*NPREF+2, LE-1, AMD, NPREF, GT-1) NPREF=NPREF-1

Do 110 I=1, NPREF

IPREAS(NPREF+1-I)=NX-3*I+3

110 Continue

Do 120 J=1, NPREF

IPREAS(NPREF+J-1)=NY-3*J+3

120 Continue

Locations where x-directed current is set to zero given by SUBSCR1

00500

00510

00520

00530

00540

00550

00560

00570

00580

00590

00600

00610
C
C WRITE(6,2) W,CSUNDP
2 FORMAT(1'ENTER ITERATION',/,'OSHAPE RATIO =',F5.3,5X,
1'STARTING FREQ =',2D12.4)
WRITE(6,3)
3 FORMAT(1'O',10X,'CIP SYMMETRY',6X,'PULSES',3X,'PRESS:AM LOC:INS')
WRITE(6,4) HX,(MES(I,14X),I=1,4),NX,(IPRESS(J),J=1,NPREJ)
4 FORMAT(1'O',41,'-DIR',5X,4X,16,5X,10'3)
WRITE(6,4) HY,(MES(I,14Y),I=1,4),NY,(JIPRESS(J),J=1,NPREF)
WRITE(6,5)
5 FORMAT(1'O',11X,'COMPLEX FREQ',17X,'DETERMINANT')
CX(1)=CSUNDP
CALL ZANLYTICPLATE,1.,0.-50,4.,0.,1.,1.,CX,100.,INFER,IFF)
WRITE(6,6) CX(1)
6 FORMAT(1'RETURN FROM ITERATION',/,'COMPLEX LOC:INS',2D12.4)
CALL WDE
CSUNDP= CX(1)
200 CONTINUE
GO TO 100
999 STOP
END
SUBROUTINE MODE
IMPLICIT REAL*8(A,R,D,N,H,C-Z),COMPLEX*16(C)
COMMON /MAT/,C MAT(25,25),CMATY(25,25),CHOM(50,10),CHOM(10,25),
NPTCHS,NDIM1,NDIMCJ,NDIMCJ,NORD
COMMON /GENM/ XSYM,YSYM,W,NX,NY,NPRESAS(10),NPPEAS(10),NPRES1,NPREJ
DIMENSION PRX(5,5),CPXY(5,5)
DIMENSION CJ(50)
NPRES-NPRES1+NPPEJ
NPRES=NPRES1-1
NPRES=NPRES1-1
CALL HOMSLV(CMATX,NPTCHS,NPTCHS,NPRES,J-NSURS)
1          CMATY,NPTCHS,NPTCHS,NPRES,J-NDIM1,J-NDIM1,
2          CHOM,NDIM1,NDIMCJ,CMAT4,NDIMCJ,NDIM1,CJ,NORD)
NXM1=NX-1
NYM1=NY-1
NSURS=0
DO 470 JS=1,NY
DO 450 IS=1,NXM1
J=(JS-1)*NX+IS
PRX(IS,JS)=CJ(J-NSURS)
J=J-NSURS
450 CONTINUE
J=JS*NX
1 IF(JS-NPRES(JS)=NSURS(JS+1))GO TO 460
NSURS=MIND(NSURS(JS)+1,NPRES1)
CPXY(IS,JS)=(0.,0.)
GO TO 470
460 CPXY(IS,JS)=CJ(J-NSURS)
470 CONTINUE
DO 500 IS=1,NX
DO 500 JS=1,NYM1
J=(JS-1)*NX+IS
CPXY(IS,JS)=CJ(NPRES-JS+1)
500 CONTINUE
NXM1=NX-1
NYM1=NY-1
NSURS=0
DO 510 JS=1,NX
DO 510 IS=1,NYM1
J=(JS-1)*NX+IS
1 IF(IS-NPRES(JS)=NSURS(JS+1))GO TO 510
CPXY(IS,JS)=(0.,0.)
NSURS=MIND(NSURS(JS)+1,NPRES1)
GO TO 510
510 CPXY(IS,JS)=CJ(NPRES-JS+1-NSURS)
530 CONTINUE
WRITE(6,16)
16 FORMAT(1000000*NATURAL MODE******/,'OX-DIRECTED CURRENT*')
DO 540 I=1,NX
WRITE(6,17) (CPXY(I,J),J=1,NY)
540 CONTINUE
WRITE(6,18)
18 FORMAT('OX-DIRECTED CURRENT*')
DO 550 I=1,NX
WRITE(6,17) (CPXY(I,J),J=1,NY)
550 CONTINUE
WRITE(6,19)
19 FORMAT('OHOMOGENEOUS EXPANSION CEFS**')
WRITE(6,17) (CJ(2*NPRES1+I),I=1,NPRES1)
RETURN
END
COMPLEX FUNCTION C^L (CSINV1)

DETERMINANT EVALUATION ROUTINE FOR HALLEN-TYPE AUGMENTED MOMENT

MATLAB FOR THE THIN PLATE SCATTERER

FOR S.E.M APPLICATIONS

BY L.W. FEDERSON, A.R.

IMPLICIT COMPLEX(16,C, REAL*8, D, H, C, Z)

COMMON /GEOM/, XSYM, YSYM, W, NX, NY, IPPFAS(10), NPREI, NPREJ

COMMON /MAT/, CMATX(25,25), CMATY(25,25), CMATM(150,101), CMAT4(10,25),

NPTCHS, NDMI, NDMC, NDMCJ, NOPR

REAL*8 ORG, DIMARG, DIFRFAS(20), DIMFAS(20), DUM(20), DUM(20), DUM(20)

DIMENSION (INTER(10,10), CMATX(25,25), CMATY(25,25), CRSTY(10), CIPTM(10)

INTER(10,10), SYMM(3), ETAT1, IC1, IC1, IANTI, SYMM, ETAT1, IC1/

DATA C/(3.008, 0.00), PLUS/1/3/PT/3.141592653589793/

NDM1=25

NDM1=50

NDM1=12

NDM1=50

FORMATION SETUP ROUTINES

TX=1

IY=1

IF(XSYM, NY, PLUS) TMX = 2

IF(YSYM, NY, PLUS) TMY = 2

IM(X/Y) = 2 INDICATES ANTISYMMETRIC

DISTR OF X-DIRECTED CURR WRT X

/Y AXIS

NTPCHS=NX*NY

NXM1=NX-1

NYM1=NY-1

EDGAT = 0.5

EDGAT = EDGAT*EDGAT

DX = 1./FLOAT(2*NX-2)*2*EDGAT

DY = W/FLOAT(2*NY-2)*2*EDGAT

VXT = 2*NX-2

VYT = 2*NY-2

WS = CSNUM/2/*

IMPTS = 13

DXAT = DX/12

DYAT = DY/12

SYMXX=-1.0*TMX

SYMXY=-1.0*TMY

NSYM1 = NSYM

NSYM1 = NSYMX*NSYM

NSMYV = NSYM

SYMXX = 2

TP(NSYM1 GT 0) NINDX = 1

NINDX = 1 INDICATES EVEN-EVEN OR ODD-ODD SYMMETRY FOR X-DIR CURR

NSCC1=1

IF(XSYM, EQ, PLUS) NSCCS = 2

47
VSXS = 2 INDICATES EVEN SYMM WRT Y FOR X DIR CURR (1 E COSINE EXPAN
NSION OF HOMOGENEOUS SPLINE)

NPRE = NPREI + NPREJ

TOT NO OF PREASSIGNED CURVE VALS

NPREJ = NPREJ - 1

NPREI = NPREI - 1

NPRE1 = NPRE + 1

END OF INPUT/SPECIFICATION ROUTINES

ROUTINE TO FILL INTERACTION MATRIX FROM WHICH MOMENT MATRIX IS
CONSTRUCTED

DIAG = DSORT(DX,DX,DY,DY)

ALNTH = 2*LOG((1 + (DY/DO))/DX)

ALNYM = 2*LOG((1 + (DX/DO))/DY)

DYSH = DO/DO

DXSH = DO/DO

CSX = CS*DX

CSY = CS*DY

CSX2 = CSX*CSX

CSX3 = CSX*CSX2

CSX4 = CSX*CSX3

CSX5 = CSX*CSX4

CSX6 = CSX*CSX5

CSY2 = CSY*CSY

CSY3 = CSY*CSY2

CSY4 = CSY*CSY3

CSY5 = CSY*CSY4

CSY6 = CSY*CSY5

COMPONENT TERMS FOR SELF-PATCH SHEETS

CXTERM = -0.500*CSX*CSX*ALNTH + 0.500*CSX2*DYSH*DYSH - CSX3*IDOXY*DIAG(12*

10X) + DIANXH/24) + CSX4*DIANXH/24)*IDOXY*IDOXY/3/24

CYTERM = -0.500*CSY*ALNTH + 0.500*CSY2*DYSH*DYSH - CSY3*IDOXY*DIAG(12*

10X) + DIANXH/24) + CSY4*DIANXH/24)*IDOXY*IDOXY/3/24

CALC INDIV SERIES FOR SELF-INTER

CINTER(1,1) = -2/CS*(CXTERM + CYTERM)

COMPUTE SELF-INTERACTION

DO 220 IS = 1,2

X = (FLAT(I) - 1.500)*DX

DO 220 JS = 1,2

LOOP TO CALC ADJACENT PATCH INTER

Y = (FLAT(J) - 1.500)*DY

DO 210 J = 1, INTPTS

XP = FLAT(I) + INT(-1)*DXINT

X2TM2 = 1 + XP

X2TM2*X2TM2*Y2TM2

DO 720 J = 1, INTPTS

YP = FLAT(J) + INT(-1)*DYINT

X2TM2*Y2TM2

* = DSORT(X2TM2 + Y2TM2)

* = (INTY(JINT) * COSXP(-2*CS*XP)) / D

EVAL INTEGRAND

CONTINUE

CALL OWEQDL(1,INTX,INTPTS,OYINT,INTX(1,INT))

CONTINUE

INT WRT Y TO YIELD X INTEGRAND

CONTINUE

CALL OWEQDL(INTX,INTPTS,DXINT,CINTER(IS,JS))

CONTINUE

INT WRT X

48
**DO 250 IS=1,NX**

X2TM2=DFLOAT((IS-1)*DX)

X2TM2=X2TM2*X2TM2

**DO 250 JS=1,NY**

**LOOPS FOR REMAINDER OF INTERACTION N CALCULATED BY ONE-DIM RECTANG RULE**

IF(IS+JS)LT.4.OR.IS.EQ.2.AND.JS.EQ.2) GO TO 250

Y2TM=FLOAT((JS-1)*DY)

Z=DSORT(X2TM2-Y2TM*Y2TM)

CONTINUE

**END OF LOOP TO FILL INTERACTION MATRIX**

**250 BEGIN CONSTRUCTION OF MOMENT MATRIX**

**DO 350 IM=1,NX**

**DO 350 JN=1,NY**

**INDEXING OF MATCH-POINTS OVER ENTIRE QUADRANT**

**ONE-DIM MATCH-PT INDEX GENERATED**

CALLWISE ALONG X-DIRECTION

**INDEX OVER SOURCE PATCHES Y-DIF**

**1ST AND 2ND QUAD J 'DIFFERENCE INDEX**

**3RD & 4TH QUAD J 'DIFFERENCE INDEX**

**NOTE THAT 'DIFFERENCE INDICES' ARE E = 'INDEX DIFFERENCE' + 1 FOR THE SENSE OF FORTRAN INDEXING**

**INDEX OVER SOURCE PATCHES X-DIF**

**1ST & 4TH QUAD 'DIFF INDEXX**

**2ND & 3RD QUAD 'DIFF INDEX**

**OVER-DIM SOURCE-PT INDEX**

**SUM OF SOURCE CNT FROM Q1 & Q1I**

**SUM OF SOURCE CNT FROM QII & QIV**

**SUMMAT ENTRY FOR X-DIR CURV**

**SUMMAT ENTRY FOR Y-DIR CURV**

**NOTE THAT EVEN O/S CNT NEGATIVE**

**END OF SOURCE LOOP FOR INTERFERENCE PATCHES**

**CONSTRUCTION OF SOURCE TERMS FROM 2PS(X)#6 EDGE FOLLOWS**

**DO 310 IS=1,NX**

**CONTINUE**

310 END
CJ=CINTER(ID1,JD1)+NSMIT*INTER(ID2,JD2)
CJ=NSMIT*CINTER(ID2,JD1)+NSMV*INTER(ID1,JD2)
CJ=SUM OF SOURCE CONT FROM UI & QIII
CJ+CINTER(ID2,JD1)+NSMV*INTER(ID1,JD2)
CJ=SUM OF SOURCE CONT FROM QIII & QIV
CMATY(I,J)=(CC-CF)*EDGFA
CMATY(I,J)=SUMMAT ENTRY FOR Y-DIR CURR'S
IF(JS,NE,JPREAS(VSURS+1)) GO TO 325
NSURS=NSURS+1,NPREJM)
GO TO 330
325 CMATY(I,J-NSURS)=(CF+CC)*EDGFA
CMATY(I,J-NSURS)=SUMMAT ENTRY FOR X-DIR CURR'S
END ROUTINE FOR ABS(X)=1 EDGE TERMS
330 CONTINUE
END LOOP OVER Y COORD FOR INTERIOR PATCHES
BEGIN ROUTINE FOR CONSTRUCTION OF SOURCE TERMS FOR ABS(Y)=1 EDGE
J01=18AS(NV-JM)+1
J02=NY+JM
NSURS=NSURS+1
VSURS=0
07 340 (S=1,NX)
INDEX DOWN X COORD'S INTERIOR PATCHES
T01=18AS(NX-1M)+1
T02=TS+1W
J=(NVMT)*NX+1S
CJ=CINTER(ID1,JD1)+NSVIII*INTER(ID2,JD2)
CJ=NSVIII*CINTER(ID2,JD1)+NSMV*INTER(ID1,JD2)
CJ=SUM OF SOURCE CONT FROM QIII & QIV
CJ=SUM OF SOURCE CONT FROM QIV & QIV
CMATX(I,J-NSURS)=(CF+CC)*EDGFA
CMAX(I)I,J-NSURS=(CC-CF)*EDGFA
SUMMAT ENTRY FOR X-DIR CURR'S
IF(JS,NE,JPREAS(NSURS+1)) GO TO 335
VSURS=NSURS+1,NPREJ)
GO TO 340
335 CMATX(I,J-NSURS)=(CC-CF)*EDGFA
CMATX(I,J-NSURS)=SUMMAT ENTRY FOR X-DIR CURR'S
NOTE THAT EVEN QIV CONT NEGATIVE FOR Y-DIR CURR'S
340 CONTINUE
END ROUTINE FOR ABS(Y)=1 EDGE
CONSTRUCTION OF CORNER SOURCE TERM
I01=18AS(NX-1M)+1
I02=NX+1W
J=NX+MV
CJ=CINTER(ID1,JD1)+NSVIII*INTER(ID2,JD2)
CJ=NSVIII*CINTER(ID2,JD1)+NSMV*INTER(ID1,JD2)
CJ=SUM OF SOURCE CONT FROM QIII & QIV
CJ=SUM OF SOURCE CONT FROM QIV & QIV
IF(NY,NE,JPREAS(NPREJ)) CMATX(I,J-NPREJ)=(CF+CC)*EDGFA
SUMMAT ENTRY FOR X-DIR CURR'S
IF(NX,NE,JPREAS(NPREJ)) CMATX(I,J-NPREJ)=(CC-CF)*EDGFA
3390
03400
03410
03420
03430
03440
03450
03460
03470
03480
03490
03500
03510
03520
03530
03540
03550
03560
03570
03580
03590
03600
03610
03620
03630
03640
03650
03660
03670
03680
03690
03700
03710
03720
03730
03740
03750
03760
03770
03780
03790
03800
03810
03820
03830
03840
03850
03860
03870
50
350 CONTINUE
END DF Moment Matrix Interaction Construction
BEGIN Routine to Enter Homogeneous Solution Expansion Coefs in Matrix
360 NRES=2*NPPE
IF(MINDEX.EQ.2) NRES=NRES-1
D7 400 IM=1,4X
X=(FLOAT(IM)-0.500)*DX
D7 400 JN=1,NY
Y=(FLOAT(JN)-0.500)*DY
I=(JN-1)*NX+NY
INDEXING THRU MATCH-PTS
PHT=ATAN(Y/X)
R-PS=360*(X*Y*Y)
DSEG=2*DIMAG(CS)*PHI
DTMARG=2*REAL(CS)*PHI
ARG=FLOAT(MINDEX/400)+1.E-2" GO TO 364
IF FEAL ARG SKIP TO FEAL RES CALL
CALL ASCLZ(DPARG, DPIMARG, NRES, DIMRES, NPRES, 0, 0, 0, 0, 0, 0, 0)
1=UM3, DIM4)
GET TABLE OF RESEL FUNCTIONS
C 364 GO TO 368
CALL ASCLZ(DPARG, DPIMARG, NRES, NPRES, 0, 0, 0, 0, 0, 0, 0)
C 368 CONSTM(1)=0
CSTNM(1)=0
DO 370 IT=1,NPPEP1
INDEX=2*IT-NINDEX
IF(INDEX.EQ.0) GO TO 370
INDEX THRU CALF OF COEF CONSTR VECTOR
CALF SERIES INDEX
SKIP CALF OF BFORE TERM FOR ZERO INDEX - IT WAS SET TO ZERO ABOVE
ARG=FLOAT(MINDEX-1)*PHI
C 370 CONTINUE
DO 380 JJ=1,NPPEJ
INDEX=2*JJ-NINDEX
C 380 CONTINUE
...
GO TO (371, 372), NSCFS

SELECT INDEX FOR REPLACING TERM

SELECT PROPER SERIES OF FE ACCORDI
NG TO Y SYMMETRY CONDITION

NSCFS indicates COINCINE IN Y

CURRENT END

371 CHOM(I, JJ) = -PI/(2 * CS) * (0, DO, 1, DO) * TINDX(JJ) = CSTN(JJ) + 1

1 NCOSTJ(JJ + 1)

GO TO 380

372 CHOM(I, JJ) = -PI/(2 * CS) * (0, DO, 1, DO) * TINDX(JJ) = CSTN(JJ + 1)

1 NCOSTJ(JJ + 1)

1 CSTAT(JJ + 1)

380 CONTINUE

GO TO 390 [I = 1, NPREJ]

J = (NY - 1) * NX + NPREAS(I) + NPTCHS

LOOP TO REPLACE COLS FOR PREFASS!

J = TT + NPREJ

INDX = 2 * (TT + NPREFJ) - NINDX

GO TO (381, 382), NSCFS

381 CHOM(I, JJ) = -PI/(2 * CS) * (0, DO, 1, DO) * TINDX(JJ) = CSTN(TT + NPREFJ) - 1

1 CSTN(TT + NPREFJ + 1)

CHOM(I, NPTCHS, JJ) = -PI/(2 * CS) * (0, DO, 1, DO) * TINDX(JJ) = CSTN(TT + NPREFJ) + 1

1 NCOSTJ(TT + NPREFJ + 1)

GO TO 390

382 CHOM(I, JJ) = -PI/(2 * CS) * (0, DO, 1, DO) * TINDX(JJ) = CSTN(TT + NPREFJ + 1) - 1

1 NCOSTJ(TT + NPREFJ + 1)

CHOM(I, NPTCHS, JJ) = -PI/(2 * CS) * (0, DO, 1, DO) * TINDX(JJ) = CSTN(TT + NPREFJ + 1)

1 NCOSTJ(TT + NPREFJ + 1)

390 CONTINUE

400 CONTINUE

END OF MOMENT MATRIX CONSTRUCTION

405 CONTINUE

CALL SPRHOM(CWATX, NPTCHS, NPTCHS - NPREJ, NOMP, 1, NIMP, 1)

1 CWATX, NPTCHS, NPTCHS - NPREJ, NOMP, 1, NIMP, 1

2 CWATX, NPTCHS, NPTCHS - NPREJ, NOMP, 1, NIMP, 1, CNFET

FRAT = NABS(CWATX(1, 1))

CPLATE = CNFET / FRAT

WRITE(6, 20) CNFORD, CPLATE

20 FORMAT(1E14, 95X, 2F12.4, 5X, 2F12.4)

RETURN

END

52
SUBROUTINE SPPHOM(CMAT1,NI1,NJ1,NDIM1I,NDIM1J,CMAT2,NI2,NJ2,NDIM2I,NDIM2J,CMAT3,NI3,NJ3,NDIM3I,NDIM3J,CMAT4,NDIM4I,NDIM4J,CDET)

IMPLICIT COMPLEX*(C),REAL*(A,B,D-H,C-Z)

C
SUBROUTINE TO DIAGONALIZE AND CALC DETERMINANT OF A SPARSELY-COUPLED MATRIX
BY L W PEARSON 7/74
REVISED 5/75

DIMENSION CMAT1(NDIM1I,NDIM1J),CMAT2(NDIM2I,NDIM2J),CMAT3(NDIM3I,NDIM3J),CMAT4(NDIM4I,NDIM4J)
N3=NI1+NI2
NJ3=NI3-NJ2-NJI
CALL ZEROZ(CMAT4,4*NDIM4I*NDIM4J)
CDET=1

INITIALIZE PRODUCT ACCUMULATOR

NPR=3
NJ1M1=NJ1-1
NJ1L=NJ1
IF(NJ2*NJ3.GE.1) GO TO 95
NJ1L=NJ1L-1
NPR=1

DO 155 M=1,NJ1L

INDEX ACROSS COL

MP1=M+1
FMAX=CDABS(CMAT1(M,M))
K=M
IF(MP1.GT.NI1) GO TO 105
DO 100 I=MPI,NI1

LGCF TO SEARCH FOR PIVOT IN MTH COL

FCK=CDABS(CMAT1(I,M))
IF(FCK.LE.FMAX) GO TO 100
K=I

IF LARGER ELEMENT FOUND MARK ROW

FMAX=FCK
USE NEW LARGE ELEMENT AS COMPARISON VALUE

100 CONTINUE

SAVE VAL OF PIVOT ELEMENT
MULT PIVOT INTO PROD ACCUMULATOR
IF PIVOT CN DIAG SKIP RCW EXCH
CHANGE SIGN BECAUSE OF RCW EXCH

DO 110 J=M,NJ1

LOOP TO EXCH DIAG AND PIVOT ROWS

CSTC=CMAT1(K,J)
CMAT1(K,J)=CMAT1(M,J)
CMAT1(M,J)=CSTC

CONTINUE

IF(NJ3.LT.1) GO TO 115
DO 112 J=1,NJ3
CST=CMAT3(K,J)
CMAT3(K,J)=CMAT3(M,J)
CMAT3(M,J)=CST

CONTINUE

115 CONTINUE

IF(MP1.GT.NI1) GO TO 155

53
DO 150 I=MP1,N1
C CFC=C MAT1(I,M)/CSTOR
C IF(MP1.GT.NJ1) GO TO 125
DD 120 J=MP1,NJ1
C CMAT1(I,J)=CMAT1(I,J)-CMAT1(M,J)*CFC
120 CONTINUE
IF(NJ3.LT.1) GO TO 150
125 DD 130 J=1,NJ3
C CMAT3(I,J)=CMAT3(I,J)-CMAT3(M,J)*CFC
130 CONTINUE
150 CONTINUE
155 CONTINUE
NI4=NI1-NJ1
IF(NI4.LE.0) GO TO 290
C
C BEGIN ROUTINE TO CREATE/"DIAGONALIZE" CMAT4

C NPIV=NI4
IF(NI4.GT.NJ2) NPIV=NJ2
DO 250 M=1,NPIV
C MP1=M+1
FMAX=CCABS(CMAT2(I,M))
K=1
IF(NI2.LT.2) GO TO 205
DO 200 I=2,NI2
C FCK=CDABS(CMAT2(I,P))
IF(FCK.LE.FMAX) GO TO 200
205 CONTINUE
200 CSTOR=CMAT2(K,M)
C CDCT=CDET*CSTOR
C CDCT=-CDCT
C DO 210 J=M,NJ2
C CSTO=CMAT4(M,J)
CMAT4(M,J)=CMAT2(K,J)
CMAT2(K,J)=CSTC
210 CONTINUE
K3=K+NI1
M3=NJ1+M
IF(NJ3.LT.1) GO TO 213
DO 212 J=1,NJ3
C CSTO=CMAT3(K3,J)
CMAT3(K3,J)=CMAT3(M3,J)
CONTINUE
CMAT3(M3,J)=CST0

212 CONTINUE
213 IF(NI2.LT.1) GC TC 290
235 DO 250 I=1,NI2

C

I3=NI1+1
CFAC=CMAT2(I,M)/CSTOR
IF(MP1.GT.NJ2) GC TC 242
DO 240 J=MP1,NJ2

C

CMAT2(I,J)=CMAT2(I,J)-CMAT4(M,J)*CFAC

240 CONTINUE
242 DO 245 J=1,NJ3

C

CMAT3(I3,J)=CMAT3(I3,J)-CMAT3(M3,J)*CFAC
245 CONTINUE
250 CONTINUE

C

END ROUTINE TC 'DIAGONALIZE' CMAT4
290 IF(NI4.GE.NJ2) GC TC 250

C

IF DIAGONAL DOES NOT PASS THRU SKIP DIAGONALIZATION FOR CMAT2
C

C

BEGIN ROUTINE TC 'DIAGONALIZE' CMAT2
C

NI4P1=NI4+1
NJ2L=NJ2
IF(NJ3.GE.1) GC TC 295
NJ2L=NJ2L-1
NPR=2

295 DO 350 M=NI4P1,NJ2L
MI=M-NI4
M3=MI+NI1
MP1=M+1
MP1=M+1
FMAX=COABS(CMAT2(M,P))
K=MI
IF(MP1.GT.NI2) GC TC 305
DO 300 I=NI1,NI2

C

FCK=COABS(CMAT2(I,P))
IF(FCK.LE.FMAX) GO TO 300
K=1

C

FMAX=FCK
IF LARGER ELEMENT FOUND MARK ROW
C

300 CONTINUE
305 CSTOR=CMAT2(K,P)

C

K3=K+NI1
CDET=CDET*CSTOR
SAVE VAL OF PIVOT ELEMENT
C

IF(K.EQ.MI) GC TC 315

C

CDET=-CDET
MULT PIVOT INTO PROD ACCUMULATOR
C

CHANGE SIGN BECAUSE OF ROW EXCH
DO 310 J=M, NJ2
CSTO=CMAT2(K, J)
CMAT2(K, J)=CMAT2(MI, J)
CMAT2(MI, J)=CSTO
310 CONTINUE
IF(NJ3 LT 1) GO TC 315
DO 312 J=1, NJ3
C
CSTO=CMAT3(K3, J)
CMAT3(K3, J)=CMAT3(M3, J)
CMAT3(M3, J)=CSTO
312 CONTINUE
IF(MI1 GT NI2) GO TO 390
DO 330 I=MI1, NI2
I3=I+NI1
CFAC=CMAT2(I, M)/CSTOR
IF(MI1 GT NJ2) GO TO 335
DO 330 J=M, NJ2
C
LOCP FOR EXCH IN CMAT3
C
CMAT2(I, J)=CMAT2(I, J)-CMAT2(MI, J)*CFAC
330 CONTINUE
IF(NJ3 LT 1) GO TC 350
DO 345 J=1, NJ3
C
LOOP ACROSS ROW IN CMAT3
C
CMAT3(I3, J)=CMAT3(I3, J)-CMAT3(M3, J)*CFAC
345 CONTINUE
350 CONTINUE
C BEGIN ROUTINE TO 'DIAGONALIZE' CMAT3
C
390 NJ3M1=NJ3-1
IF(NJ3M1 LT 1) GO TC 455
DO 450 M=1, NJ3M1
C INDEX ACROSS COL
C
MP1=M+1
MI=M+NI1+NJ2
MIP1=MI+1
FMAX=CDABS(CMAT3(MI, M))
K=MI
IF(MI1 GT NJ3) GO TO 405
DO 400 I=MPI1, NI3
C
LOCP TO SEARCH FOR PIVCT IN MTH COL
C
FCK=CDABS(CMAT3(I, M))
IF(FCK LE FMAX) GO TO 400
K=I
C
FMAX=FCK
C
IF LARGER ELEMENT FOUND MARK ROW COL
C
USE NEW LARGE ELEMENT AS COMPARE VALUE
C
400 CONTINUE
405 CSTOR=CMAT3(K, M)
C
COET=COET*CSTOR
C
IF(K EQ MI) GO TO 415
C
COET=-COET
C
CHANGE SIGN BECAUSE OF ROW EXCH
C
56
DO 410 J=M,NJ3
   CSTO=CMAT3(K,J)
   CMAT3(K,J)=CMAT3(MI,J)
   CMAT3(MI,J)=CSTO
   CONTINUE
410 CONTINUE
   DO 450 I=MPI1,NI3
   CFAC=CMAT3(I,M)/CSTOR
   CONTINUE
450 CONTINUE
   DO 445 J=MPI1,NJ3
   CMAT3(I,J)=CMAT3(I,J)-CMAT3(K,J)*CFAC
   CONTINUE
445 CONTINUE
455 GO TO (461,462,463), AFR
461 CDET=CDET*CMAT1(N1,N1)
   RETURN
462 CDET=CDET*CMAT2(N12,N12)
   RETURN
463 CDET=CDET*CMAT3(N13,N13)
   RETURN
C END

LOOP TO EXCH DIAG AND PIVOT RCWS 02390
ELIMINATION LOOP 02500
LOOP ACROSS RCW IN CMAT3 02530
MULT LAST ELEMENT INTO DETERM 02590

02400
02410
02420
02430
02440
02480
02490
02500
02510
02520
02530
02540
02550
02570
02572
02574
02576
02578
02582
02584
02600
02610
SUBROUTINE SPRSLV(CMAT1,N1,NJ1,NDIM1,NDIM1J,CMAT2,N12,NJ2,NDIM2J)
    DIMENSION CMAT1(NDIM1,NDIM1J),CMAT2(NDIM2J),CMAT3(NDIM3),CMAT4(NDIM4J,CRHS,CSCIN)
SUBROUTINE TO BACKSOLVE A TRIANGULARIZED SYSTEM OF SPARCELY-COUPLED LINEAR EQUATION
BY L W PEARSON 7/74
REVISED 5/75
STORAGE FORM COMPATIBLE WITH THE TRIANGULARIZATION ROUTINE SPARCE
THE ENTRY 'HOMSLV' BELOW ALLS THE SOLUTION FOR NATURAL VECTORS OF HOMOGENEOUS SYSTEMS PROVIDED THE DETERMINANT OF THE SYSTEM IS ZERO
IMPLICIT COMPLEX*16(C),REAL*8(A,B,O-H,C-Z)
DIMENSION CMAT1(NDIM1J),CMAT2(NDIM2J),CMAT3(NDIM3),CMAT4(NDIM4J,CRHS,CSCIN)
LOGICAL LHOM
SETUP FOR INHOMOGENEOUS SYSTEM
LHCM=.FALSE.
NI3=NI1*NI2
NJ3=NI3-NJ1-NJ2
NI4=NI1-NJ1
ND2=NJ2-NI4
NPR=3
IF(NJ3.LT.1)NPR=2
IF(NJ3+NJ2.LT.1)NPR=1
GO TO (81, 82, 83), NPR
81 CSCLN(NI3)=CRHS(NI3)/CMAT1(NI1,NJ1)
GO TO 100
82 CSCLN(NI3)=CRHS(NI3)/CMAT2(NI2,NJ2)
GO TO 100
83 CSCLN(NI3)=CRHS(NI3)/CMAT3(NI3,NJ3)
GO TO 100
SOLVE FOR 'LAST' UNKNOWN
GO TO 100
END OF SETUP FOR INHOM SYSTEM
BEGIN ENTRY/SETUP FOR HOMOGENEOUS SYSTEM
ENTRY HOMSLV(CMAT1,N1,NJ1,NDIM1,NDIM1J,CMAT2,N12,NJ2,NDIM2J,NDIM2)
ENTRY ACIPJ,5(CMAT1,NDIM1,ACIPJ3J,CMAT4,NDIM4,J,CSOLN,NORD)
LHCM=.TRUE.
LOGICAL INDICATOR FOR HOMOGEN SYSTEM
NI3=NI1+NI2
NJ3=NI3-NJ1-NJ2
NI=NI1-NJ1
ND2=NJ2-NI4
CSOLN(NI3)=1

ASSIGN ARBITRARY ELEMENT IN SCLN

END SETUP FOR HOMOGENEOUS ENTRY

BEGIN BACKSOLVE FOR EQUATIONS INVOLVING ONLY CMAT3 (LAST NJ3 EQS)

100 FMAX=CCABS(CSOLN(NI3))
IMAX=NI3
IF(NJ3.LT.2) GC TC 200

SKIP ROUTINE IF ONLY LAST VARIABLE COUPLES (IT WAS SOLVED/ASSIGNED ABOVE)

DO 150 IC=2,NJ3
ICM1=IC-1
I=NI3-IC+1
I=NI3-IC+1

CALC MATRIX ROW INDX FROM COMPLEMENTARY INDX

JD3=I-.AJ1-NJ2

COL INDX FOR CMAT3 WHICH DEFINES DIAG CF MATRIX

CSUM=0
DO 110 J3C=1,ICM1

LOOP TO ACCLM NEGATIVE SUM CF PREVIOUSLY CALC'D UNKNS

J3=NJ3+1-J3C
J=NJ3+1-J3C

COL OF COEF IN CMAT3

CSUM=CSUM+CMAT3(I,J3)*CSOLN(J)

CONTINUE

IF(.NOT.LHOM) CSUM=CSUM+CRHS(I)

ADD R H S TO SUM

CSOLN(I)=CSUM/CMAT3(I,JD3)

DIVIDE BY DIAG COEF

IF(CDABS(CSOLN(I))*.LE.FMAX) GO TO 150

FMAX=CDABS(CSOLN(I))
IMAX=1

CHECK FCR MAX ELEMENT

CONTINUE

BEGIN ROUTINE TC SCLVE FOR ELEMENTS INVOLVING CMAT3 & CMAT2

200 IF(NJ3.GE.NI2) GO TO 300

SKIP ROUTINE IF DIAG DOES NOT PASS THRU CMAT2

DO 250 IC=1,NC2
ICM1=IC-1
I2=NJ2-NJ3+1-IC
I3=NJ3-NJ3+1-IC
JD2=NJ2+1-IC
NCM1=NJ3+IC-1
CSUM=0
IF(NJ3.LT.1) GO TO 215

LOOPE TO SUM CONTRIB FROM CMAT3

59
BEGIN ROUTINE TO SOLVE FOR ELEMENTS INVOLVING CMAT3 & CMAT4

300 IF(NI4.LT.1) GO TO 400
DO 350 IC=1,NI4
   I4=NI4+1-IC
   J04=I4
   I3=NI1+1-IC
   CSUM=0
   IF(INJ3.LT.1) GO TO 315
   DO 310 J3C=1,NJ3
      J3=NI3+1-J3C
      J=M3-J3C
      CSUM=CSUM-CMAT3(I3,J3)*CSOLN(J)
      310 CONTINUE
      315 NSUBS=NSUBS+IC-1
      IF(NSUBS.LT.1) GO TO 325
      DO 320 J4C=1,NSUBS
         J4=NI2+1-J4C
         J=NI3+1-J4C
         CSUM=CSUM-CMAT4(I4,J4)*CSOLN(J)
      320 CONTINUE
      325 IF(.NOT.LHOM) CSUM=CSUM+CMAT3(I3)*CSOLN(I3)
      CONTINUE
      IF(CDABS(CSOLN(I3)).LE.FMAX) GO TO 350
      FMAX=CDABS(CSOLN(I3))
      IMAX=I3
      350 CONTINUE
   C
   BEGIN ROUTINE TO SOLVE FOR ELEMENTS INVOLVING CMAT3 & CMAT1

400 IF(NJ1.LT.1) GO TO 455
   DO 450 IC=1,NJ1
      I=NI1+1-IC
      ICMI=IC-1
      CSUM=0
      IF(INJ3.LT.1) GO TO 415
      DO 410 J3C=1,NJ3
         J3=NI3+1-J3C
         J=M3-J3C
         CSUM=CSUM-CMAT3(I3,J3)*CSOLN(J)
      410 CONTINUE
      415 IF(ICML.LT.1) GO TO 425
DO 420 JC=1,ICM1
   J=NJ1+1-JC
   CSUM=CSUM+CMAT1(I,J)*CSOLN(J)
420 CONTINUE
425 IF(.NOT.LHOM) CSUM=CSUM+CMAT(I)
   CSOLN(I)=CSUM/CMAT1(I,I)
   IF(CDABS(CSOLN(I)) .LE. FMAX) GO TO 450
   FMAX=CDABS(CSOLN(I))
   IMAX=I
450 CONTINUE
C
C END OF SOLUTION
C
455 IF(.NOT.LHOM) RETURN
RETURN IF INHOM SYSTEM
C
C BEGIN NORMALIZATION ROUTINE FOR NATURAL VECTCR FOR HCMCGENEUS
C CASE
C
   CSSCALE=1./CSOLNI(MAX)
   DO 500 I=1,NI3
      CSOLN(I)=CSOLN(I)*CSSCALE
500 CONTINUE
C
RETURN
C
END
SUBROUTINE COPYZ(X, Y, N)
DIMENSION X(I), Y(I)
DO 100 I=1, N
X(I) = Y(I)
100 CONTINUE
RETURN
END
SUBROUTINE ZEROZ(ARRAY,N)
DIMENSION ARRAY(1)
DO 100 I=1,N
   ARRAY(I)=0
100 CONTINUE
RETURN
END
SUBROUTINE ONEDDL(FCN,N,DELTA,VINT)
IMPLICIT REAL*8(A-H,O-Z)
COMPLEX*16 FCN,C,VINT
DIMENSION FCN(N)
DIMENSION COEF(6)
DATA COEF/2.00,5.00,1.00,6.00,1.00,5.00/
IF((N-1)/6*6.E0,N-1) GO TO 100
WRITE(6,1)
1 FORMAT('INCORRECT POINTS TO ONEDDL')
A=1/0
100 CONTINUE
VINT=0
DO 200 J=1,N
JCROW=J-((J-1)/6)*6
VINT=VINT+COEF(JCROW)*FCN(J)
200 CONTINUE
VINT=(VINT-FCN(1)-FCN(N))*0.300+0.00)*DCMPLX(DELTA,0.00)
RETURN
END
FUNCTION

- DETERMINATION OF ZEROS OF AN ANALYTIC COMPLEX

FUNCTION USING MULLER'S METHOD WITH

DEFLATION

USAGE

- CALL ZANLYT (F, EPS, NSIG, KN, NGUESS, N, X, ITMAX, INFEP, IER)

PARAMETERS

F - A FUNCTION SUBPROGRAM, F(Z), WRITTEN BY THE

USER SPECIFYING THE EQUATION WHOSE ROOTS

ARE TO BE FOUND. F MUST BE TYPE-NAMED AS

FOLLOW - COMPLEX FUNCTION F(16)(Z)

EPS - 1ST STOPPING CRITERION. A ROOT Z IS ACCEPTED IF

ABSOLUTE VALUE OF F(Z) ≤ EPS (INPUT) ZAN09780

NSIG - 2ND STOPPING CRITERION. A ROOT IS ACCEPTED IF

TWO SUCCESSIVE APPROXIMATIONS TO A GIVEN

ROOT ARE IN THE FIRST NSIG DIGITS. (INPUT) ZAN09820

KN - THE NUMBER OF KNOWN ROOTS WHICH MUST BE STORED ZAN09860

TO X(1), ..., X(KN), PRIOR TO ENTRY TO ZANLYT ZAN09870

NGUESS - THE NUMBER OF INITIAL GUESSES PROVIDED. THESE GUESS MUST BE STORED IN X(KN+1), ..., ZAN09880

X(KN+NGUESS) AND NGUESS MUST BE SET EQUAL TO ZERO IF NO GUESSES ARE PROVIDED. (INPUT) ZAN09900

N - THE NUMBER OF NEW ROOTS TO BE FOUND BY

ZANLYT (INPUT)

X - A LONG-WORD COMPLEX VECTOR ARRAY OF LENGTH 1 = X(1), ..., X(KN), X(1), ..., X(KN) ON INPUT ZAN09950

MUST CONTAIN ANY KNOWN ROOTS. X(KN+1), ..., X(KN+NGUESS) ON INPUT MAY, AT THE USER'S OPTION, CONTAIN INITIAL GUESSES FOR THE N NEW ROOTS WHICH ARE TO BE COMPUTED. ON OUTPUT, X(KM+1), ..., X(KM+N) CONTAIN EITHER A ROOT CORRECT TO WITHIN A CONVERGENCE CRITERON OR THE VALUE (12, 5678), (12, 5678) ≠ 0) INDICATIVE OF A FAILURE TO ACHIEVE THE SPECIFIED CONVERGENCE FOR THAT ROOT, SAY X(KM+J). IN THE LATTER CASE, THE MOST RECENT APPROXIMATION TO X(KM+J) IS AVAILABLE IN X(ISUB), WHERE ISUB = K(KM+J)+1 ZAN09970

ITMAX - THE MAXIMUM ALLOWABLE NUMBER OF ITERATIONS PER ROOT (INPUT) ZAN10080

INFEP - AN INTEGER VECTOR OF LENGTH CF, KN+N+1. ON OUTPUT INFEP(J) CONTAINS THE NUMBER OF ITERATIONS USED IN FINDING THE J-TH ROOT WHEN CONVERGENCE WAS ACHIEVED. IF CONVERGENCE WAS NOT ACHIEVED IN ITMAX ITERATIONS, INFEP(J) WILL Contain ITMAX+1 (OUTPUT)

IER - ERROR PARAMETER (OUTPUT)

CALL ZANLYT (F, EPS, NSIG, KN, NGUESS, N, X, ITMAX, INFEP, IER)

WARNING: ERROR = 32 + N

M = 1 FAILURE TO CONVERGE WITHIN ITMAX ITERATIONS FOR ONE OF THE (N) NEW ROOTS TO BE FOUND

PRECISION - DOUBLE ZAN10180

REIDO IMSL ROUTINES - VEG12

AUTHOR/IMPLEMENTOR - P. G. JOHNSON/L. L. WILLIAMS

LANGUAGE - FORTRAN

UPDATE: SEPTEMBER 1, 1971

56
C
SUBROUTINE ZANLYT (F,FPS,NSIG,KN,NGUESS,N,X,ITMAX,INFER,IER)
INTEGER*4 1
DOUBLE PRECISION X(1),ONE,D,DO,DEN,DI,FPRF,FRT,
            OZ,FPS,FPS1,INFER(1)
IER = 0
ONE = (1.00+00,0.00+00)
FPS1 = 10.00+00**(-NSIG)
ICONJ = 0
ITMAX = 0
SET NUMBER OF ITERATIONS

C
MR1 = KN+1
MR2 = KN+N
LSTART = MR2+1
MPG = MR1+NGUESS
DO 2 I = MPG,MR2
2   X(I) = (O.0+0.0,0.0+0.0)
L = MR1
IF (KN .EQ. 0) GO TO 5
DO 3 I = 1,KN
3   INFER(I) = 0
      TEMP = MB2+I
      X(TEMP) = X(I)
      ITEMP = MA2+ITEMP
3   X(TEMP) = X(I)
   5 JK = 0
   QZ = CADB(X(L))
   IF (QZ .LE. 1.00-15) GO TO 25
   POCTFSTIM/JTF
   ROOT ESTIMATE NOT EQUAL TO ZERO
   10 RT = (-90+00,0.00+00)*X(L)
       ASSIGN 15 TO NN
       GO TO 135
   15 XO = FPRT
       RT = (1.10+00,0.00+00)*X(L)
       ASSIGN 20 TO NN
       GO TO 135
   20 X1 = FPRT
       H = X(L)-RT
       RT = X(L)
       ASSIGN 40 TO NN
       GO TO 135
   25 RT = -ONE
       ASSIGN 30 TO NN
       GO TO 135
   30 XO = FPRT
       RT = ONE
       ASSIGN 35 TO NN
       G7 TO 135
   35 X1 = FPRT
6   RT = (0.00+00,0.00+00)
       H = -ONE
       ASSIGN 40 TO NN
       GO TO 135
   40 X2 = FPRT
   45 DO = (-0.50+00,0.00+00)
   BEGIN MAIN ALGORITHM
   50 T1 = X0*DO
       T2 = X1*DO*DO

ZAN10280
ZAN10290
ZAN10300
ZAN10310
ZAN10320
ZAN10330
ZAN10340
ZAN10350
ZAN10360
ZAN10370
ZAN10380
ZAN10390
ZAN10400
ZAN10410
ZAN10420
ZAN10430
ZAN10440
ZAN10450
ZAN10460
ZAN10470
ZAN10480
ZAN10490
ZAN10500
ZAN10510
ZAN10520
ZAN10530
ZAN10540
ZAN10550
ZAN10560
ZAN10570
ZAN10580
ZAN10590
ZAN10600
ZAN10610
ZAN10620
ZAN10630
ZAN10640
ZAN10650
ZAN10660
ZAN10670
ZAN10680
ZAN10690
ZAN10700
ZAN10710
ZAN10720
ZAN10730
ZAN10740
ZAN10750
ZAN10760
ZAN10770
ZAN10780
ZAN10790
ZAN10800
ZAN10810
ZAN10820
ZAN10830
ZAN10840
ZAN10850
ZAN10860
ZAN10870
ZAN10880
XX = X2*DD
T3 = X2*D
RT = T1-T2+XX+T3
DEN = RT**(-1.0*(2.0+0.0+0.0)+0.0)**((XX+T1)-T3*(T2-XX))

USE DENOMINATOR OF MAXIMUM AMPLITUDE

T1 = COSORT(DEN)
T2 = RT + T1
T3 = RT - T1
OZ = CDARS(T2) - CDARS(T3)
IF (OZ = 0.0) GO TO 60
55 DEN = T3
GO TO 65

60 DEN = T2

65 OZ = CDARS(DEN)
IF (OZ = GT. 1.0-15) GO TO 75
70 DEN = ONE
75 DT = ((-2.0+0.0+0.0+0.01)*XX)/DEN
H = DT + H
RT = RT + H

CHECK CONVERGENCE OF THE FIRST KIND

80 ASSIGN 85 TO MW
GO TO 135
85 OZ = CDARS(FRPT) - CDARS(X2*(1.0D0+0.0D0))
IF (OZ = LE. 0.0D0) GO TO 95

TAKE REMEDIAL ACTION TO INDUCE
CONVERGENCE

90 DT = DT*(0.50+00,0.00+00)
H = H*(0.50+00,0.00+00)
RT = RT-H
C2 = 135
95 XO = X1
X1 = X2
X2 = FRPT
D = DT
GO TO 50

A ROOT HAS BEEN FOUND

100 FRPT = F(RT)
105 XIL (L) = RT
ITEMP = MA2+L-1-ROMP
X(ITEMP) = RT
ITEMP = MA2+MA2+L
X(ITEMP) = RT

CHECK TO SEE IF COMPLEX-CONJUGATE
IS ALSO A ROOT

IF (CDARS(F(I)CONJG(X(L)))) .GT. 1.0D0+00CDARS(FRT)) GO TO 115
OZ = CDARS(X(L)-CONJG(X(L)))
IF (ICONJG .NE. 0.00. OZ .LT. 1.0D0) GO TO 115
ISTART = L+2
INSEP = L+1
X(INSEP) = X(INSEP1)

CHECK TO SEE IF COMPLEX-CONJUGATE
IS ALSO A ROOT

110 TVFRP1 = INSEP
XIL+1 = OCONJG(X(L))
TLMJ = 1
C2 = 120
115 TCONJ = 0
120 CONTINUE
125 TVFRP(L) = JK

67
L = L + 1
IF (L .LE. MR2) GO TO 5
RETURN TO CALLING PROGRAM
130 GO TO 185
135 JK = JK + 1
IF (JK .GT. ITMAX) GO TO 180
140 FPRT = F(RT)
FPRT = FPRT
C
TEST TO SEE IF FIRST ROOT IS DETERMINED
IF (L .EQ. 1) GO TO 160
IF (L .LE. ITRM2 + 1) GO TO 160
C
COMPUTE DENOMINATOR FOR MODIFIED FUNCTION
145 LTUP = MR2 + L - TRM2 - 1
DO 150 I = LSTRT, LTUP
TEM = RT - X(I)
QZ = CDARS(TEM)
IF (QZ .LT. -5.00 - 15) GO TO 175
150 FPRT = FPRT/TEM
C
CHECK CONVERGENCE OF THE SECOND KIND
160 QZ = CDARS(FPRT)
IF (QZ .GE. EPS) GO TO 170
165 QZ = CDARS(FPRT)
IF (QZ .LT. EPS) GO TO 105
170 GO TO NN, (15 + 20, 30, 35, 40, 85)
175 RT = RT * (1.00 .0000010 + 0.0000000 + 0)
GO TO 135
C
WARNING ERRORS, ITMAX = MAXIMUM
180 IFR = -53
INEFR = ITMAX + 1
TRM2 = TRM2 + 1
X(L) = (12345678, 12345678 + 0, 12345678, 12345678 + 0)
ITEMP = MR2 + MR2 + 1
X(TITEMP) = RT
L = L + 1
IF (L .LE. MR2) GO TO 5
185 IF (IFR .EQ. 0) GO TO 9005
9000 CONTINUE
CALL UERST(TFR, ZANLYT)
9005 RETURN
END
SUBROUTINE UFRSTT(IER, NAME)

DIMENSION ITYP(5,4), ITJT(4)
INTEGER*2 NAME(3)
INTEGER WARN,WARE,TERM, PRINTR
EQUVALENCE (ITJT(1), WARN), (ITJT(2), WARE), (ITJT(3), TERM)
DATA ITYP/*WARNING**, **10**, **10**, **10**
* /WARNING**, **10**, **10**, **10**
* /TERM**, **10**, **10**, **10**
* /NON-TER**
DATA ITJT /32, 64, 128, 0/
ITP2=ITF0
IF (ITP2 .GE. WARN) GO TO 5
IER1 = 6
GO TO 20
5 IF (IER2 .LT. TERM) GO TO 10
IER2 = 3
GO TO 20
10 IF (IER2 .LT. WARE) GO TO 15
IER2 = 2
GO TO 20
15 IER1 = 1
EXTRACT INT
20 IER2 = IER2-IERT(IER1)
WRITE (PRINT, 25) (ITYP(I, IER1), I=1, 5), NAME, IER2
25 FORMAT(*** I = S(LUFRSTT) *** I = 5(4,4X,3A?,4X,1?)
RETURN
END
SUBROUTINE ASLJZ(X,FJ,NMAX,X,NMAX,FJAPRX,RF)
IMPLICIT REAL*8(A-H,O-Z)
DIMENSION FJ(1),FJAPRX(1),RF(1)
NMAX=NMAX
IF(NMAX.GT.0) GO TO 20
IF(NMAX.GT.15) GO TO 10
GO TO 20
10 TERR=4
RETURN
20 NMAX&R=IARS(NMAX)
NMAX=1
30 IF(A.GT.O) GO TO 40
IF(NMAX.LT.1.00-15) GO TO 40
TERR=1
RETURN
40 IF(A.LT.1.0) GO TO 70
TERR=2
RETURN
70 IF(A.GT.O) GO TO 130
TERR=3
RETURN
130 TERR=0
EPSION=5.00E10,**(-NO)
NMAX=NMAX+1
OUT 160 N=1,NMAX
160 FJAPRX(N)=0.0
CM=(X/X2.)*A/NMAX(1.+A)
DL=2.02600+N1.36330
IF(NMAX.LT.0) GO TO 230
Y=2.00E0/NMAX
CALL TIZ(Y,TANS)
Y=NMAX*TANS
GO TO 240
230 P=0.0
240 Y=2.35760/D1/X
CALL TIZ(Y,TANS)
S=2.35910*D1*X*TANS
IF(E.GT.5) GO TO 280
N1=1+TINT(0)
GO TO 290
280 N1=1+TINT(0)
290 M=0
FL=1.
LIMIT=(N1/2)
320 M=M+1
FL=FL*(M+1)/(MAX.00)
IF(M.LT.LIMIT) GO TO 321
M=2*M
P=P+1
S=S+1
390 DENOM=2.*A*N)/X-R
IF(AARDS(DENOM).LE.1.00-15)DENOM=DENOM+1.00-15
430 R=1./DENOM
W=-MOD(N1,2)
IF(WNE.0) GO TO 480
FL=FL*(N+2.0)/(N+2.0)
FL=FLAFL*(N+1)
GO TO 490
480 FL=FLAFL=0.0
490 S=*(FLWAFLS)
IF(N,LE,NMAX)RF(N)=0
N=N-1
TF(N,76,1)GO TO 39
FJ(J)*SUM(1,E)
IF(N4&XT,EQ,0)GO TO 570
DO 560 N=1,NM4XT
560 FJ(N+1)=FJ(N)*FJ(N)
570 DO 640 N=1,NMP1
IF(RHS((FJ(N)-FJAPXX(N))/FJ(N)),LE,FPSLOP)GO TO 640
DO 610 N=1,NMP1
610 FJAPXX(N)=FJ(M)
NUM=N+5
GO TO 230
640 CONTINUE
IF(NMAX,CF,0)RETURN
FJ(2)=2*A*(FJ(1)/X-FJ(2))
TF(NMAX&,FO,0)RETURN
DO 650 N=2,NMAX&
650 FJ(N+1)=2.*(A-N)*FJ(N)/X-FJ(N-1)
RETURN
END
610 T=2*(*(A+N)*X*Y1+Y*P2)*2+X*P+Y1*P1)**2
S1=y1**(A+N)*X*Z2*P1
FS=Fl*(N+1.0)/(N+1.0)
C=2.*(A+N)*FL
FLAMB1=C*G
FLAMR2=R*G
C=G
C1=-C
C2=G
S1=1*(FLAMR1+S1)-2*(FLAMR2+S2)
S2=1*(FLAMR2+S2)+2*(FLAMR1+S1)
S1=S
TF1(N,GT,NMAX)GO TO 770
PR1(N)=R1
R2(N)=R2
770 N=N+1
IF(N>GE.11)GO TO 610
C1=1.+S1)**2+S2**2
U11=(SUM1*(1.+S1)+SUM2*S2)/C
V11=(SUM2*(1.+S1)-SUM1*S2)/C
IF(NMAX,EQ.0)GO TO 850
ON 840 N=1,NMAX
U(N+1)=PR1(N)*U(N)-OP2(N)*V(N)
850 V(N+1)=PR1(N)*V(N)+OP2(N)*U(N)
TO 880
860 ON 870 N=1,NMAX
870 V(N)=V(N)
880 ON 950 N=1,NMAX
TEMP1=U(N)-VAPRX(N)**2
TEMP1=TEMP1+V(N)-VAPRX(N)**2
TEMP1=TEMP1+((N)**2+V(N)**2)
TEMP1=LE_EPSLO1MO TO 950
ON 920 U=1,NMAX
VAPRX(N)=U(N)
920 VAPRX(M)=V(N)
N=N+1
GO TO 460
950 CONTINUE
RETURN
END
SUBROUTINE T2(Y,T,F,S)
F=SLR(Y,Z,P,T,A,N,LOG)
F(Y,GT.10.0)GO TO 40
P=.00005741/D998Y-.001761488000
P=13001.0000
P=3901.0000
P=13001.0000
RETURN
40 Z=LOG(Y)+.77500
P=1.77500-OLGZ1/(1.0+Z)
TANS=Y/(1.+P*Z)
RETURN
END
REFERENCES


