AFCRL-71-0174

COMPUTER PROGRAMS FOR CHARACTERISTIC MODES OF WIRE OBJECTS

by

Joseph R. Mautz

Roger F. Harrington

Electrical Engineering Department
Syracuse University
Syracuse, New York 13210

Contract No. F19628-68-C-0180

Project No. 5635

Task No. 563506

Work Unit No. 56350601

SCIENTIFIC REPORT NO. 11

March 1971

Contract Monitor: John F. McIlvenna

Microwave Physics Laboratory

Prepared for

AIR FORCE CAMBRIDGE RESEARCH LABORATORIES

AIR FORCE SYSTEMS COMMAND

UNITED STATES AIR FORCE

BEDFORD, MASSACHUSETTS 01730

ABSTRACT

Computer programs for calculating the characteristic modes of wire objects of arbitrary shape are given. A program for computing the generalized impedance matrix of wire objects is included. It is valid for systems of N wires of arbitrary shape, using triangle functions for both expansion and testing. A program for using the characteristic modes in plane-wave scattering problems, showing convergence of the modal solution, is also given. Programs for making Calcomp plots of the characteristic currents, gain patterns and modal solutions are included. This report gives program descriptions, operating instructions, listings, and sample input-output data for each program.

CONTENTS

I. INTRODUCTION

This report gives computer programs and sample input-output data for the computation of characteristic modes for thin wire objects of arbitrary shape. The modes are those defined by Garbacz [1], for which the general theory is summarized in Scientific Report No. 9 (reference [2]). The notation of this report is consistent with that of [2], which should be referred to for detailed identification of the symbols used. The programs given in this report are those used to compute the numerical results for the wire arrow in [2]. The sample computations given in this report are for the bent wire defined in Fig. 1.

Five computer programs are documented here. These are defined according to their function:

#1. Calculate the generalized impedance matrix Z.

#2. Calculate the characteristic currents (eigencurrents)

#3. Calculate the gain patterns of the eigencurrents.

#4. Calculate σ/λ^2 (scattering cross section divided by the square of the wavelength) for an incident plane wave traveling in the +z and/or -z directions.

#5. Plot the eigencurrents, the gain patterns of the eigencurrents, and the scattering cross section calculated by programs #2, #3, and #4.

The next five sections of this report discuss, give operating instructions, list, and give sample input-output data for these programs.

II. GENERALIZED IMPEDANCE MATRIX

Program #1, which calculates the generalized impedance matrix Z for a thin conducting wire or wires, consists of a subroutine CALZ and a main program. The activity on data sets 1 (punched card input) and 6 (unformated direct access input-output) is described as follows:

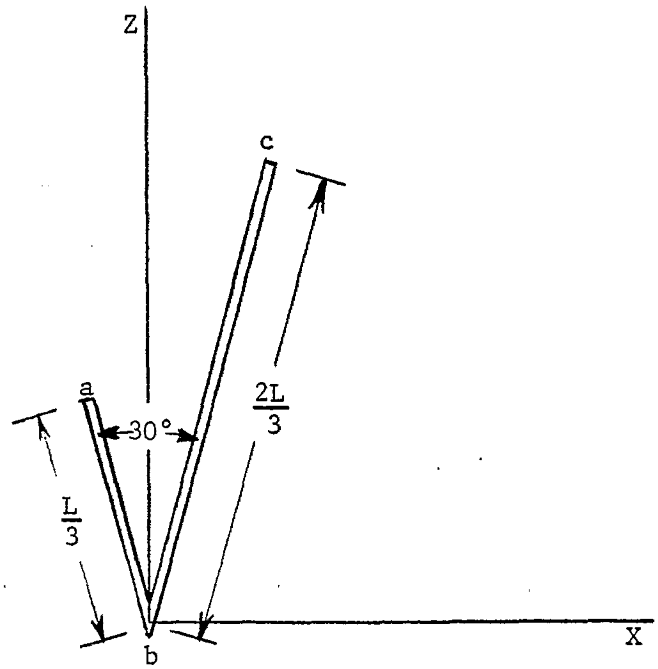

Figure 1. Bent wire object used for sample input-output data. Wire length L is 1.2 wavelengths, wire radius is 0.01 wavelength.

```
      READ (1,15) MD5, MD1
15    FORMAT (20I3)
      READ (1,15)(MD6(I), I=1, MD5)
      REWIND 6
      SKIP MD1 RECORDS ON DATA SET 6
      DO 14 K=1, MD5
23    READ (1,27) NP, NW, BK
27    FORMAT (2I3, E 14.7)
      READ (1,10)(PX(I), I=1, NP)
      READ (1,10)(PY(I), I=1, NP)
      READ (1,10) (PZ(I), I=1, NP)
10    FORMAT (10F 8.4)
      READ (1,15)(LL(I), I=1, NW)
      READ (1,34)(RAD(I), I=1, NW)
34    FORMAT (5E 14.7)
      NZ = N*N
      WRITE (6)(Z(I), I=1, NZ)
14    CONTINUE
```

Here, N is the order of the impedance matrix Z written on data set 6. MD6(K) $\neq$ 1 if BK is the only variable that changes in going from K-1 to K. Otherwise, MD6(K) = 1. MD6(1) is always 1. BK is the wave number $k = \omega\sqrt{\mu\varepsilon}$. PX, PY, and PZ are the x,y, and z coordinates of NP data points that describe the axes of NW wires. Each wire is specified by an odd number greater than or equal to 5 of data points which do not have to be equally spaced. There should be a data point at the end points of each wire. If the first wire closes on itself so that the first data point at the beginning of the wire is identical to the Jth data point at the end of the wire, a (J+1)th and a (J+2)th data point should be defined on the first wire so that the (J+1)th data point is identical to the second data point and the (J+2)th data point is identical to the third data point. The wire that closes on itself is really a junction with two branches formed by the juxtaposition of two extremities of wires. As explained previously, one of the extremities must be extended two data points (not counting the data point at the junction) through the junction. A junction with n branches is the juxtaposition of n extremities of wires. A junction with n branches may be treated by extending n-1 of the extremities two data points (not counting the data point of the junction) through the junction so that there is overlapping on n-1 of the branches. The data point at the junction will appear n times. On one branch each of the two data points nearest the junction will appear once, but on the other n-1 branches each of the two data points nearest the junction will appear twice. The LL(I)th data point starts the Ith wire. LL(1) should be 1. RAD(I) is the radius of the Ith wire.

The main program transmits the data appearing in the common statement to the subroutine CALZ. The main program also prints the impedance matrix computed by CALZ and writes this impedance matrix on data set 6.

The computation of the impedance matrix by CALZ is discussed next. The elements of the generalized impedance matrix Z for a thin wire are defined by

$$Z_{mn} = \int_C \vec{W}_m \cdot L \vec{F}_n \, d\ell \qquad (1)$$

where $\vec{F}_n$ is a current expansion function defined along the axis of the wire, $\vec{W}_m$ is a testing function defined along a contour C on the surface of the wire parallel to the axis of the wire, and -L operates on $\vec{F}_n$ to obtain the electric field produced by $\vec{F}_n$. The choice of $\vec{F}_n$ defined on the axis of the wire is equivalent to lumping the actual current density on the surface of the wire into a single filamentary current along the axis of the wire.

At present only one wire is being considered. The wire axis is defined by connecting the odd number 2N+3 of data points $\vec{r}_1, \vec{r}_2 \ldots \vec{r}_{2N+3}$ by straight line segments. Let ℓ be the length variable measured along the wire axis so that ℓ_i corresponds to the point $\vec{r}_i$. Assume that the ith straight line segment (ith subsection) of the wire has length $\Delta\ell_i$. Consider a triangular function $T_n(\ell)$ extending over four of these subsections.

$$T_n(\ell) = \begin{cases} 1 + \dfrac{\ell - \ell_{2n+1}}{\Delta\ell_{2n-1} + \Delta\ell_{2n}} & \ell_{2n-1} \leq \ell \leq \ell_{2n+1} \\ 1 - \dfrac{\ell - \ell_{2n+1}}{\Delta\ell_{2n+1} + \Delta\ell_{2n+2}} & \ell_{2n+1} \leq \ell \leq \ell_{2n+3} \end{cases} \tag{2}$$

The derivative of (2) is given by

$$\frac{dT_n(\ell)}{d\ell} = \begin{cases} \dfrac{1}{\Delta\ell_{2n-1} + \Delta\ell_{2n}} & \ell_{2n-1} \leq \ell \leq \ell_{2n+1} \\ \dfrac{-1}{\Delta\ell_{2n+1} + \Delta\ell_{2n+2}} & \ell_{2n+1} < \ell \leq \ell_{2n+3} \end{cases} \tag{3}$$

A four pulse approximation to $T_n(\ell)$ is chosen for the expansion function $\vec{F}_n$. A four impulse approximation to $T_m(\ell)$ is chosen for the testing function W_m. Thus,

$$F_n(\ell) = \frac{T(4n-3)}{\Delta\ell_{2n-1}} P_{2n-1} + \frac{T(4n-2)}{\Delta\ell_{2n}} P_{2n} + \frac{T(4n-1)}{\Delta\ell_{2n+1}} P_{2n+1} + \frac{T(4n)}{\Delta\ell_{2n+2}} P_{2n+2}$$

(4)

$$\frac{dF_n(\ell)}{d\ell} = \frac{T'(4n-3)}{\Delta\ell_{2n-1}} P_{2n-1} + \frac{T'(4n-2)}{\Delta\ell_{2n}} P_{2n} + \frac{T'(4n-1)}{\Delta\ell_{2n+1}} P_{2n+1} + \frac{T'(4n)}{\Delta\ell_{2n+2}} P_{2n+2}$$

$$W_m(\ell) = T(4m-3)\delta_{2m-1} + T(4m-2)\ \delta_{2m} + T(4m-1)\ \delta_{2m+1} + T(4m)\ \delta_{2m+2}$$

(5)

$$\frac{dW_m(\ell)}{d\ell} = T'(4m-3)\ \delta_{2m-1} + T'(4m-2)\ \delta_{2m} + T'(4m-1)\ \delta_{2m+1} + T'(4m)\ \delta_{2m+2}$$

where $P_i = P_i(\ell)$ is a pulse function

$$P_i(\ell) = \begin{cases} 1 & \ell_i \leq \ell \leq \ell_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$\delta_i = \delta(\ell - \frac{\ell_i + \ell_{i+1}}{2})$ is a Dirac delta function at $\ell = \frac{\ell_i + \ell_{i+1}}{2}$

and

$$T(4n-3) = \Delta\ell_{2n-1} \frac{\frac{1}{2}\Delta\ell_{2n-1}}{\Delta\ell_{2n-1} + \Delta\ell_{2n}}$$

$$T(4n-2) = \Delta\ell_{2n} \frac{\Delta\ell_{2n-1} + \frac{1}{2}\Delta\ell_{2n}}{\Delta\ell_{2n-1} + \Delta\ell_{2n}}$$

$$T(4n-1) = \Delta\ell_{2n+1} \frac{\frac{1}{2}\Delta\ell_{2n+1} + \Delta\ell_{2n+2}}{\Delta\ell_{2n+1} + \Delta\ell_{2n+2}} \qquad (6)$$

$$T(4n) = \Delta\ell_{2n+2} \frac{\frac{1}{2}\Delta\ell_{2n+2}}{\Delta\ell_{2n+1} + \Delta\ell_{2n+2}}$$

$$T'(4n-3) = \frac{\Delta\ell_{2n-1}}{\Delta\ell_{2n-1} + \Delta\ell_{2n}}$$

$$T'(4n-2) = \frac{\Delta\ell_{2n}}{\Delta\ell_{2n-1} + \Delta\ell_{2n}} \qquad (7)$$

$$T'(4n-1) = \frac{-\Delta\ell_{2n+1}}{\Delta\ell_{2n+1} + \Delta\ell_{2n+2}}$$

$$T'(4n) = \frac{-\Delta\ell_{2n+2}}{\Delta\ell_{2n+1} + \Delta\ell_{2n+2}}$$

Equations (4) and (5) give only the magnitudes of the vector functions $\vec{F}_n(\ell)$ and $\vec{W}_m(\ell)$ directed along the axis of the wire. Actually, $\vec{W}_m(\ell)$

should be defined not on the axis of the wire but on the contour C. Because of the impulsive nature of $\vec{W}_m$ the contour C degenerates into a series of 2N+2 field points. The exact position of the ith field point adjacent to the point $\frac{\vec{r}_i + \vec{r}_{i+1}}{2}$ on the axis of the wire will not be defined precisely. The contribution to the electric field due to the current on the subsection $\Delta\ell_i$ is evaluated at a point $\frac{\vec{r}_i + \vec{r}_{i+1}}{2} + \vec{u}_i a$ where $\vec{u}_i$ is any unit vector perpendicular to the direction $\vec{r}_{i+1} - \vec{r}_i$ of the straight line segment defining the subsection $\Delta\ell_i$, and a is the radius of the wire. The contribution to the electric field due to the current on a different subsection $\Delta\ell_j$ is evaluated at a point $\frac{\vec{r}_i + \vec{r}_{i+1}}{2} + \vec{u}_j a$, where $\vec{u}_j$ is a unit vector perpendicular to the plane of the vectors $\vec{r}_{j+1} - \vec{r}_j$, and $\frac{\vec{r}_{i+1} + \vec{r}_i}{2} - \frac{\vec{r}_{j+1} + \vec{r}_j}{2}$.

The previously used phrase "contribution to the electric field" is somewhat misleading because the integral (1) will be written as [3]

$$Z_{mn} = \int_C d\ell \int_{axis} d\ell' \left[j\omega\mu \, \vec{W}_m \cdot \vec{F}_n + \frac{1}{j\omega\varepsilon} \frac{dW_m}{d\ell} \frac{dF_n}{d\ell'} \right] \frac{e^{-jkR}}{4\pi R} \qquad (8)$$

where R is the distance from the source point to the field point. Using the expansion functions $\vec{F}_n$ given by (4) and the testing functions $\vec{W}_m$ given by (5), equation (8) becomes

$$Z_{mn} = \sum_{i=1}^{4} \sum_{j=1}^{4} [j\omega\mu \, T(4m-4+i) \, T(4n-4+j) \, D(2m-2+i, 2n-2+j)$$
$$+ \frac{1}{j\omega\varepsilon} T'(4m-4+i) \, T'(4n-4+j)] \, \psi(2m-2+i, 2n-2+j) \qquad (9)$$

where D(i,j) is the dot product between unit vectors in the directions $\vec{r}_{i+1} - \vec{r}_i$ and $\vec{r}_{j+1} - \vec{r}_j$ and

$$\psi(i,j) = \frac{1}{4\pi\Delta\ell_j} \int_{axis} d\ell' \, \frac{e^{-jkR}}{R} \qquad (10)$$

in which R is the distance between the field point adjacent to $\frac{\vec{r}_i + \vec{r}_{i+1}}{2}$ and the source point on the jth segment of length $\Delta\ell_j$. Formulas for the computation of $\psi(i,j)$ are given in the Appendix of reference [4].

The present subroutine CALZ is an attempt to reduce the execution time and storage requirements of Chao's subroutine of the same name [3]. CALZ computes the generalized impedance matrix Z_{mn} of (9) extended to multiple thin wires. The input and output for the subroutine CALZ appear in the common statement

```
COMPLEX Z(1600)
COMMON Z, KT, NP, N, LL(5), RAD(4), BK, PX(100), PY(100), PZ(100)
```

The variable KT should be 1 the first time CALZ is called. If CALZ is called again with merely a change in the propagation constant BK = $\omega\sqrt{\mu\varepsilon}$, it is more efficient that KT not be 1. The axes of the wires are defined by NP data points of which the Jth has x,y,z coordinates PX(J), PY(J), and PZ(J). The axes of the wires consist of most of the straight line segments that connect the successive data points. The variable LL indicates which straight line segments are omitted. If LL(J) = I, the straight line segment between the (I-1)th and Ith data point is omitted and the Ith data point marks the beginning of the Jth wire. LL(1) should be 1 indicating that the first data point is the beginning of the first wire. If there are NW wires, one must set LL(NW+1) > NP because for each data point on the NWth wire an inquiry is made as to whether the first data point on the (NW+1)th wire has been reached.

A closed loop of wire is treated as one wire by overlapping the last three data points with the first three such that

PX(NP-2) = PX(1)
PX(NP-1) = PX(2)
PX(NP) = PX(3)

and similarly for PY and PZ. The presence of PX(NP-1) and PX(NP) causes the first and last expansion functions to overlap on the portion of the wire between PX(1) and PX(3). Similarly an n branch junction is treated by overlapping data points on n-1 of the branches. The data point at the junction will appear n times. On each of n-1 branches the two data points nearest the junction will appear twice. There should be n-1 triangular expansion functions $\vec{F}_j$ with their peaks at the junction. On each of n-1 branches two of these $\vec{F}_j$ should oppose each other. In this way Kirchhoff's current law will be satisfied.

RAD(I) is the radius of the Ith wire. N and Z are computed by CALZ. The generalized impedance matrix of order N will be stored columnwise in the complex dimensioned variable Z. All of the previous data is in MKS units.

If Z is larger than a 40 × 40 matrix, if NP > 100 or if there are more than four wires, the common statement must be altered to allot more space to some of the dimensioned variables. Similarly the dimension statements in CALZ may have to be changed. Minimum storage allocations are as follows:

```
COMPLEX   PSI(4*N1), Z(N×N)
COMMON    LL(NW+1), RAD(NW), PX(NP), PY(NP), PZ(NP)
DIMENSION L(NW+1), XX(N1), XY(N1), XZ(N1), TX(N1)
    TY(N1), TZ(N1), AL(N1), T(4*N), TP(4×N), DC(4*N1),
    RAD2(NW)
```

where NW = number of wires

N1 = NP - NW

$$N = \frac{NP - NW}{2} - NW$$

The DO loop 8 computes XX(N1), XY(N1), XZ(N1), TX(N1), TY(N1), and TZ(N1), the x,y,z coordinates and the x,y,z components of the unit tangent vector at the midpoint of the N1th straight segment of wire of length AL(N1). DO loop 8 also squares the radius RAD(I) of the Ith wire. Upon exit from DO loop 8, N1 is the total number of straight segments of wire, N = J4-2 is the number of expansion functions $\vec{F}_n$, and L(I) indicates that $\vec{F}_{L(I)}$ is the first expansion function on the Ith wire. DO loop 5 stores T and T' of equations (6) and (7) in the dimensioned variables T and TP.

The index NS of DO loop 10 corresponds to n in equation (9). The index K of DO loop 15 corresponds to the "j" of D(2m-2+i, 2n-2+j) and ψ(2m-2+i, 2n-2+j) appearing in equation (9). If F_{NS} is not the first expansion function on a wire, KK = 3 in which case D and ψ stored in DC and PSI have already been calculated for KK = 1, 2. DO loop 14 is necessary because DC and PSI for KK = 1,2 and the present NS correspond to KK = 3, 4 at the previous value of NS. The index NF of DO loop 16 corresponds to the argument 2m-2+i of D and ψ. DO loop 16 uses Harrington's formulas [4] to compute ψ and store it in PSI. The following table relates some variables in the computer program to those of Harrington.

Computer Program	Harrington
R	r
RT	$\|z\|$
RH	ρ^2
ALP	α
AR	α/r
AI1	I_1
AI2	I_2
AI3	I_3
AI4	I_4
ZR	$\|z\|/r$
A0	A_o
A1	A_1
A2	A_2
A3	A_3
A4	A_4
PSI	ψ

Note that ψ is even z. If $|z| - \alpha \leq 0$ Harrington's equation (130) is replaced by

$$I_1 = \log\left(\frac{\left[|z| + \alpha + \sqrt{\rho^2 + (|z| + \alpha)^2}\right]\left[\alpha - |z| + \sqrt{\rho^2 + (\alpha - |z|)^2}\right]}{\rho^2}\right)$$

The index NF of DO loop 25 corresponds to the m in equation (9). The i,j sum in equation (9) is performed in nested DO loops 23 and 24, JS corresponding to j and JF corresponding to i. The coefficients of $j\omega\mu$ and $\frac{1}{j\omega\varepsilon}$ appearing in (9) are accumulated in U5 and U6 inside nested DO loops 23 and 24. The impedance matrix is stored columnwise in the dimensioned complex variable Z.

Listing of Program #1

```
//                (0034,EE,3,2),'MAUTZ,JOE',MSGLEVEL=1
// EXEC FORTGCLG,PARM.FORT='MAP'
//FORT.SYSIN DD *
      SUBROUTINE CALZ
      COMPLEX U,U3,U4,U2,U5,U6,PSI(400),Z(1600)
      COMMON Z,KT,NP,N,LL(5),RAD(4),BK,PX(100),PY(100),PZ(100)
      DIMENSION L(5),XX(100),XY(100),XZ(100),TX(100),TY(100),TZ(100)
      DIMENSION AL(100),T(200),TP(200),DC(400),RAD2(4)
      IF(KT.NE.1) GO TO 9
      U=(0.,1.)
      PI=3.141593
      ETA=376.730
      C1=.125/PI
      C2=.25/PI
      J4=2
      N1=0
      J1=1
      DO 8 J=1,NP
      IF(LL(J1)-J) 7,6,7
    6 J4=J4-1
      L(J1)=J4
      RAD2(J1)=RAD(J1)*RAD(J1)
      J1=J1+1
      GO TO 8
    7 N1=N1+1
      J3=J-1
      IF((N1/2*2-N1).EQ.0) J4=J4+1
      XX(N1)=.5*(PX(J)+PX(J3))
      XY(N1)=.5*(PY(J)+PY(J3))
      XZ(N1)=.5*(PZ(J)+PZ(J3))
      S1=PX(J)-PX(J3)
      S2=PY(J)-PY(J3)
      S3=PZ(J)-PZ(J3)
      S4=SQRT(S1*S1+S2*S2+S3*S3)
      TX(N1)=S1/S4
      TY(N1)=S2/S4
      TZ(N1)=S3/S4
      AL(N1)=S4
    8 CONTINUE
      N=J4-2
      L(J1)=J4
      J1=1
      J2=-2
      DO 5 J=1,N
      IF(L(J1)-J) 3,4,3
    4 J2=J2+2
      J1=J1+1
    3 J3=(J-1)*4
      J4=J3+1
      J5=J4+1
      J6=J5+1
      J7=J6+1
      K4=J2+1
      K5=K4+1
      K6=K5+1
      K7=K6+1
      S1=AL(K4)+AL(K5)
      S2=AL(K6)+AL(K7)
      T(J4)=AL(K4)*.5*AL(K4)/S1
      T(J5)=AL(K5)*(AL(K4)+.5*AL(K5))/S1
```

```
      T(J6)=AL(K6)*(AL(K7)+.5*AL(K6))/S2
      T(J7)=AL(K7)*.5*AL(K7)/S2
      TP(J4)=AL(K4)/S1
      TP(J5)=AL(K5)/S1
      TP(J6)=-AL(K6)/S2
      TP(J7)=-AL(K7)/S2
      J2=J2+2
    5 CONTINUE
    9 U3=U*BK*ETA
      U4=-U/BK*ETA
      BK2=BK*BK/2.
      BK3=BK2*BK/3.
      N9=0
      N2=1
      N3=-2
      DO 10 NS=1,N
      IF(L(N2)-NS) 12,11,12
   11 KK=1
      N3=N3+2
      N2=N2+1
      GO TO 13
   12 KK=3
      DO 14 NF=1,N1
      N4=NF+N1
      N5=N4+N1
      N6=N5+N1
      DC(NF)=DC(N5)
      DC(N4)=DC(N6)
      PSI(NF)=PSI(N5)
      PSI(N4)=PSI(N6)
   14 CONTINUE
   13 N4=N2-1
      DO 15 K=KK,4
      N7=N3+K
      K1=(K-1)*N1
      DO 16 NF=1,N1
      N8=NF+K1
      S1=XX(N7)-XX(NF)
      S2=XY(N7)-XY(NF)
      S3=XZ(N7)-XZ(NF)
      R2=S1*S1+S2*S2+S3*S3+RAD2(N4)
      R=SQRT(R2)
      RT=ABS(S1*TX(N7)+S2*TY(N7)+S3*TZ(N7))
      RT2=RT*RT
      RH=(R2-RT2)
      ALP=.5*AL(N7)
      AR=ALP/R
      S1=BK*R
      U2=COS(S1)-U*SIN(S1)
      IF(AR-.1) 22,22,21
   21 U2=U2*C1/ALP
      S1=RT-ALP
      S2=RT+ALP
      S3=SQRT(S1*S1+RH)
      S4=SQRT(S2*S2+RH)
      IF(S1) 18,18,19
   18 AI1=ALOG((S2+S4)*(-S1+S3)/RH)
      GO TO 20
   19 AI1=ALOG((S2+S4)/(S1+S3))
   20 AI2=AL(N7)
```

```
      AI3=(S2*S4-S1*S3+RH*AI1)/2.
      AI4=AI2*(RH+ALP*ALP/3.+RT2)
      S3=AI1*R
      S1=AI1-BK2*(AI3-R*(2.*AI2-S3))
      S2=-BK*(AI2-S3)+BK3*(AI4-3.*AI3*R+R2*(3.*AI2-S3))
      GO TO 28
   22 U2=U2*C2/R
      BA=BK*ALP
      BA2=BA*BA
      AR2=AR*AR
      AR3=AR2*AR
      ZR=RT/R
      ZR2=ZR*ZR
      ZR3=ZR2*ZR
      ZR4=ZR3*ZR
      H1=(3.-30.*ZR2+35.*ZR4)*AR3/40.
      A1=AR*(-1.+3.*ZR2)/6.
      A0=1.+AR*(A1+H1)
      A2=-ZR2/6.-AR2*(1.-12.*ZR2+15.*ZR4)/40.
      A3=AR*(3.*ZR2-5.*ZR4)/60.
      A4=ZR4/120.
      S1=A0+BA2*(A2+BA2*A4)
      S2=BA*(A1+BA2*A3)
   28 PSI(N8)=U2*(S1+U*S2)
      DC(N8)=TX(NF)*TX(N7)+TY(NF)*TY(N7)+TZ(NF)*TZ(N7)
   16 CONTINUE
   15 CONTINUE
      N3=N3+2
      J3=(NS-1)*4
      J7=-2
      J9=1
      DO 25 NF=1,N
      J1=(NF-1)*4
      IF(L(J9)-NF) 26,27,26
   27 J9=J9+1
      J7=J7+2
   26 N9=N9+1
      U5=0.
      U6=0.
      J5=0
      DO 23 JS=1,4
      J4=J3+JS
      J8=J5+J7
      DO 24 JF=1,4
      J6=J8+JF
      J2=J1+JF
      U5=T(J2)*T(J4)*DC(J6)*PSI(J6)+U5
      U6=TP(J2)*TP(J4)*PSI(J6)+U6
   24 CONTINUE
      J5=J5+N1
   23 CONTINUE
      Z(N9)=U5*U3+U6*U4
      J7=J7+2
   25 CONTINUE
   10 CONTINUE
      RETURN
      END
      COMPLEX Z(1600)
      COMMON Z,KT,NP,N,LL(5),RAD(4),BK,PX(100),PY(100),PZ(100)
      DIMENSION MD6(30)
```

```
      REWIND 6
      READ(1,15) MD5,MD1
   15 FORMAT(20I3)
      WRITE(3,16) MD5,MD1
   16 FORMAT('1MD5 MD1'/1X,2I4)
      READ(1,15)(MD6(I),I=1,MD5)
      WRITE(3,20)(MD6(I),I=1,MD5)
   20 FORMAT('0MD6'/(1X,20I3))
      IF(MD1) 23,23,24
   24 DO 26 J=1,MD1
      READ(6)
   26 CONTINUE
   23 DO 14 K=1,MD5
      READ(1,27) NP,NW,BK
   27 FORMAT(2I3,E14.7)
      WRITE(3,28) NP,NW,BK
   28 FORMAT('0 NP NW       BK'/1X,2I3,E14.7)
      KT=MD6(K)
      READ(1,10)(PX(I),I=1,NP)
      READ(1,10)(PY(I),I=1,NP)
      READ(1,10)(PZ(I),I=1,NP)
   10 FORMAT(10F8.4)
      WRITE(3,29)(PX(I),I=1,NP)
      WRITE(3,30)(PY(I),I=1,NP)
      WRITE(3,31)(PZ(I),I=1,NP)
   29 FORMAT('0PX'/(1X,10F8.4))
   30 FORMAT('0PY'/(1X,10F8.4))
   31 FORMAT('0PZ'/(1X,10F8.4))
      READ(1,15)(LL(I),I=1,NW)
      WRITE(3,33)(LL(I),I=1,NW)
   33 FORMAT('0LL'/(1X,10I3))
      LL(NW+1)=200
      READ(1,34)(RAD(I),I=1,NW)
   34 FORMAT(5E14.7)
      WRITE(3,35)(RAD(I),I=1,NW)
   35 FORMAT('0RAD'/(1X,5E14.7))
      CALL CALZ
      NZ=N*N
      WRITE(6)(Z(I),I=1,NZ)
      WRITE(3,38) N
   38 FORMAT('0IMPEDANCE MATRIX OF ORDER',I3)
      DO 36 J=1,2
      J1=(J-1)*N+1
      J2=J1+N-1
      WRITE(3,37)(Z(I),I=J1,J2)
   37 FORMAT(1X,10E11.4)
   36 CONTINUE
   14 CONTINUE
      STOP
      END
/*
//GO.FT06F001 DD DSNAME=SURC0677.ZNEW,DISP=OLD,UNIT=2314,               X
//               VOLUME=SER=SU0005,DCB=(RECFM=V,BLKSIZE=2596,LRECL=2592)
//GO.SYSIN DD *
  1 59
  1
 55  1 0.1396263E+00
 -4.6587 -4.3999 -4.1411 -3.8823 -3.6235 -3.3646 -3.1058 -2.8470 -2.5882 -2.3294
 -2.0706 -1.8117 -1.5529 -1.2941 -1.0353 -0.7765 -0.5176 -0.2588 -0.0000  0.2588
  0.5176  0.7765  1.0353  1.2941  1.5529  1.8117  2.0706  2.3294  2.5882  2.8470
```

```
 3.1058   3.3646   3.6235   3.8823   4.1411   4.3999   4.6587   4.9176   5.1764   5.4352
 5.6940   5.9528   6.2117   6.4705   6.7293   6.9881   7.2469   7.5058   7.7646   8.0234
 8.2822   8.5410   8.7998   9.0587   9.3175
 0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
 0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
 0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
 0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
 0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000   0.0000
 0.0000   0.0000   0.0000   0.0000   0.0000
17.3867  16.4207  15.4548  14.4889  13.5230  12.5570  11.5911  10.6252   9.6593   8.6933
 7.7274   6.7615   5.7956   4.8296   3.8637   2.8978   1.9319   0.9659   0.0000   0.9659
 1.9319   2.8978   3.8637   4.8296   5.7956   6.7615   7.7274   8.6933   9.6593  10.6252
11.5911  12.5570  13.5230  14.4889  15.4548  16.4207  17.3867  18.3526  19.3185  20.2844
21.2504  22.2163  23.1822  24.1481  25.1141  26.0800  27.0459  28.0118  28.9778  29.9437
30.9096  31.8755  32.8415  33.8074  34.7733
  1
 0.4500000E+00
/*
//
```

Sample Output

```
MD5 MD1
  1  59

MD6
  1

 NP NW       BK
 55  1 0.1396263E 00

PX
 -4.6587 -4.3999 -4.1411 -3.8823 -3.6235 -3.3646 -3.1058 -2.8470 -2.5882 -2.3294
 -2.0706 -1.8117 -1.5529 -1.2941 -1.0353 -0.7765 -0.5176 -0.2588  0.0     0.2588
  0.5176  0.7765  1.0353  1.2941  1.5529  1.8117  2.0706  2.3294  2.5882  2.8470
  3.1058  3.3646  3.6235  3.8823  4.1411  4.3999  4.6587  4.9176  5.1764  5.4352
  5.6940  5.9528  6.2117  6.4705  6.7293  6.9881  7.2469  7.5058  7.7646  8.0234
  8.2822  8.5410  8.7998  9.0587  9.3175

PY
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0

PZ
 17.3867 16.4207 15.4548 14.4889 13.5230 12.5570 11.5911 10.6252  9.6593  8.6933
  7.7274  6.7615  5.7956  4.8296  3.8637  2.8978  1.9319  0.9659  0.0     0.9659
  1.9319  2.8978  3.8637  4.8296  5.7956  6.7615  7.7274  8.6933  9.6593 10.6252
 11.5911 12.5570 13.5230 14.4889 15.4548 16.4207 17.3867 18.3526 19.3185 20.2844
 21.2504 22.2163 23.1822 24.1481 25.1141 26.0800 27.0459 28.0118 28.9778 29.9437
 30.9096 31.8755 32.8415 33.8074 34.7733

LL
  1

RAD
  0.4500000E 00

IMPEDANCE MATRIX OF ORDER 26
  0.1551E 01-0.3581E 03 0.1539E 01 0.1283E 03 0.1503E 01 0.5266E 02 0.1445E 01 0.1232E 02 0.1367E 01 0.5298E 01
  0.1271E 01 0.2816E 01 0.1159E 01 0.1521E 01 0.1036E 01 0.9366E 00 0.7199E-01 0.5876E-01-0.8622E 00-0.6620E 00
 -0.9288E 00-0.9118E 00-0.9817E 00-0.1100E 01-0.1019E 01-0.1066E 01-0.1043E 01-0.5953E 00-0.1043E 01 0.3894E 00
 -0.1028E 01 0.1297E 01-0.9956E 00 0.1439E 01-0.9477E 00 0.1040E 01-0.8551E 00 0.6190E 00-0.3101E 00 0.3092E 00
 -0.7253E 00 0.1839E 00-0.6334E 00 0.1613E 00-0.5373E 00 0.1[illegible]E 00-0.4399E 00 0.2193E 00-0.3443E 00 0.2495E 00
 -0.2530E 00 0.2682E 00
  0.1539E 01 0.1283E 03 0.1551E 01-0.3581E 03 0.1539E 01 0.1283E 03 0.1503E 01 0.5266E 02 0.1445E 01 0.1232E 02
  0.1367E 01 0.5298E 01 0.1271E 01 0.2516E 01 0.1159E 01 0.1521E 01 0.7935E-01 0.1017E 00-0.9704E 00-0.1155E 01
 -0.1030E 01-0.1522E 01-0.1074E 01-0.1711E 01-0.1101E 01-0.1284E 01-0.1110E 01 0.1020E 00-0.1100E 01 0.1649E 01
 -0.1073E 01 0.1994E 01-0.1028E 01 0.1288E 01-0.9672E 00 0.5266E 00-0.4931E 00 0.1189E 00-0.8077E 00-0.7779E-02
 -0.7140E 00 0.3958E-02-0.6150E 00 0.6514E-01-0.5135E 00 0.1332E 00-0.4127E 00 0.1906E 00-0.3154E 00 0.2308E 00
 -0.2242E 00 0.2528E 00
```

III. EIGENCURRENTS

The method of computation of the eigencurrents is described on pages 26-28 of reference [2]. The activity on data sets 1 (punched card input) and 2 (unformated direct access input-output) is described as follows:

```
      READ (1,7) MD5
    7 FORMAT(10I3)
      READ(1,7)(MD1(I), I=1, MD5)
      READ(1,7)(MD2(I), I=1, MD5)
      REWIND 6
      DO 143 KAP = 1, MD5
      READ(1,26) N, EPS, EPL
   26 FORMAT(I3, 2E 11.4)
      READ(1,27)(ZL(I), I = 1, N)
   27 FORMAT(7E 11.4)
      SKIP MD1(KAP) RECORDS ON DATA SET 6
      NZ = N*N
      READ(6)(Z(I), I = 1, NZ)
      SKIP MD2(KAP) RECORDS ON DATA SET 6
      J3 = N*NE
      WRITE(6)(FI(I), I = 1, J3)
  143 CONTINUE
```

Here, Z is the impedance matrix of order N calculated by program #1. The set of NE eigencurrents is stored columnwise in FI. Note that the variable NE appears as JM in the present program #2. The name NE is introduced to be compatible with subsequent programs #3, #4, and #5. R and X appearing in (2-25) of [2] are respectively the real and imaginary parts of the impedance matrix Z.

$$Z = R + jX \tag{11}$$

All eigenvalues of R that are less than EPS times the largest eigenvalue of R are set equal to zero. A lumped complex impedance load ZL(I) placed at the peak of the Ith triangular expansion function is treated by replacing the Ith diagonal element Z_{II} of Z by Z_{II} + ZL(I). If the absolute value of ZL(I) is greater than EPL, the coefficient of the Ith triangular function in the expansion of the current is set equal to zero such that ZL(I) is eliminated from all the computations. The variable EPL is introduced because the method of computation described in reference [2] does not give accurate eigencurrents when the magnitude of one of the impedance loads is several orders of magnitude larger than that of any of the diagonal elements of the impedance matrix.

If either of the variables MD5 or N in the previously read punched card data is greater than 30, more space must be assigned to the dimensioned variables. Minimum allocations are given by

```
COMPLEX   Z(N*N), ZL(N)
DIMENSION U(N*N), R(N*N), T2(N*N), A22(N*N)
B(N*N), X(N*N), A(N*N), Y(N*N), T3(N*N),
FI(N*N), EU(N), RU(N), AMD(N), LB(N), MB(N),
MD1(MD5), MD2(MD5), RL(N)
```

DO loop 30 reduces N to the number of non-zero coefficients in the triangular function expansion of the current. RL(J) $\leq$ 0 indicates that the Jth coefficient is zero. DO loop 11 stores the real and imaginary parts of the impedance matrix Z in R and X respectively. The matrices R and X are made symmetric by averaging corresponding off diagonal elements. The impedance matrix Z computed by program #1 would be symmetric if the testing functions were exactly equal to the expansion functions. As further justification for making Z symmetric, exploratory computations indicate that σ/λ^2 does not change appreciably when Z is made symmetric by taking the average of corresponding off diagonal terms or by setting the elements of Z below the main diagonal equal to those above the main diagonal or by setting the elements of Z above the main diagonal equal to those below the main diagonal. If RL(J) $\leq$ 0 indicating that the Jth coefficient in the triangular function expansion of the current is zero, DO loop 11 eliminates both the Jth row and the Jth column of the matrices R and X. DO loop 11 also adds the real and imaginary parts of ZL(J) to the Jth diagonal elements of the matrices R and X respectively. The matrix X is stored columnwise but the matrix R is stored according to the symmetric mode of storage dictated by the subroutine EIGEN in the scientific subroutine package [5]. After execution of statement 130, the new diagonal elements of R are the eigenvalues of the original matrix R and U is the orthogonal eigenvector matrix of the original matrix R. The columns of U are the orthonormal eigenvectors of the original matrix R arranged in the order of descending eigenvalues.

Referring to (2-27) of [2], DO loop 75 puts the matrix [XU] in T2. DO loop 78 puts the matrix $[\tilde{U} \times U]$ in A.

JN eigenvalues of the original matrix R are set equal to zero. If JN = 0, the matrices $[A_{12}]$ and $[A_{22}]$ of [2] disappear. Equation (2-36) of [2] then becomes

$$[\mu_{11}^{-1/2} A_{11} \mu_{11}^{-1/2}][y] = \lambda[y] \tag{12}$$

and the expression

$$[T2] = \begin{bmatrix} [\delta] \\ \\ [-A_{22}^{-1} \tilde{A}_{12}] \end{bmatrix} [\mu_{11}^{-1/2} y] \tag{13}$$

reduces to $[\mu_{11}^{-1/2} y]$. The logic between statements 146 and 151 stores $[\mu_{11}^{-1/2} y]$ in T2. Note that $[\mu_{11}^{-1/2} y]$ is a square matrix because [y] is a square matrix whose columns are the eigenvectors of (12).

The logic between statements 145 and 147 obtains the expression [T2] of (13) when JN $\neq$ 0. In particular, DO loop 73 stores the matrix A_{22} of [2] in A22. Statement 128 calls the matrix inversion subroutine MINV from the scientific subroutine package [5]. DO loop 81 puts $A_{22}^{-1} \tilde{A}_{12}$ in T3. DO loop 84 obtains the matrix [B] of equation (2-36) of [2]. Statement 129 obtains the eigenvector matrix [y]. DO loop 91 stores $[\mu_{11}^{-1/2} y]$ in the first JM rows of T2. DO loop 93 stores

$$[-A_{22}^{-1} \tilde{A}_{12}][\mu_{11}^{-1/2} y]$$

in the last JN rows of T2.

With [T2] of (13), equation (2-37) of [2] becomes

$$[I] = [U][T2] \tag{14}$$

The columns of the matrix [I] of (14) are the eigencurrents or rather the coefficients in the triangular function expansion of the eigencurrents. The index J of DO loop 96 indicates the Jth column of the matrix [I]. DO loop 97 stores [I] of (14) in FI. DO loop 137 normalizes the largest element in the Jth column of [I] to unity.

Listing of Program #2

```
//              (0034,EE,4,2),'MAUTZ,JOE',MSGLEVEL=1
// EXEC SSPCLG,PARM.FORT='MAP'
//FORT.SYSIN DD *
      COMPLEX Z(900),U1,ZL(30)
      DIMENSION U(900),R(900),T2(900),A22(900),B(900),X(900),A(900)
      DIMENSION Y(900),T3(900),FI(900),EU(30),RU(30),AMD(30),LB(30)
      DIMENSION MB(30),MD1(30),MD2(30),RL(30)
      EQUIVALENCE (R(1),T2(1),A22(1),B(1)),(X(1),A(1),Y(1))
      EQUIVALENCE (T3(1),FI(1)),(EU(1),AMD(1))
      REWIND 6
      READ(1,7) MD5
      WRITE(3,13) MD5
   13 FORMAT('1MD5'/1X,I3)
      READ(1,7)(MD1(I),I=1,MD5)
      READ(1,7)(MD2(I),I=1,MD5)
    7 FORMAT(10I3)
      WRITE(3,14)(MD1(I),I=1,MD5)
      WRITE(3,15)(MD2(I),I=1,MD5)
   14 FORMAT('0MD1'/(1X,10I3))
   15 FORMAT('0MD2'/(1X,10I3))
      DO 143 KAP=1,MD5
      READ(1,26) N,EPS,EPL
   26 FORMAT(I3,2E11.4)
      WRITE(3,3) N,EPS,EPL
    3 FORMAT('0  N     EPS',8X,'EPL'/1X,I3,2E11.4)
      READ(1,27)(ZL(I),I=1,N)
   27 FORMAT(7E11.4)
      WRITE(3,28)(ZL(I),I=1,N)
   28 FORMAT('0ZL'/(1X,7E11.4))
      NZ=N*N
      J1=IABS(MD1(KAP))
      IF(MD1(KAP)) 16,17,18
   16 DO 19 J=1,J1
      BACKSPACE 6
   19 CONTINUE
      GO TO 17
   18 DO 20 J=1,J1
      READ(6)
   20 CONTINUE
   17 READ(6)(Z(I),I=1,NZ)
      J1=IABS(MD2(KAP))
      IF(MD2(KAP)) 21,22,23
   21 DO 24 J=1,J1
      BACKSPACE 6
   24 CONTINUE
      GO TO 22
   23 DO 25 J=1,J1
      READ(6)
   25 CONTINUE
   22 NN=N
      DO 30 J=1,NN
      RL(J)=EPL-CABS(ZL(J))
      IF(RL(J)) 31,31,30
   31 N=N-1
   30 CONTINUE
      J5=0
      J6=0
      DO 11 J=1,NN
      IF(RL(J)) 11,11,32
   32 J2=(J-1)*NN
```

```
      J6=J6+1
      K6=(J6-1)*N
      J7=0
      DO 12 I=1,J
      IF(RL(I)) 12,12,33
   33 J3=J2+I
      J5=J5+1
      J7=J7+1
      J4=(I-1)*NN+J
      U1=.5*(Z(J3)+Z(J4))
      R(J5)=U1
      J8=K6+J7
      J9=J6+(J7-1)*N
      X(J8)=AIMAG(U1)
      X(J9)=X(J8)
   12 CONTINUE
      R(J5)=R(J5)+REAL(ZL(J))
      X(J8)=X(J8)+AIMAG(ZL(J))
   11 CONTINUE
  130 CALL EIGEN(R,U,N,0)
      J1=0
      DO 104 J=1,N
      J1=J1+J
      EU(J)=R(J1)
      RU(J)=1./SQRT(ABS(EU(J)))
  104 CONTINUE
      WRITE(3,141)(EU(J),J=1,N)
  141 FORMAT('0EIGENVALUES OF THE MATRIX R'/(1X,7E11.4))
      DO 75 J=1,N
      J1=(J-1)*N
      DO 76 I=1,N
      J2=J1+I
      T2(J2)=0.
      J3=(I-1)*N
      DO 77 K=1,N
      K1=K+J3
      K2=K+J1
      T2(J2)=T2(J2)+X(K1)*U(K2)
   77 CONTINUE
   76 CONTINUE
   75 CONTINUE
      DO 78 J=1,N
      J1=(J-1)*N
      DO 79 I=1,J
      J2=J1+I
      A(J2)=0.
      J3=(I-1)*N
      DO 80 K=1,N
      K1=K+J3
      K2=K+J1
      A(J2)=A(J2)+U(K1)*T2(K2)
   80 CONTINUE
      J4=J3+J
      A(J4)=A(J2)
   79 CONTINUE
   78 CONTINUE
      X2=EU(1)*EPS
      DO 70 J=1,N
      IF(EU(J)-X2) 72,144,144
  144 JM=J
```

```
   70 CONTINUE
   72 JN=N-JM
      JM1=JM+1
      IF(JN) 145,146,145
  146 J2=0
      DO 148 J=1,N
      J3=(J-1)*N
      DO 149 I=1,J
      J2=J2+1
      J4=J3+I
      B(J2)=A(J4)*RU(J)*RU(I)
  149 CONTINUE
  148 CONTINUE
      CALL EIGEN(B,Y,JM,0)
      J1=0
      DO 150 J=1,N
      J1=J1+J
      AMD(J)=B(J1)
  150 CONTINUE
      WRITE(3,58)(AMD(J),J=1,N)
      DO 151 J=1,N
      J1=(J-1)*N
      DO 152 I=1,N
      J2=I+J1
      T2(J2)=Y(J2)*RU(I)
  152 CONTINUE
  151 CONTINUE
      GO TO 147
  145 J1=0
      DO 73 J=JM1,N
      J2=(J-1)*N
      DO 74 I=JM1,N
      J1=J1+1
      J3=J2+I
      A22(J1)=A(J3)
   74 CONTINUE
   73 CONTINUE
  128 CALL MINV(A22,JN,D,LB,MB)
      J1=0
      DO 81 J=1,JM
      J3=(J-1)*N+JM
      DO 82 I=1,JN
      J2=(I-1)*JN
      J1=J1+1
      T3(J1)=0.
      DO 83 K=1,JN
      K1=J2+K
      K2=J3+K
      T3(J1)=T3(J1)+A22(K1)*A(K2)
   83 CONTINUE
   82 CONTINUE
   81 CONTINUE
      J2=0
      DO 84 J=1,JM
      J3=(J-1)*N
      J5=(J-1)*JN
      DO 85 I=1,J
      J2=J2+1
      J4=J3+I
      B(J2)=A(J4)
```

```
      J6=(I-1)*N+JM
      DO 86 K=1,JN
      K1=K+J6
      K2=K+J5
      B(J2)=B(J2)-A(K1)*T3(K2)
   86 CONTINUE
      B(J2)=B(J2)*RU(J)*RU(I)
   85 CONTINUE
   84 CONTINUE
  129 CALL EIGEN(B,Y,JM,0)
      J1=0
      DO 107 J=1,JM
      J1=J1+J
      AMD(J)=B(J1)
  107 CONTINUE
      WRITE(3,58)(AMD(J),J=1,JM)
   58 FORMAT('OEIGENVALUES OF THE MATRIX B'/(1X,5E14.7))
      DO 91 J=1,JM
      J1=(J-1)*JM
      J4=(J-1)*N
      DO 92 I=1,JM
      J3=I+J4
      J2=I+J1
      T2(J3)=Y(J2)*RU(I)
   92 CONTINUE
   91 CONTINUE
      S1=0.
      DO 93 J=1,JM
      J1=(J-1)*N
      DO 94 I=1,JN
      J2=J1+I+JM
      T2(J2)=0.
      DO 95 K=1,JM
      K1=(K-1)*JN+I
      K2=K+J1
      T2(J2)=T2(J2)-T3(K1)*T2(K2)
   95 CONTINUE
   94 CONTINUE
   93 CONTINUE
  147 DO 96 J=1,JM
      S1=0.
      J1=(J-1)*N
      J6=(J-1)*NN
      J7=0
      DO 97 I=1,NN
      J2=J6+I
      FI(J2)=0.
      IF(RL(I)) 97,97,34
   34 J7=J7+1
      DO 98 K=1,N
      K2=K+J1
      K1=(K-1)*N+J7
      FI(J2)=FI(J2)+U(K1)*T2(K2)
   98 CONTINUE
      S2=ABS(FI(J2))
      IF(S2-S1) 97,133,133
  133 S1=S2
      J5=J2
   97 CONTINUE
      S1=1./FI(J5)
```

```
      J2=J6+1
      J3=J6+NN
      WRITE(3,138) AMD(J)
  138 FORMAT('OEIGENCURRENT FOR WHICH LAMBDA = ',E11.4)
      DO 137 I=J2,J3
      FI(I)=FI(I)*S1
  137 CONTINUE
      WRITE(3,60)(FI(I),I=J2,J3)
   60 FORMAT(1X,10F8.4)
   96 CONTINUE
      WRITE(6)(FI(I),I=1,J3)
  143 CONTINUE
      STOP
      END
/*
//GO.FT06F001 DD DSNAME=SURC0677.ZNEW,DISP=OLD,UNIT=2314,                X
//                VOLUME=SER=SU0005,DCB=(RECFM=V,BLKSIZE=2596,LRECL=2592)
//GO.SYSIN DD *
  1
 59
  0
 26 0.1000E-03 0.1000E+11
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00
/*
//
```

```
M05
  1

M01
 59

M02
  0

  N    EPS          EPL
 26 0.1000E-03 0.1000E 11

ZL
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0           0.0           0.0           0.0           0.0
 0.0           0.0           0.0

EIGENVALUES OF THE MATRIX R
 0.2592E 02 0.9174E 01 0.2379E 01 0.1058E 01 0.3252E 00 0.3108E-01 0.1541E-01
 0.8349E-03 0.6355E-03 0.5730E-03 0.5524E-03 0.5102E-03 0.4359E-03 0.3986E-03
 0.3302E-03 0.2956E-03 0.2720E-03 0.1152E-03 0.8505E-05-0.4609E-04-0.3562E-03
-0.3795E-03-0.4551E-03-0.4809E-03-0.6641E-03-0.6871E-03

EIGENVALUES OF THE MATRIX B
 0.8315814E 01 0.7697922E 00-0.2573003E 01-0.1351125E 03-0.1706173E 03
-0.5537598E 04-0.1329068E 05

EIGENCURRENT FOR WHICH LAMBDA =  0.8316E 01
  0.2394   0.4109   0.5796   0.7288   0.8511   0.9395   0.9894   1.0000   0.9703   0.9619
  0.9657   0.9185   0.8426   0.7426   0.6245   0.4959   0.3646   0.2381   0.1230   0.0249
 -0.0522  -0.1054  -0.1337  -0.1374  -0.1178  -0.0810

EIGENCURRENT FOR WHICH LAMBDA =  0.7698E 00
 -0.2398  -0.4070  -0.5688  -0.7082  -0.8188  -0.8943  -0.9306  -0.9301  -0.8951  -0.8055
 -0.6919  -0.5308  -0.3334  -0.1105   0.1240   0.3550   0.5677   0.7484   0.8858   0.9714
  1.0000   0.9699   0.8829   0.7440   0.5580   0.3465

EIGENCURRENT FOR WHICH LAMBDA = -0.2573E 01
  0.2799   0.3972   0.4596   0.4534   0.3823   0.2535   0.0727  -0.1305  -0.4198  -0.6738
 -0.8165  -0.8984  -0.8966  -0.8134  -0.6553  -0.4353  -0.1718   0.1128   0.3931   0.6438
  0.8409   0.9646   1.0000   0.9386   0.7727   0.5261

EIGENCURRENT FOR WHICH LAMBDA = -0.1351E 03
 -0.2927  -0.2597  -0.1390   0.0287   0.1781   0.2517   0.1912  -0.0088  -0.5657  -0.9392
 -0.9485  -0.7757  -0.4437  -0.0451   0.3620   0.7050   0.9293   1.0000   0.9053   0.6581
  0.2947  -0.1286  -0.5399  -0.8585  -0.9867  -0.8857

EIGENCURRENT FOR WHICH LAMBDA = -0.1706E 03
 -0.5737  -0.6253  -0.5126  -0.2604   0.0639   0.4008   0.6919   0.8958   1.0000   0.6664
  0.4370   0.2173   0.0306  -0.1159  -0.2170  -0.2723  -0.2853  -0.2623  -0.2106  -0.1383
 -0.0549   0.0290   0.1021   0.1520   0.1684   0.1454

EIGENCURRENT FOR WHICH LAMBDA = -0.5538E 04
  0.1132   0.1074   0.0328  -0.0899  -0.2118  -0.2738  -0.1831   0.1022   1.0000   0.9505
  0.4831  -0.0769  -0.5809  -0.8907  -0.9533  -0.7694  -0.3954   0.0738   0.5241   0.8458
  0.9509   0.7971   0.4054  -0.1494  -0.6697  -0.9848

EIGENCURRENT FOR WHICH LAMBDA = -0.1329E 05
  0.8467   0.3433  -0.2555  -0.7056  -0.7923  -0.4915   0.0710   0.6957   1.0000   0.0568
 -0.2123  -0.2856  -0.2108  -0.0884   0.0299   0.1046   0.1251   0.1039   0.0565   0.0007
 -0.0456  -0.0679  -0.0598  -0.0265   0.0172   0.0565
```

IV. GAIN EIGENPATTERNS

Six different gain patterns of the eigencurrents may be computed. The [(J-1)*2+K]th pattern of an eigencurrent is specified by letting J=1,2,3 denote the x=0, y=0, z=0 planes respectively and by letting K=1,2 denote $\vec{u}_\theta$, $\vec{u}_\phi$ polarizations respectively. The activity on data sets 1 (punched card input) and 6 (direct access input-output) is described as follows:

```
        READ(1,154) MD5
        READ(1,154)(MD1(I),I = 1, MD5)
        READ(1,154)(MD2(I),I = 1, MD5)
  154   FORMAT(20I3)
        REWIND 6
        DO 143 KAP = 1, MD5
        READ(1,35) NP, NT, NS, NW, LG, BK
   35   FORMAT(5I3, E14.7)
        READ(1,10)(PX(I),I = 1, NP)
        READ(1,10(PY(I), I = 1, NP)
        READ(1,10)(PZ(I),I = 1, NP)
   10   FORMAT(10F8.4)
        READ(1,154)(LL(I),I = 1, NW)
        READ(1,33)(RAD(I),I = 1, NW)
   33   FORMAT(5E14.7)
        SKIP MD1(KAP) RECORDS ON DATA SET 6
        NZ = N*N
        READ(6)(Z(I),I = 1, NZ)
        MD6 = MD2(KAP)
        DO 158 ILD = 1, MD6
        READ(1,154) NE, MD3, MD4
        READ(1,33)(AMD(I),I = 1, NE)
        J1 = NE*6
        READ(1,154)(LC(I), I = 1, J1)
        READ(1,100)(ZL(I), I = 1,N)
  100   FORMAT(7E11.4)
```

```
          SKIP MD3 RECORDS ON DATA SET 6
          NZ1 = N*NE
          READ(6)(FI(I),I = 1, NZ1)
          SKIP MD4 RECORDS ON DATA SET 6
          WRITE(6)(G(J),J = 1, J2)
158       CONTINUE
143       CONTINUE
```

Here, Z is the impedance matrix of order N computed by program #1 and FI is the set of NE eigencurrents computed by program #2. The present program #3 stores the gain patterns in G.

For the KAPth Z, PX, PY, and PZ are the x,y,z coordinates of the NP data points that describe NW wires. The gain will be computed at NT equally spaced angles (measured from z=0 for the patterns in the planes x=0, y=0 and from x=0 for the patterns in the plane z=0) between 0° and 360°/LG. LG should be 2 in order to be compatible with the subsequent plot program #5. Only the results at the first, (NS+1)th, (2*NS+1)th ... angles will be printed. BK is the propagation constant. The LL(I)th data point marks the beginning of the Ith wire. RAD(I) is the radius of the Ith wire. AMD appears under the heading "Eigenvalues of the Matrix B" in the printed output of program #2. If LC((J-1)*NE + K) = 0, the Jth pattern of the Kth eigencurrent is not computed. Otherwise, this pattern is computed. If LC dictates that LP patterns are computed, then J2 = NT*LP is the number of words stored in G. The impedance load to be added to the Ith diagonal element of Z is read in through the complex variable ZL(I).

Minimum allocations are given by

```
COMPLEX   Z(N*N), VR(NT*N*6), E(NT), ZL(N)
DIMENSION LL(NW+1), RAD(NW), PX(NP), PY(NP),
          PZ(NP), L(NW+1), BX(NP-NW), BY(NP-NW), BZ(NP-NW),
          AL(NP-NW), T(N*4), FI(N*NE), X(N*N),
          P(NE), PP(NE), SN(NT), CS(NT), AMD(NE),
          LC(NE*6), TH(NT), MD1(MD5), MD2(MD5),
          TX(NP-NW), TY(NP-NW), TZ(NP-NW)
```

The variables PH, G, and R should be allotted the same amount of space which is the maximum of (NP-NW)*6 complex words and NT* (maximum number of patterns calculated for a single value of ILD in DO loop 158) real words and N*N real words.

DO loop 8 computes L(J1) such that the L(J1)th triangular expansion function is the first triangular expansion function on the J1th wire. The axes of the wires are composed of straight line segments connecting data points. TX, TY, and TZ are the $\vec{u}_x$, $\vec{u}_y$, and $\vec{u}_z$ components of the unit tangent to a segment. BX, BY, and BZ are the normalized rectangular coordinates kx, ky, and kz of the midpoint of a segment of length AL. In DO loop 5, T(J4), T(J5), T(J6) and T(J7) are the values of the Jth triangular expansion function at the centers of the four segments over which this function extends times the segment length.

The far field $\vec{E}(\theta,\phi)$ of a filament current $\vec{u}_{\ell'} I(\ell')$ is given by

$$\vec{E}(\theta,\phi) = \frac{-j\omega\mu e^{-jkr}}{4\pi r} \int_{\substack{\text{wire}\\ \text{axes}}} \vec{E}^r(\ell') \cdot \vec{u}_{\ell'} I(\ell') d\ell' \qquad (15)$$

where $\vec{E}^r(\ell')$ is the electric field of a unit plane wave coming from the direction (θ,ϕ)

$$\vec{E}^r(\ell') = \vec{u}_r\, e^{jk(x'\sin\theta\cos\phi + y'\sin\theta\sin\phi + z'\cos\theta)} \qquad (16)$$

In (15), $\vec{u}_r$ is a unit vector specifying the polarization of $\vec{E}^r(\ell')$. The coordinates x', y', and z' depend upon ℓ'. Equation (16) may be specialized to the planes x=0, y=0, and z=0, and to the polarizations $\vec{u}_\theta$ and $\vec{u}_\phi$.

$$\vec{E}^r \cdot \vec{u}_{\ell'} = (-\vec{u}_y \cos\theta - \vec{u}_z \sin\theta) \cdot \vec{u}_{\ell'} e^{jk(-y'\sin\theta + z'\cos\theta)} \qquad \vec{u}_\theta,\ x=0 \tag{17}$$

$$\vec{E}^r \cdot \vec{u}_{\ell'} = \vec{u}_x \cdot \vec{u}_{\ell'} e^{jk(-y'\sin\theta + z'\cos\theta)} \qquad \vec{u}_\phi,\ x=0 \tag{18}$$

$$\vec{E}^r \cdot \vec{u}_{\ell'} = (\vec{u}_x \cos\theta - \vec{u}_z \sin\theta) \cdot \vec{u}_{\ell'} e^{jk(x'\sin\theta + z'\cos\theta)} \qquad \vec{u}_\theta,\ y=0 \tag{19}$$

$$\vec{E}^r \cdot \vec{u}_{\ell'} = \vec{u}_y \cdot \vec{u}_{\ell'}\ e^{jk(x'\sin\theta + z'\cos\theta)} \qquad \vec{u}_\phi,\ y=0 \tag{20}$$

$$\vec{E}^r \cdot \vec{u}_{\ell'} = -\ \vec{u}_z \cdot \vec{u}_{\ell'}\ e^{jk(x'\cos\phi + y'\sin\phi)} \qquad \vec{u}_\theta,\ z=0 \tag{21}$$

$$\vec{E}^r \cdot \vec{u}_{\ell'} = (-\vec{u}_x \sin\phi + \vec{u}_y \cos\phi) \cdot \vec{u}_\ell\ e^{jk(x'\cos\phi + y'\sin\phi)} \qquad \vec{u}_\phi,\ z=0 \tag{22}$$

Equations (17)-(22) lead respectively to the six different gain patterns promised earlier. The exponentials appearing in (17)-(22) are put in U3, U4, and U5. Next, the dot products appearing in (17)-(22) are put in S2, S3, S4, S5, S6, and S7. The quantities (17)-(22) themselves are put in the dimensioned variable PH. DO loop 28 stores the normalized electric field

$$\int_{\substack{\text{wire}\\ \text{axis}}} \vec{E}^r(\ell') \cdot \vec{u}_{\ell'}\ I(\ell')d\ell'$$

in VR((KK-1)*NT*N + (J-1)*N+I) where KK indicates the KKth of the six expressions (17)-(22), J denotes either the angle θ in (17)-(20) or the angle ϕ in (21)-(22) and I denotes that $I(\ell')$ of (15) is the Ith triangular expansion function.

The index ILD of DO loop 158 indicates that the set of eigencurrents to be read from data set 6 has been computed by program #2 from an impedance matrix to which has been added the ILDth set of loads (ZL(I), I = 1,N) to the diagonal elements. DO loop 16 stores the real and imaginary parts of the impedance matrix Z in R and X. The matrix Z is made symmetric by averaging corresponding off diagonal elements. The impedance loads ZL are added to the diagonal elements of Z. DO loop 13 puts the JCth diagonal elements of $\tilde{I}XI$ and $\tilde{I}RI$ in PP(JC) and P(JC). The quantities $\tilde{I}XI/\tilde{I}RI$ are printed so that they can be compared to the eigenvalues of

$$[X][I] = \lambda[R][I] \tag{23}$$

The index JP of DO loop 156 indicates which of the 6 gain patterns denoted by (17) - (22) is being computed. The index JC of DO loop 53 indicates the JCth eigencurrent. DO loop 54 obtains U1 by multiplying the eigencurrent FI by the receiver voltage excitation VR. The gain G is given by

$$G = \frac{k^2 \, |U1|^2}{4\pi(\tilde{I}RI)} \tag{24}$$

The phase factor $-je^{-jkr}$ is suppressed from the far field E of an eigencurrent. E is normalized so that

$$G = |E|^2 \tag{25}$$

```
//              (0034,EE,5,3),'MAUTZ,JOE',REGION=200K
// EXEC FORTGCLG,PARM.FORT='MAP'
//FORT.SYSIN DD *
      COMPLEX Z(900),U,U1,U3,U4,U5,PH(1533),VR(11388),E(145)
      COMPLEX ZL(30),R(1533)
      DIMENSION LL(5),RAD(4),PX(100),PY(100),PZ(100),L(5)
      DIMENSION BX(100),BY(100),BZ(100),AL(100),T(200),FI(900)
      DIMENSION X(900),P(30),PP(30),SN(145),CS(145),G(3066)
      DIMENSION AMD(30),LC(150),TH(145),MD1(15),MD2(15)
      DIMENSION TX(100),TY(100),TZ(100)
      EQUIVALENCE (G(1),PH(1),R(1))
      REWIND 6
      READ(1,154) MD5
      WRITE(3,75) MD5
   75 FORMAT('1MD5'/1X,I3)
      READ(1,154)(MD1(I),I=1,MD5)
      READ(1,154)(MD2(I),I=1,MD5)
      WRITE(3,77)(MD1(I),I=1,MD5)
      WRITE(3,78)(MD2(I),I=1,MD5)
   77 FORMAT('OMD1'/(1X,20I3))
   78 FORMAT('OMD2'/(1X,20I3))
      DO 143 KAP=1,MD5
      READ(1,35) NP,NT,NS,NW,LG,BK
   35 FORMAT(5I3,E14.7)
      WRITE(3,36) NP,NT,NS,NW,LG,BK
   36 FORMAT('0 NP NT NS NW LG       BK'/1X,5I3,E14.7)
      READ(1,10)(PX(I),I=1,NP)
   10 FORMAT(10F8.4)
      WRITE(3,11)(PX(I),I=1,NP)
   11 FORMAT('0PX'/(1X,10F8.4))
      READ(1,10)(PY(I),I=1,NP)
      WRITE(3,37)(PY(I),I=1,NP)
   37 FORMAT('0PY'/(1X,10F8.4))
      READ(1,10)(PZ(I),I=1,NP)
      WRITE(3,38)(PZ(I),I=1,NP)
   38 FORMAT('0PZ'/(1X,10F8.4))
      READ(1,154)(LL(I),I=1,NW)
  154 FORMAT(20I3)
      WRITE(3,73)(LL(I),I=1,NW)
   73 FORMAT('0LL'/(1X,20I3))
      LL(NW+1)=100
      READ(1,33)(RAD(I),I=1,NW)
      WRITE(3,72)(RAD(I),I=1,NW)
   72 FORMAT('0RAD'/(1X,5E14.7))
      BK5=.5*BK
      PI=3.141593
      FN=(NT-1)*LG
      DEL=2.*PI/FN
      ETA=376.730
      C1=ETA*BK*BK/4./PI
      U=(0.,1.)
      N1=0
      J4=2
      J1=1
      DO 8 J=1,NP
      IF(LL(J1)-J) 7,6,7
    6 J4=J4-1
      L(J1)=J4
      J1=J1+1
      GO TO 8
```

```
 7 N1=N1+1
   J3=J-1
   IF((N1/2*2-N1).EQ.0) J4=J4+1
   S1=PX(J)-PX(J3)
   S2=PY(J)-PY(J3)
   S3=PZ(J)-PZ(J3)
   S4=SQRT(S1*S1+S2*S2+S3*S3)
   TX(N1)=S1/S4
   TY(N1)=S2/S4
   TZ(N1)=S3/S4
   BX(N1)=BK5*(PX(J)+PX(J3))
   BY(N1)=BK5*(PY(J)+PY(J3))
   BZ(N1)=BK5*(PZ(J)+PZ(J3))
   AL(N1)=S4
 8 CONTINUE
   N=J4-2
   NZ=N*N
   L(J1)=J4
   J1=1
   J2=-2
   DO 5 J=1,N
   IF(L(J1)-J) 3,4,3
 4 J2=J2+2
   J1=J1+1
 3 J3=(J-1)*4
   J4=J3+1
   J5=J4+1
   J6=J5+1
   J7=J6+1
   K4=J2+1
   K5=K4+1
   K6=K5+1
   K7=K6+1
   S1=AL(K4)+AL(K5)
   S2=AL(K6)+AL(K7)
   T(J4)=AL(K4)*.5*AL(K4)/S1
   T(J5)=AL(K5)*(AL(K4)+.5*AL(K5))/S1
   T(J6)=AL(K6)*(AL(K7)+.5*AL(K6))/S2
   T(J7)=AL(K7)*.5*AL(K7)/S2
   J2=J2+2
 5 CONTINUE
   S2=180./PI
   DO 27 J=1,NT
   S3=(J-1)*DEL
   TH(J)=S3*S2
   SN(J)=SIN(S3)
   CS(J)=COS(S3)
27 CONTINUE
   NTN=NT*N
   DO 28 J=1,NT
   J8=(J-1)*N
   DO 30 I=1,N1
   S1=SN(J)*BY(I)
   J1=I+N1
   J2=J1+N1
   J3=J2+N1
   J4=J3+N1
   J5=J4+N1
   S2=CS(J)*BZ(I)
   S3=-S1+S2
```

```
      S4=SN(J)*BX(I)+S2
      S5=CS(J)*BX(I)+S1
      U3=COS(S3)+U*SIN(S3)
      U4=COS(S4)+U*SIN(S4)
      U5=COS(S5)+U*SIN(S5)
      S1=SN(J)*TZ(I)
      S9=CS(J)*TY(I)
      S2=-S9-S1
      S3=TX(I)
      S4=CS(J)*TX(I)-S1
      S5=TY(I)
      S6=-TZ(I)
      S7=-SN(J)*TX(I)+S9
      PH(I)=S2*U3
      PH(J1)=S3*U3
      PH(J2)=S4*U4
      PH(J3)=S5*U4
      PH(J4)=S6*U5
      PH(J5)=S7*U5
   30 CONTINUE
      J4=-2
      J5=1
      DO 49 I=1,N
      J2=(I-1)*4
      IF(L(J5)-I) 50,51,50
   51 J4=J4+2
      J5=J5+1
   50 J6=I+J8
      DO 21 KK=1,6
      VR(J6)=0.
      J3=(KK-1)*N1+J4
      DO 52 K=1,4
      K3=J3+K
      K2=J2+K
      VR(J6)=T(K2)*PH(K3)+VR(J6)
   52 CONTINUE
      J6=J6+NTN
   21 CONTINUE
      J4=J4+2
   49 CONTINUE
   28 CONTINUE
      J1=IABS(MD1(KAP))
      IF(MD1(KAP)) 55,56,57
   55 DO 58 J=1,J1
      BACKSPACE 6
   58 CONTINUE
      GO TO 56
   57 DO 59 J=1,J1
      READ(6)
   59 CONTINUE
   56 READ(6)(Z(I),I=1,NZ)
      MD6=MD2(KAP)
      DO 158 ILD=1,MD6
      READ(1,154) NE,MD3,MD4
      WRITE(3,161) NE,MD3,MD4
  161 FORMAT('ONE MD3 MD4'/1X,3I3)
      READ(1,33)(AMD(I),I=1,NE)
   33 FORMAT(5E14.7)
      WRITE(3,34)(AMD(I),I=1,NE)
   34 FORMAT('0AMD'/(1X,5E14.7))
```

```
      J1=NE*6
      READ(1,154)(LC(I),I=1,J1)
      WRITE(3,155)(LC(I),I=1,J1)
  155 FORMAT('0LC'/(1X,20I3))
      READ(1,100)(ZL(I),I=1,N)
  100 FORMAT(7E11.4)
      WRITE(3,101)(ZL(I),I=1,N)
  101 FORMAT('0ZL'/(1X,7E11.4))
      DO 16 J=1,N
      J2=(J-1)*N
      DO 18 I=1,J
      J3=J2+I
      J4=(I-1)*N+J
      U1=.5*(Z(J3)+Z(J4))
      X(J3)=AIMAG(U1)
      X(J4)=X(J3)
      R(J3)=U1
      R(J4)=R(J3)
   18 CONTINUE
      X(J3)=X(J3)+AIMAG(ZL(J))
      R(J3)=R(J3)+ZL(J)
   16 CONTINUE
      NZ1=N*NE
      J1=IABS(MD3)
      IF(MD3) 90,91,92
   90 DO 93 J=1,J1
      BACKSPACE 6
   93 CONTINUE
      GO TO 91
   92 DO 94 J=1,J1
      READ(6)
   94 CONTINUE
   91 READ(6)(FI(I),I=1,NZ1)
      J1=IABS(MD4)
      IF(MD4) 97,98,99
   97 DO 95 J=1,J1
      BACKSPACE 6
   95 CONTINUE
      GO TO 98
   99 DO 60 J=1,J1
      READ(6)
   60 CONTINUE
   98 DO 13 JC=1,NE
      J3=(JC-1)*N
      S2=0.
      S4=0.
      DO 14 J=1,N
      J1=(J-1)*N
      S1=0.
      S3=0.
      DO 15 K=1,N
      J2=K+J1
      J4=K+J3
      S1=S1+X(J2)*FI(J4)
      S3=S3+R(J2)*FI(J4)
   15 CONTINUE
      J5=J+J3
      S2=S2+S1*FI(J5)
      S4=S4+S3*FI(J5)
   14 CONTINUE
```

```
      PP(JC)=S2
      P(JC)=S4
   13 CONTINUE
      WRITE(3,102)(P(J),J=1,NE)
  102 FORMAT('OIRI'/(1X,7E11.4))
      WRITE(3,103)(PP(J),J=1,NE)
  103 FORMAT('OIXI'/(1X,7E11.4))
      DO 76 J=1,NE
      PP(J)=PP(J)/P(J)
      P(J)=ABS(P(J))
   76 CONTINUE
      WRITE(3,162)(PP(J),J=1,NE)
  162 FORMAT('OIXI/IRI'/(1X,7E11.4))
      J9=-NT
      J8=0
      NTN=NT*N
      DO 156 JP=1,6
      J7=(JP-1)*NTN
      DO 53 JC=1,NE
      J8=J8+1
      IF(LC(J8)) 150,53,150
  150 J9=J9+NT
      J1=(JC-1)*N
      DO 54 J=1,NT
      J2=(J-1)*N+J7
      U1=0.
      DO 96 K=1,N
      K1=J1+K
      K2=J2+K
      U1=U1+VR(K2)*FI(K1)
   96 CONTINUE
      E(J)=U1*SQRT(C1/P(JC))
      S5=CABS(E(J))
      K5=J+J9
      G(K5)=S5*S5
   54 CONTINUE
      WRITE(3,74) AMD(JC)
   74 FORMAT('OLAMBDA= ',E11.4)
      GO TO (61,62,63,87,84,85),JP
   61 WRITE(3,64)
   64 FORMAT(' ELECTRIC FIELD AND GAIN IN THE PLANE X=0')
      WRITE(3,70)
   70 FORMAT('0    -',10X,'-',11X,'-',11X,'-')
      GO TO 67
   62 WRITE(3,64)
      WRITE(3,71)
   71 FORMAT('0    -',10X,'/',11X,'/',11X,'/')
      GO TO 67
   63 WRITE(3,65)
   65 FORMAT(' ELECTRIC FIELD AND GAIN IN THE PLANE Y=0')
      WRITE(3,70)
      GO TO 67
   87 WRITE(3,65)
      WRITE(3,71)
      GO TO 67
   84 WRITE(3,86)
   86 FORMAT(' ELECTRIC FIELD AND GAIN IN THE PLANE Z=0')
      WRITE(3,70)
      GO TO 67
   85 WRITE(3,86)
```

```
      WRITE(3,71)
   67 WRITE(3,66)
   66 FORMAT('+    O     REAL(EO)      IMAG(EO)         GAINO')
      DO 68 K=1,NT,NS
      J2=J9+K
      WRITE(3,69) TH(K),E(K),G(J2)
   69 FORMAT(1X,F6.1,3E12.4)
   68 CONTINUE
   53 CONTINUE
  156 CONTINUE
      J2=J9+NT
      WRITE(6)(G(J),J=1,J2)
  158 CONTINUE
  143 CONTINUE
      STOP
      END
/*
//GO.FT06F001 DD DSNAME=SURC0677.ZNEW,DISP=OLD,UNIT=2314,                X
//             VOLUME=SER=SU0005,DCB=(RECFM=V,BLKSIZE=2596,LRECL=2592)
//GO.SYSIN DD *
  1
 59
  1
 55 73  4  1  2 0.1396263E+00
 -4.6587 -4.3999 -4.1411 -3.8823 -3.6235 -3.3646 -3.1058 -2.8470 -2.5882 -2.3294
 -2.0706 -1.8117 -1.5529 -1.2941 -1.0353 -0.7765 -0.5176 -0.2588 -0.0000  0.2588
  0.5176  0.7765  1.0353  1.2941  1.5529  1.8117  2.0706  2.3294  2.5882  2.8470
  3.1058  3.3646  3.6235  3.8823  4.1411  4.3999  4.6587  4.9176  5.1764  5.4352
  5.6940  5.9528  6.2117  6.4705  6.7293  6.9881  7.2469  7.5058  7.7646  8.0234
  8.2822  8.5410  8.7998  9.0587  9.3175
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000
 17.3867 16.4207 15.4548 14.4889 13.5230 12.5570 11.5911 10.6252  9.6593  8.6933
  7.7274  6.7615  5.7956  4.8296  3.8637  2.8978  1.9319  0.9659  0.0000  0.9659
  1.9319  2.8978  3.8637  4.8296  5.7956  6.7615  7.7274  8.6933  9.6593 10.6252
 11.5911 12.5570 13.5230 14.4889 15.4548 16.4207 17.3867 18.3526 19.3185 20.2844
 21.2504 22.2163 23.1822 24.1481 25.1141 26.0800 27.0459 28.0118 28.9778 29.9437
 30.9096 31.8755 32.8415 33.8074 34.7733
  1
 0.4500000E+00
  7  0  0
 0.8315814E+01 0.7697922E+00-0.2573003E+01-0.1351125E+03-0.1706178E+03
-0.5537598E+04-0.1329068E+05
  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
  1  1
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00
/*
//
```

Sample Output


```
MD5
  1

MD1
 59

MD2
  1

 NP  NT  NS  NW  LG        BK
 55  73   4   1   2 0.1396263E 00

PX
 -4.6587 -4.3999 -4.1411 -3.8823 -3.6235 -3.3646 -3.1058 -2.8470 -2.5882 -2.3294
 -2.0706 -1.8117 -1.5529 -1.2941 -1.0353 -0.7765 -0.5176 -0.2588  0.0      0.2588
  0.5176  0.7765  1.0353  1.2941  1.5529  1.8117  2.0706  2.3294  2.5882  2.8470
  3.1058  3.3646  3.6235  3.8823  4.1411  4.3999  4.6587  4.9176  5.1764  5.4352
  5.6940  5.9528  6.2117  6.4705  6.7293  6.9881  7.2469  7.5058  7.7646  8.0234
  8.2822  8.5410  8.7998  9.0587  9.3175

PY
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0

PZ
 17.3867 16.4207 15.4548 14.4889 13.5230 12.5570 11.5911 10.6252  9.6593  8.6933
  7.7274  6.7615  5.7956  4.8296  3.8637  2.8978  1.9319  0.9659  0.0      0.9659
  1.9319  2.8978  3.8637  4.8296  5.7956  6.7615  7.7274  8.6933  9.6593 10.6252
 11.5911 12.5570 13.5230 14.4889 15.4548 16.4207 17.3867 18.3526 19.3185 20.2844
 21.2504 22.2163 23.1822 24.1481 25.1141 26.0800 27.0459 28.0118 28.9778 29.9437
 30.9096 31.8755 32.8415 33.8074 34.7733

LL
  1

RAD
 0.4500000E 00

NE MD3 MD4
  7  C   0

AMD
 0.8315813E 01 0.7697922E 00-0.2573003E 01-0.1351125E 03-0.1706178E 03
-0.5537598E 04-0.1329068E 05

LC
  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
  1  1

ZL
 0.0          0.0          0.0          0.0          0.0          0.0          0.0
 0.0          0.0          0.0          0.0          0.0          0.0          0.0
 0.0          0.0          0.0          0.0          0.0          0.0          0.0
 0.0          0.0          0.0          0.0          0.0          0.0          0.0
 0.0          0.0          0.0          0.0          0.0          0.0          0.0
 0.0          0.0          0.0          0.0          0.0          0.0          0.0
```

```
0.0          0.0          0.0          0.0          0.0          0.0          0.0
0.0          0.0          0.0

IRI
0.2086E 02 0.1370E 03 0.7206E 02 0.6654E 01 0.1604E 01 0.2239E 00 0.5695E-01

IXI
0.1734E 03 0.1055E 03-0.1854E 03-0.8983E 03-0.2730E 03-0.1266E 04-0.7937E 03

IXI/IRI
0.8316E 01 0.7693E 00-0.2573E 01-0.1351E 03-0.1706E 03-0.5533E 04-0.1324E 05
```

LAMBDA= 0.8316E 01
ELECTRIC FIELD AND GAIN IN THE PLANE X=0

θ	REAL(Eθ)	IMAG(Eθ)	GAINθ
0.0	0.0	0.0	0.0
10.0	0.1531E-02	-0.4014E-01	0.2132E-02
20.0	-0.9614E-02	-0.3694E-01	0.7651E-02
30.0	-0.4043E-01	-0.1121E 00	0.1420E-01
40.0	-0.8521E-01	-0.1073E 00	0.1878E-01
50.0	-0.1220E 00	-0.6524E-01	0.1913E-01
60.0	-0.1224E 00	-0.3818E-03	0.1498E-01
70.0	-0.7715E-01	0.4905E-01	0.8359E-02
80.0	-0.1491E-01	0.4780E-01	0.2507E-02
90.0	0.1427E-01	0.1033E-06	0.2037E-03
100.0	-0.1491E-01	-0.4780E-01	0.2507E-02
110.0	-0.7715E-01	-0.4905E-01	0.8359E-02
120.0	-0.1224E 00	0.3816E-03	0.1498E-01
130.0	-0.1220E 00	0.6524E-01	0.1913E-01
140.0	-0.8521E-01	0.1073E 00	0.1878E-01
150.0	-0.4043E-01	0.1121E 00	0.1420E-01
160.0	-0.9614E-02	0.3694E-01	0.7651E-02
170.0	0.1531E-02	0.4014E-01	0.2132E-02
180.0	0.1371E-07	0.1685E-06	0.2857E-13

LAMBDA= 0.7698E 00
ELECTRIC FIELD AND GAIN IN THE PLANE X=0

θ	REAL(Eθ)	IMAG(Eθ)	GAINθ
0.0	0.0	0.0	0.0
10.0	0.1186E 00	-0.5315E-01	0.1689E-01
20.0	0.2317E 00	-0.1431E 00	0.7419E-01
30.0	0.3150E 00	-0.3027E 00	0.1909E 00
40.0	0.3125E 00	-0.5436E 00	0.3932E 00
50.0	0.1480E 00	-0.8238E 00	0.7005E 00
60.0	-0.2243E 00	-0.1025E 01	0.1100E 01
70.0	-0.7433E 00	-0.9862E 00	0.1526E 01
80.0	-0.1217E 01	-0.5162E 00	0.1661E 01
90.0	-0.1410E 01	-0.1195E-05	0.1989E 01
100.0	-0.1217E 01	0.5162E 00	0.1661E 01
110.0	-0.7433E 00	0.9862E 00	0.1526E 01
120.0	-0.2243E 00	0.1025E 01	0.1100E 01
130.0	0.1480E 00	0.8238E 00	0.7005E 00
140.0	0.3125E 00	0.5436E 00	0.3932E 00
150.0	0.3150E 00	0.3027E 00	0.1909E 00
160.0	0.2313E 00	0.1431E 00	0.7419E-01
170.0	0.1186E 00	0.5315E-01	0.1689E-01
180.0	0.4277E-06	0.1691E-06	0.2115E-12

plus 15 more pages

V. SCATTERING CROSS SECTIONS

Program #4 calculates the bistatic scattering cross section per square wavelength σ/λ^2 for a plane wave traveling in the $\pm z$ direction with its electric vector parallel to the x axis. Only σ_ϕ/λ^2 in the plane x=0 and σ_θ/λ^2 in the plane y=0 are considered. Here, σ_θ is obtained from the $\vec{u}_\theta$ component of the scattered field and σ_ϕ from the $\vec{u}_\phi$ component. The cross sections obtained by using an admittance matrix Y^a which is an eigencurrent approximation to Z^{-1} will be compared to those obtained by using Z^{-1} itself. The pth column of the admittance matrix Y is composed of the expansion coefficients of the current $\vec{J}$ which results when

$$\begin{aligned} \int \vec{E}^i \cdot \vec{W}_n d\ell &= 0 \qquad n \neq p \\ \int \vec{E}^i \cdot \vec{W}_n d\ell &= 0 \qquad n = p \end{aligned} \tag{26}$$

where $\vec{E}^i$ is the electric field that supports $\vec{J}$ and $\{\vec{W}_n\}$ is the set of testing functions. For eigencurrents $\vec{J}_n$ which are not normalized, equation (1-30) of [2] is

$$\vec{J} = \sum_n \frac{\vec{J}_n \int \vec{J}_n \cdot \vec{E}^i d\ell}{(\tilde{I}RI)_n \ (1+j\lambda_n)} \tag{27}$$

where $(\tilde{I}RI)_n$ is the nth diagonal element of the matrix $[\tilde{I}RI]$ in which R is the real part of Z and I is the eigencurrent matrix.

$$\vec{J}_n = \sum_m I_{mn} \vec{F}_m \tag{28}$$

If the set $\{W_m\}$ of testing functions is the same as the set $\{\vec{F}_m\}$ of expansion functions, equations (26), (27), and (28) lead to

$$\vec{J} = \sum_m \left(\sum_n \frac{I_{mn} \ I_{pn}}{(\tilde{I}RI)_n (1+j\lambda_n)} \right) \vec{F}_m \tag{29}$$

which implies that

$$Y = I \left[\frac{1}{(\tilde{I}RI)_n(1+j\lambda_n)}\right]\tilde{I} \qquad (30)$$

where $\left[\frac{1}{(\tilde{I}RI)_n(1+j\lambda_n)}\right]$ is a square diagonal matrix whose nth diagonal element is $\frac{1}{(\tilde{I}RI)_n(1+j\lambda_n)}$. An approximation Y^a is obtained by suppressing certain eigencurrents from the sum (27) which is equivalent to replacing certain of the numbers $\frac{1}{(\tilde{I}RI)_n(1+j\lambda_n)}$ in (30) by zero.

The activity on data sets 1 (punched card input) and 6 (unformated direct access input-output) is described as follows:

```
      READ(1,64) MD5
64    FORMAT(20I3)
      READ(1,64)(MD1(I), I = 1, MD5)
      READ(1,64)(MD2(I), I=1, MD5)
      REWIND 6
      DO 143 KAP = 1, MD5
      READ(1,62) NP, NT, NS, LS, NW, BK
62    FORMAT(5I3, E14.7)
      READ(1,10)(PX(I), I = 1, NP)
10    FORMAT(10F8.4)
      READ(1,10)(PY(I), I = 1, NP)
      READ(1,10)(PZ(I), I = 1, NP)
      READ(1,64)(LL(I), I = 1, NW)
      READ(1,70)(RAD(I), I = 1, NW)
      SKIP MD1(KAP) RECORDS ON DATA SET 6
      NZ = N*N
      READ(6)(Z(I), I = 1, NZ)
      MD6 = MD2(KAP)
      DO 152 ILD = 1, MD6
      READ(1,64) NE, NC, INC, MD3, MD4
```

```
      READ(1,70)(AMD(I), I = 1, NE)
 70   FORMAT(5E14.7)
      READ(1,64)(LR(I), I = 1, NC)
      READ(1,108)(ZL(I), I = 1, N)
108   FORMAT(7E11.4)
      SKIP MD3 RECORDS ON DATA SET 6
      NZ1 = N*NE
      READ(6) (FI(I), I = 1, NZ1)
      SKIP MD4 RECORDS ON DATA SET 6
      WRITE(6)(SIG(J), J = 1, J8)
152   CONTINUE
143   CONTINUE
```

PX, PY, and PZ are the x,y, and z coordinates of the NP data points that describe the NW wires. Using the admittance matrix Z^{-1}, σ/λ^2 is computed at NT equally spaced angles θ of (18) and (19) between 0° and 360°/LS. Using NC approximate admittance matrices Y^a specified by LR, σ/λ^2 is also computed at $\frac{NT-1}{NS} + 1$ equally spaced angles between 0° and 360°/LS. Both σ/λ^2 computed using Z^{-1} and σ/λ^2 computed using Y^a are printed at $\frac{NT-1}{NS} + 1$ equally spaced angles between 0° and 360°/LS. To be compatible with the plot program #5, LS should be 1. BK is the propagation constant. The LL(I)th data point marks the beginning of the Ith wire of radius RAD(I). The impedance matrix Z of order N has been computed by program #1. Program #2 has stored NE eigencurrents in FI. The incident plane wave has $e^{j(INC)kz}$ dependence. AMD appears under the heading "Eigenvalues of the matrix B" in the printed output of program #2. The Ith approximate admittance matrix Y^a is computed using only the LR(1)th, LR(2)th, LR(3)th ... LR(I)th eigencurrents. The impedance load to be added to the Ith diagonal element of Z is read in through the complex variable ZL(I). The set FI of NE eigencurrents has been computed by program #2. The present program stores σ/λ^2 in SIG. The σ/λ^2 computed using Z^{-1} is stored in SIG(1) to SIG(NT) for the $\vec{u}_\phi$ polarization in the x=0 plane and in SIG(1 + NT + M*NC) to SIG(2*NT + M*NC) for the $\vec{u}_\theta$ polarization in the y=0 plane. Here,

$M = \frac{NT-1}{NS} + 1$ is the number of points on a pattern computed using Y^a. Also, σ/λ^2 computed using the Ith approximate admittance matrix Y^a is stored in SIG(1 + NT + M*(I-1)) to SIG(NT + M*I) for the $\vec{u}_\phi$ polarization in the x=0 plane and in SIG(1 + 2*NT + M*(NC + I-1)) to SIG(2*NT + M*(NC+I)) for the $\vec{u}_\theta$ polarization in the plane y=0. There are J8 = 2*(NT + M*NC) words stored in SIG.

Minimum allocations are given by

```
   COMPLEX Z(N*N), VR(2*NT*N), T3(N*NE), T4((JC+1)*N),
           E(N), E(NT), ZL(N), E3(N)
   DIMENSION LL(NW+1), RAD(NW), PX(NP), PY(NP), PZ(NP),
           L(NW+1), BX(NP-NW), BY(NP-NW), BZ(NP-NW),
           AL(NP-NW), T(4*N), FI(N*NE), R(N*N),
           SN(NT), CS(NT), E2(N), LR(NC), TX(NP-NW),
           TY(NP-NW), TZ(NP-NW), AMD(NE), TH(NT),
           MD1(MD5), MD2(MD5)
```

The complex variable E must be assigned the maximum of the two allocations designated. Since the variables ZZ and SIG appear in the equivalence statement, they should be allocated the same number of spaces which is the maximum of 2*(NP-NW) complex words, N*N complex words, and 2*(NT + NC* $(\frac{NT-1}{NS+1})$) real words.

DO loop 8 is identical to DO loop 8 in program #3. Except for the calculation of E2, DO loop 5 is identical to DO loop 5 in program #3. E2 represents the length variable along the axes of the wires. DO loop 30 stores expression (18) in ZZ(I) and expression (19) in ZZ(J1). The variable VR is the same as VR in program #3 except that now only the expressions (18)-(19) instead of (17)-(22) are dealt with.

DO loop 16 stores the loaded impedance matrix made symmetric in ZZ. R is merely the real part of ZZ. The statement CALL LINEQ(N,ZZ) inverts the matrix ZZ. In DO loop 87, E3 is VR for the special case in which $\vec{E}^r(\ell')$ of (16) is the incident plane wave. DO loop 91 stores the expansion coefficients of the electric current in T4. The coefficients T4(1) to T4(N) are obtained by using the admittance matrix which is the inverse of the impedance matrix. DO loop 73 stores the expression

$$\frac{1}{(\tilde{I}RI)_J(1+j\lambda_J)}$$

appearing in (30) in E(J). DO loop 81 stores the matrix

$$I[\frac{1}{(\tilde{I}RI)_n(1+j\lambda_n)}]$$

in T3. DO loop 93 stores the expansion coefficients of the electric current obtained by using the JCth approximate admittance matrix Y^a in T4(JC*N+1) to T4(JC*N+N).

Nested DO loops 20 and 53 compute σ/λ^2. JP=1 obtains the $\vec{u}_\phi$ polarization in the plane x=0 and JP=2 obtains the $\vec{u}_\theta$ polarization in the plane y=0. The index JC indicates that T4((JC-1)*N+1) to T4(JC*N) will be used. The constant C1 is necessary in DO loop 54 because

$$\frac{\sigma}{\lambda^2} = \frac{k^4\eta^2}{16\pi^3} \left| \int_{\substack{\text{wire}\\\text{axis}}} \vec{E}^r(\ell') \cdot \vec{u}_{\ell'} \, I(\ell')d\ell' \right|^2 \tag{31}$$

where $\vec{E}^r(\ell')$ is a unit plane wave coming from the direction in which σ/λ^2 is to be evaluated, $\vec{u}_{\ell'}$ is the unit vector tangent to the wire axis at ℓ' and $I(\ell')$ is the filament current at ℓ' along the axis of the wire. The phase factor $-je^{-jkr}$ is suppressed from the scattered field E. E is normalized so that

$$\frac{\sigma}{\lambda^2} = |E|^2 \tag{32}$$

Listing of Program #4

```
//                  (0034,EE,5,3),'MAUTZ,JOE',REGION=200K
// EXEC FORTGCLG,PARM.FORT='MAP'
//FORT.SYSIN DD *
      SUBROUTINE LINEQ(LL,C)
      COMPLEX C(1),STOR,STO,ST,S
      DIMENSION LR(40)
      DO 20 I=1,LL
      LR(I)=I
   20 CONTINUE
      M1=0
      DO 18 M=1,LL
      K=M
      DO 2 I=M,LL
      K1=M1+I
      K2=M1+K
      IF(CABS(C(K1))-CABS(C(K2))) 2,2,6
    6 K=I
    2 CONTINUE
      LS=LR(M)
      LR(M)=LR(K)
      LR(K)=LS
      K2=M1+K
      STOR=C(K2)
      J1=0
      DO 7 J=1,LL
      K1=J1+K
      K2=J1+M
      STO=C(K1)
      C(K1)=C(K2)
      C(K2)=STO/STOR
      J1=J1+LL
    7 CONTINUE
      K1=M1+M
      C(K1)=1./STOR
      DO 11 I=1,LL
      IF(I-M) 12,11,12
   12 K1=M1+I
      ST=C(K1)
      C(K1)=0.
      J1=0
      DO 10 J=1,LL
      K1=J1+I
      K2=J1+M
      C(K1)=C(K1)-C(K2)*ST
      J1=J1+LL
   10 CONTINUE
   11 CONTINUE
      M1=M1+LL
   18 CONTINUE
      J1=0
      DO 9 J=1,LL
      IF(J-LR(J)) 14,8,14
   14 LRJ=LR(J)
      J2=(LRJ-1)*LL
   21 DO 13 I=1,LL
      K2=J2+I
      K1=J1+I
      S=C(K2)
      C(K2)=C(K1)
      C(K1)=S
```

```
 13 CONTINUE
    LR(J)=LR(LRJ)
    LR(LRJ)=LRJ
    IF(J-LR(J)) 14,8,14
  8 J1=J1+LL
  9 CONTINUE
    RETURN
    END
    COMPLEX Z(900),ZZ(900),U,U1,U3,U4,VR(7540),T3(900),T4(900)
    COMPLEX E(145),ZL(30),E3(30)
    DIMENSION LL(5),RAD(4),PX(100),PY(100),PZ(100),L(5)
    DIMENSION BX(100),BY(100),BZ(100),AL(100),T(200),FI(900)
    DIMENSION R(900),SN(145),CS(145),SIG(1800),E2(30),LR(30)
    DIMENSION TX(100),TY(100),TZ(100),AMD(30),TH(145),MD1(30),MD2(30)
    EQUIVALENCE (ZZ(1),SIG(1))
    REWIND 6
    READ(1,64) MD5
 64 FORMAT(20I3)
    WRITE(3,105) MD5
105 FORMAT('1MD5'/1X,I3)
    READ(1,64)(MD1(I),I=1,MD5)
    READ(1,64)(MD2(I),I=1,MD5)
    WRITE(3,106)(MD1(I),I=1,MD5)
    WRITE(3,107)(MD2(I),I=1,MD5)
106 FORMAT('OMD1'/(1X,20I3))
107 FORMAT('OMD2'/(1X,20I3))
    DO 143 KAP=1,MD5
    READ(1,62) NP,NT,NS,LS,NW,BK
 62 FORMAT(5I3,E14.7)
    WRITE(3,63) NP,NT,NS,LS,NW,BK
 63 FORMAT('O NP NT NS LS NW       BK'/1X,5I3,E14.7)
    READ(1,10)(PX(I),I=1,NP)
 10 FORMAT(10F8.4)
    WRITE(3,11)(PX(I),I=1,NP)
 11 FORMAT('OPX'/(1X,10F8.4))
    READ(1,10)(PY(I),I=1,NP)
    WRITE(3,60)(PY(I),I=1,NP)
 60 FORMAT('OPY'/(1X,10F8.4))
    READ(1,10)(PZ(I),I=1,NP)
    WRITE(3,61)(PZ(I),I=1,NP)
 61 FORMAT('OPZ'/(1X,10F8.4))
    READ(1,64)(LL(I),I=1,NW)
    WRITE(3,66)(LL(I),I=1,NW)
 66 FORMAT('OLL'/(1X,20I3))
    LL(NW+1)=200
    READ(1,70)(RAD(I),I=1,NW)
    WRITE(3,67)(RAD(I),I=1,NW)
 67 FORMAT('ORAD'/(1X,5E14.7))
    PI=3.141593
    FN=(NT-1)*LS
    DEL=2.*PI/FN
    ETA=376.730
    U=(0.,1.)
    BK5=.5*BK
    BK2=BK*BK
    C1=BK2*ETA/4./SQRT(PI**3)
    N1=0
    J4=2
    J1=1
    DO 8 J=1,NP
```

```
      IF(LL(J1)-J) 7,6,7
    6 J4=J4-1
      L(J1)=J4
      J1=J1+1
      GO TO 8
    7 N1=N1+1
      J3=J-1
      IF((N1/2*2-N1).EQ.0) J4=J4+1
      S1=PX(J)-PX(J3)
      S2=PY(J)-PY(J3)
      S3=PZ(J)-PZ(J3)
      S4=SQRT(S1*S1+S2*S2+S3*S3)
      TX(N1)=S1/S4
      TY(N1)=S2/S4
      TZ(N1)=S3/S4
      BX(N1)=BK5*(PX(J)+PX(J3))
      BY(N1)=BK5*(PY(J)+PY(J3))
      BZ(N1)=BK5*(PZ(J)+PZ(J3))
      AL(N1)=S4
    8 CONTINUE
      N=J4-2
      L(J1)=J4
      J1=1
      J2=-2
      S3=0.
      DO 5 J=1,N
      IF(L(J1)-J) 3,4,3
    4 J2=J2+2
      J1=J1+1
    3 J4=(J-1)*4+1
      J5=J4+1
      J6=J5+1
      J7=J6+1
      K4=J2+1
      K5=K4+1
      K6=K5+1
      K7=K6+1
      S1=AL(K4)+AL(K5)
      S3=S3+S1
      F2(J)=S3
      S2=AL(K6)+AL(K7)
      T(J4)=AL(K4)*.5*AL(K4)/S1
      T(J5)=AL(K5)*(AL(K4)+.5*AL(K5))/S1
      T(J6)=AL(K6)*(AL(K7)+.5*AL(K6))/S2
      T(J7)=AL(K7)*.5*AL(K7)/S2
      J2=J2+2
    5 CONTINUE
      S1=5./E2(N)
      DO 151 J=1,N
      E2(J)=F2(J)*S1
  151 CONTINUE
      WRITE(3,72)(E2(J),J=1,N)
   72 FORMAT('0WIRE LENGTH VARIABLE'/(1X,7E11.4))
      S1=180./PI
      DO 27 J=1,NT
      S3=(J-1)*DEL
      TH(J)=S3*S1
      SN(J)=SIN(S3)
      CS(J)=COS(S3)
   27 CONTINUE
```

```
      NTN=NT*N
      DO 28 J=1,NT
      J8=(J-1)*N
      DO 30 I=1,N1
      J1=I+N1
      S2=CS(J)*BZ(I)
      S3=-SN(J)*BY(I)+S2
      S4=SN(J)*BX(I)+S2
      U3=COS(S3)+U*SIN(S3)
      U4=COS(S4)+U*SIN(S4)
      S3=TX(I)
      S4=CS(J)*TX(I)-SN(J)*TZ(I)
      ZZ(I)=S3*U3
      ZZ(J1)=S4*U4
   30 CONTINUE
      J4=-2
      J5=1
      DO 49 I=1,N
      J2=(I-1)*4
      IF(L(J5)-I) 50,51,50
   51 J4=J4+2
      J5=J5+1
   50 J6=I+J8
      J7=J6+NTN
      VR(J6)=0.
      VR(J7)=0.
      DO 52 K=1,4
      K3=J4+K
      K4=K3+N1
      K2=J2+K
      VR(J6)=T(K2)*ZZ(K3)+VR(J6)
      VR(J7)=T(K2)*ZZ(K4)+VR(J7)
   52 CONTINUE
      J4=J4+2
   49 CONTINUE
   28 CONTINUE
      NZ=N*N
      J1=IABS(MD1(KAP))
      IF(MD1(KAP)) 110,111,112
  110 DO 113 J=1,J1
      BACKSPACE 6
  113 CONTINUE
      GO TO 111
  112 DO 114 J=1,J1
      READ(6)
  114 CONTINUE
  111 READ(6)(Z(I),I=1,NZ)
      MD6=MD2(KAP)
      DO 152 ILD=1,MD6
      READ(1,64) NE,NC,INC,MD3,MD4
      WRITE(3,153) NE,NC,INC,MD3,MD4
  153 FORMAT('0  NE  NC INC MD3 MD4'/1X,5I4)
      READ(1,70)(AMD(I),I=1,NE)
   70 FORMAT(5E14.7)
      WRITE(3,71)(AMD(I),I=1,NE)
   71 FORMAT('0AMD'/(1X,5E14.7))
      READ(1,64)(LR(I),I=1,NC)
      WRITE(3,65)(LR(I),I=1,NC)
   65 FORMAT('0LR'/(1X,20I3))
      READ(1,108)(ZL(I),I=1,N)
```

```
  108 FORMAT(7E11.4)
      WRITE(3,109)(ZL(I),I=1,N)
  109 FORMAT('0ZL'/(1X,7E11.4))
      DO 16 J=1,N
      J2=(J-1)*N
      DO 18 I=1,J
      J3=J2+I
      J4=(I-1)*N+J
      U1=.5*(Z(J3)+Z(J4))
      R(J3)=U1
      R(J4)=R(J3)
      ZZ(J3)=U1
      ZZ(J4)=U1
   18 CONTINUE
      ZZ(J3)=Z(J3)+ZL(J)
      R(J3)=R(J3)+ZL(J)
   16 CONTINUE
      CALL LINEQ(N,ZZ)
      C2=90.*(1-INC)
      U1=INC*U
      DO 87 J=1,N
      E3(J)=REAL(VR(J))+U1*AIMAG(VR(J))
   87 CONTINUE
      DO 91 J=1,N
      T4(J)=0.
      DO 92 K=1,N
      J2=(K-1)*N+J
      T4(J)=T4(J)+E3(K)*ZZ(J2)
   92 CONTINUE
   91 CONTINUE
      J1=IABS(MD3)
      IF(MD3) 115,116,117
  115 DO 118 J=1,J1
      BACKSPACE 6
  118 CONTINUE
      GO TO 116
  117 DO 119 J=1,J1
      READ(6)
  119 CONTINUE
  116 NZ1=N*NE
      READ(6)(FI(I),I=1,NZ1)
      J1=IABS(MD4)
      IF(MD4) 120,121,122
  120 DO 123 J=1,J1
      BACKSPACE 6
  123 CONTINUE
      GO TO 121
  122 DO 124 J=1,J1
      READ(6)
  124 CONTINUE
  121 DO 73 J=1,NE
      J1=(J-1)*N
      E2(J)=0.
      DO 74 I=1,N
      S1=0.
      J4=(I-1)*N
      DO 75 K=1,N
      J3=J1+K
      J2=J4+K
      S1=S1+R(J2)*FI(J3)
```

```
 75 CONTINUE
    J2=J1+I
    E2(J)=E2(J)+S1*FI(J2)
 74 CONTINUE
    E(J)=1./(1.+U*AMD(J))/E2(J)
 73 CONTINUE
    DO 81 J=1,NE
    J1=(J-1)*N
    DO 82 I=1,N
    J2=J1+I
    T3(J2)=FI(J2)*E(J)
 82 CONTINUE
 81 CONTINUE
    DO 39 J=1,NZ
    ZZ(J)=0.
 39 CONTINUE
    DO 93 JC=1,NC
    J1=JC*N
    J7=(LR(JC)-1)*N
    DO 94 I=1,N
    J8=J1+I
    T4(J8)=0.
    J5=J7+I
    DO 95 J=1,N
    J6=J7+J
    J3=(J-1)*N+I
    ZZ(J3)=FI(J5)*T3(J6)+ZZ(J3)
    T4(J8)=T4(J8)+ZZ(J3)*E3(J)
 95 CONTINUE
 94 CONTINUE
 93 CONTINUE
    NC1=NC+1
    J8=0
    DO 20 JP=1,2
    J3=(JP-1)*NTN
    DO 53 JC=1,NC1
    J1=(JC-1)*N
    IF(JC-1) 150,125,150
125 NSK=1
    WRITE(3,68) C2
 68 FORMAT('OSCATTERED FIELD AND SCATTERING CS/W2'/' INCIDENCE FROM 0=
   1',F4.0)
    GO TO 69
150 NSK=NS
    JC1=JC-1
    WRITE(3,76) JC1,C2
 76 FORMAT('0',I3,' MODE SCATTERED FIELD AND SCATTERING CS/W2'/' INCID
   1ENCE FROM 0=',F4.0)
 69 WRITE(3,77)
 77 FORMAT('+',15X,'-')
    J9=J8+1
    DO 54 J=1,NT,NSK
    J8=J8+1
    J2=(J-1)*N+J3
    U1=0.
    DO 96 K=1,N
    K1=J1+K
    K2=J2+K
    U1=U1+VR(K2)*T4(K1)
 96 CONTINUE
```

```
      E(J)=U1*C1
      S5=CABS(E(J))
      SIG(J8)=S5*S5
   54 CONTINUE
      GO TO (55,56),JP
   55 WRITE(3,58)
   58 FORMAT(' THIS PATTERN IS IN THE PLANE X=0')
      WRITE(3,59)
   59 FORMAT('0    /',10X,'/',11X,'/',5X,'/')
      GO TO 57
   56 WRITE(3,89)
   89 FORMAT(' THIS PATTERN IS IN THE PLANE Y=0')
      WRITE(3,24)
   24 FORMAT('0    -',10X,'-',11X,'-',5X,'-')
   57 WRITE(3,80)
   80 FORMAT('+    O      REAL(EO)      IMAG(EO)     SO/(LAM)**2')
      NS1=NS/NSK
      DO 26 J=1,NT,NS
      WRITE(3,33) TH(J),E(J),SIG(J9)
   33 FORMAT(1X,F6.1,3E12.4)
      J9=J9+NS1
   26 CONTINUE
   53 CONTINUE
   20 CONTINUE
      WRITE(6)(SIG(J),J=1,J8)
  152 CONTINUE
  143 CONTINUE
      STOP
      END
/*
//GO.FT06F001 DD DSNAME=SURC0677.ZNEW,DISP=OLD,UNIT=2314,                X
//               VOLUME=SER=SU0005,DCB=(RECFM=V,BLKSIZE=2596,LRECL=2592)
//GO.SYSIN DD *
  1
 59
  1
 55145   4   1   1 0.1396263E+00
 -4.6587 -4.3999 -4.1411 -3.8823 -3.6235 -3.3646 -3.1058 -2.8470 -2.5882 -2.3294
 -2.0706 -1.8117 -1.5529 -1.2941 -1.0353 -0.7765 -0.5176 -0.2588 -0.0000  0.2588
  0.5176  0.7765  1.0353  1.2941  1.5529  1.8117  2.0706  2.3294  2.5882  2.8470
  3.1058  3.3646  3.6235  3.8823  4.1411  4.3999  4.6587  4.9176  5.1764  5.4352
  5.6940  5.9528  6.2117  6.4705  6.7293  6.9881  7.2469  7.5058  7.7646  8.0234
  8.2822  8.5410  8.7998  9.0587  9.3175
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000  0.0000
  0.0000  0.0000  0.0000  0.0000  0.0000
 17.3867 16.4207 15.4548 14.4889 13.5230 12.5570 11.5911 10.6252  9.6593  8.6933
  7.7274  6.7615  5.7956  4.8296  3.8637  2.8978  1.9319  0.9659  0.0000  0.9659
  1.9319  2.8978  3.8637  4.8296  5.7956  6.7615  7.7274  8.6933  9.6593 10.6252
 11.5911 12.5570 13.5230 14.4889 15.4548 16.4207 17.3867 18.3526 19.3185 20.2844
 21.2504 22.2163 23.1822 24.1481 25.1141 26.0800 27.0459 28.0118 28.9778 29.9437
 30.9096 31.8755 32.8415 33.8074 34.7733
  1
 0.4500000E+00
  7  7 -1  0  1
 0.8315814E+01 0.7697922E+00-0.2573003E+01-0.1351125E+03-0.1706178E+03
-0.5537598E+04-0.1329068E+05
```

```
  2  3  1  4  5  6  7
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00 0.0000E+00
 0.0000E+00 0.0000E+00 0.0000E+00
/*
//
```

Sample Output

```
MD5
  3

MD1
 60  0  0

MD2
  1  2  3

  N NF NE1
 26  7  7

L4
  1  2  3  4  5  6  7

X
 0.1923E 00 0.3846E 00 0.5769E 00 0.7692E 00 0.9615E 00 0.1154E 01 0.1346E 01
 0.1538E 01 0.1731E 01 0.1923E 01 0.2115E 01 0.2308E 01 0.2500E 01 0.2692E 01
 0.2885E 01 0.3077E 01 0.3269E 01 0.3462E 01 0.3654E 01 0.3846E 01 0.4038E 01
 0.4231E 01 0.4423E 01 0.4615E 01 0.4808E 01 0.5000E 01

FI
 0.2394E 00 0.4109E 00 0.5798E 00 0.7288E 00 0.8511E 00 0.9395E 00 0.9894E 00 0.1000E 01 0.9703E 00 0.9819E 00
 0.9657E 00 0.9185E 00 0.8426E 00 0.7426E 00 0.6245E 00 0.4959E 00 0.3646E 00 0.2381E 00 0.1230E 00 0.2489E-01

LP LP1 NT
 42 35145

L4
  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
 21 29 30 31 32 33 34 35 36 37 38 39 40 41 42

SCAL
 0.5000E 02 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01
 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01
 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01
 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01 0.1000E 01
 0.1000E 01 0.5000E 02 0.5000E 02 0.1000E 01 0.1000E 01 0.5000E 02 0.1000E 01

G
 0.0        0.1377E-03 0.5470E-03 0.1218E-02 0.2132E-02 0.3265E-02 0.4567E-02 0.6063E-02 0.7651E-02 0.9307E-02
 0.1098E-01 0.1263E-01 0.1420E-01 0.1564E-01 0.1691E-01 0.1797E-01 0.1878E-01 0.1932E-01 0.1956E-01 0.1950E-01

ONE INCH CORRESPONDS TO A GAIN OF 0.2000E-01

ONE INCH CORRESPONDS TO A GAIN OF 0.1000E 01

ONE INCH CORRESPONDS TO A GAIN OF 0.1000E 01

ONE INCH CORRESPONDS TO A GAIN OF 0.1000E 01

ONE INCH CORRESPONDS TO A GAIN OF 0.1000E 01
```

plus two more pages

VI. PLOTS OF EIGENCURRENTS, GAIN EIGENPATTERNS, AND SCATTERING CROSS SECTIONS

The present program #5 plots the eigencurrents FI, the gain eigenpatterns G, and the scattering cross sections per square wavelength SIG previously computed by programs #2, #3, and #4 respectively. The activity on data sets 1 (punched card input) and 6 (unformated direct access input-output) is described as follows:

```
        READ(1,11) MD5
 11     FORMAT(20I3)
        READ(1,11)(MD1(I), I = 1, MD5)
        READ(1,11)(MD2(I), I = 1, MD5)
        REWIND 6
        DO 80 KAP = 1, MD5
        SKIP MD1(KAP) RECORDS ON DATA SET 6
        MD4 = MD2(KAP)
        GO TO (87, 88, 89), MD4
 87     READ(1,11) N, NE, NE1
        READ(1,11)(L4(I), I = 1, NE1)
        READ(1,13)(X(I), I = 1, N)
 13     FORMAT(7E11.4)
        NZ = N + NE
        READ(6)(FI(I), I = 1, NZ)
        GO TO 80
 88     READ(1,11) LP, LP1, NT
        READ(1,11)(L4(I), I = 1, LP1)
        READ(1,13)(SCAL(I), I = 1, LP1)
        N7 = (NT-1)/2 + 1
        NZ = N7*LP
        READ(6)(G(I), I = 1, NZ)
        GO TO 80
 89     READ(1,11) NC, NC1, NT, NS
        READ(1,11)(L4(I), I = 1, NC1)
```

```
      N8 = (NT-1)/NS + 1
      N4 = NT + N8*NC
      NZ = N4*2
      READ(6)(SIG(I), I = 1, NZ)
80    CONTINUE
```

In read statement 87, N is the order of the impedance matrix and NE is the number of eigencurrents calculated by program #2. The L4(I)th eigencurrent in FI will be the Ith eigencurrent to be plotted. The variable X appears under the heading "wire length variable" in the printed output of program #4. The present variable X is called E2 in program #4. In calculating E2, program #4 has accounted for the arc length between the beginning of a wire and the peak of the first triangular expansion function on that wire but has omitted the arc length between the peak of the last expansion function on the wire and the end of the wire. Figure 2 shows the Calcomp plots of the first six mode currents for the bent wire of Fig. 1.

In read statement 88, LP is the number of eigenpatterns computed by program #3. The L4(I)th eigenpattern of G is the Ith eigenpattern to be plotted. The pattern corresponding to L4(I) is multiplied by SCAL(I) before plotting. The N7 points on each eigenpattern computed by program #3 have to be equally spaced between 0° and 180°. The angles 0° and 180° should be included in the N7 points.

Program #4 has stored 2*(NC+1) patterns of σ/λ^2 in SIG. The first and (NC+2)th patterns are respectively a $\vec{u}_\phi$ polarized pattern in the x=0 plane and a $\vec{u}_\theta$ polarized pattern in the y=0 plane, both computed using the admittance matrix which is the inverse of the impedance matrix. For I = 1,2,3,...NC, the (I+1)th and (I+NC+2)th patterns are respectively a $\vec{u}_\phi$ pattern in the x=0 plane and a $\vec{u}_\theta$ pattern in the y=0 plane, both computed using the same eigencurrent approximation to the admittance matrix. Program #4 has computed the first and (NC + 2)th patterns at NT points equally spaced between 0° and 360° and all the other patterns at N8 points equally spaced between 0° and 360°. The angles 0° and 360° are included in both the set of NT points and the set of N8 points. The (L4(I) + 1)th, the (L4(I) + NC + 2), the first, and the (NC+2)th

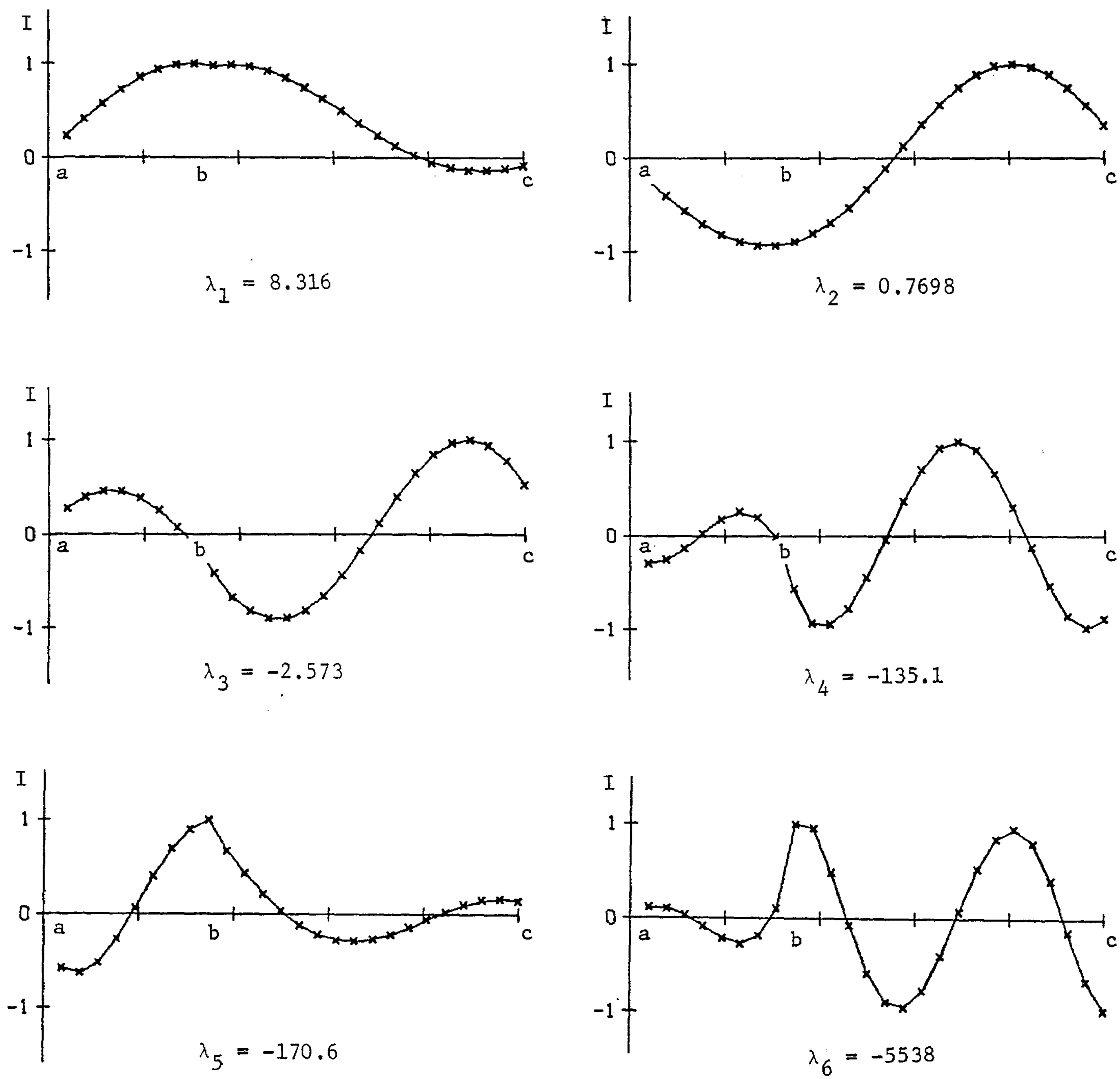

Figure 2. Calcomp plots of the lowest-order mode currents for bent wire of Fig. 1.

patterns will appear on the Ith frame of the σ/λ^2 plots.

Minimum allocations are given by

$$\text{DIMENSION TX}(\frac{NT-1}{NS} + 1),\ \text{TY}(\frac{NT-1}{NS} + 1)$$

```
X(N), X(2*NT), Y(N), Y(2*NT), SNN(NT),
CSN(NT), L4(NE1), L4(LP1), L4(NC1),
SN(NT), CS(NT), MD1(MD5), MD2(MD5), SCAL(LP1)
```

To save confusion, take the maximum of the two values of NT in read statements 88 and 89. For variables that are listed more than once, assume the maximum of the designated allocations. Since the variables FI, G, and SIG appear in the equivalence statement, they must all be allocated the same amount of space which is the maximum of N*NE, ((NT-1)/2 + 1)*LP and (NT + ((NT-1)/NS + 1)*NC)*2 words.

The variable X in DO loop 15 is used as the abscissa for plotting the eigencurrents. DO loop 96 puts tick marks on the horizontal axis. DO loop 97 puts tic marks on the vertical axis.

The index J of DO loop 20 obtains the Jth of the plots of the eigenpatterns. DO loop 52 takes advantage of the point symmetry of the eigenpatterns to extend them into the half plane corresponding to θ of (17)-(20) and to ϕ of (21)-(22) going from 180° to 360°. The vertical axis corresponds to θ or ϕ equal to zero. Point symmetry means that

$$\sigma(\pi-\theta,\ \phi+\pi) = \sigma(\theta,\phi) \tag{33}$$

Figures 3 to 7 show Calcomp plots of the six lowest-order mode gain patterns for the bent wire of Fig. 1.

DO loop 30 transfers the first and (NC+2)th patterns of σ/λ^2 to X. DO loop 28 finds that S2 is the largest X. Next, a scale factor SCL is found whereby

$$1.2 \leq S2 * SCL \leq 3. \tag{34}$$

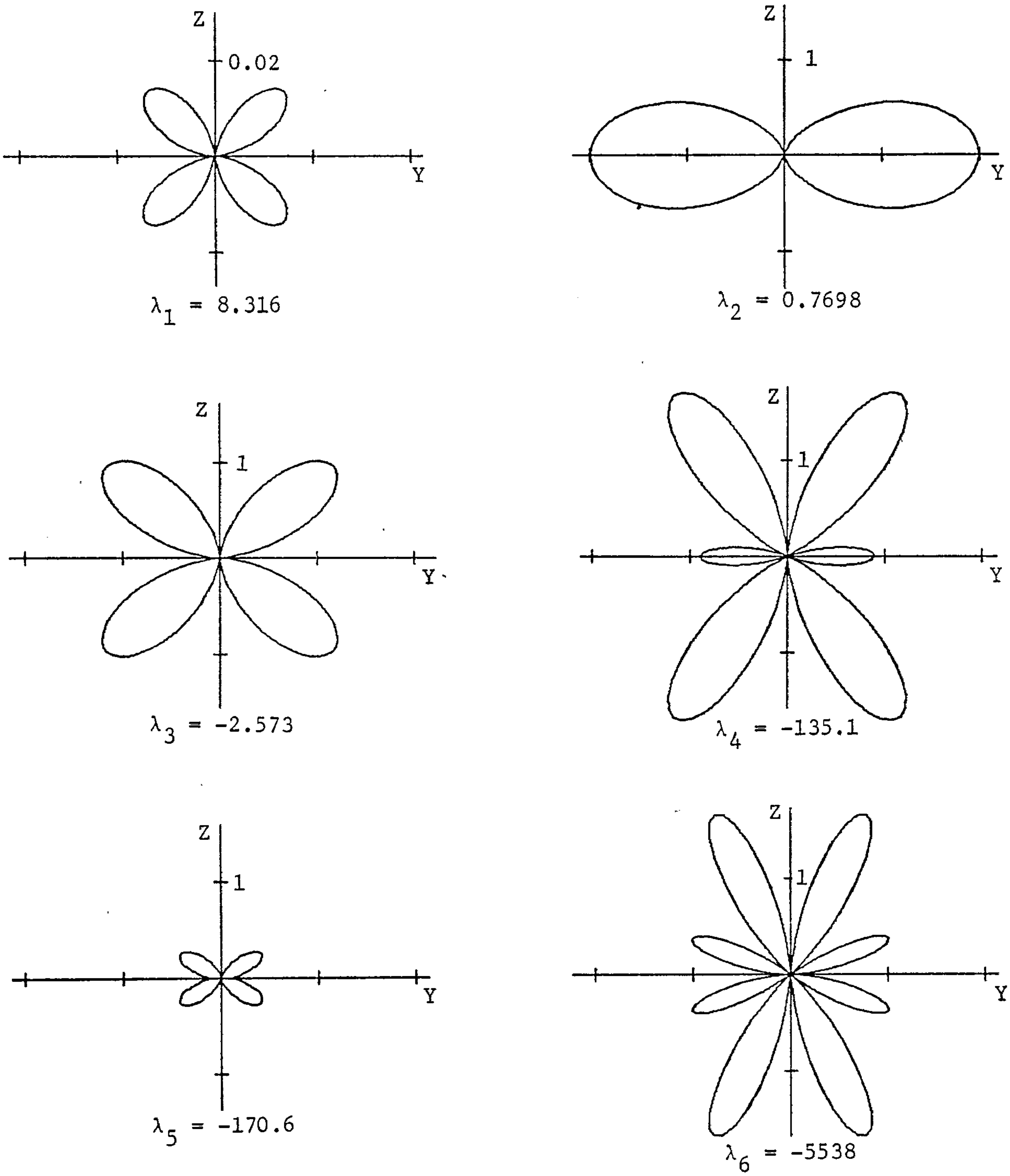

Figure 3. Calcomp plots of the lowest-order mode gain patterns $G_\theta = |E_\theta|^2$ in the x=0 plane for the bent wire of Fig. 1.

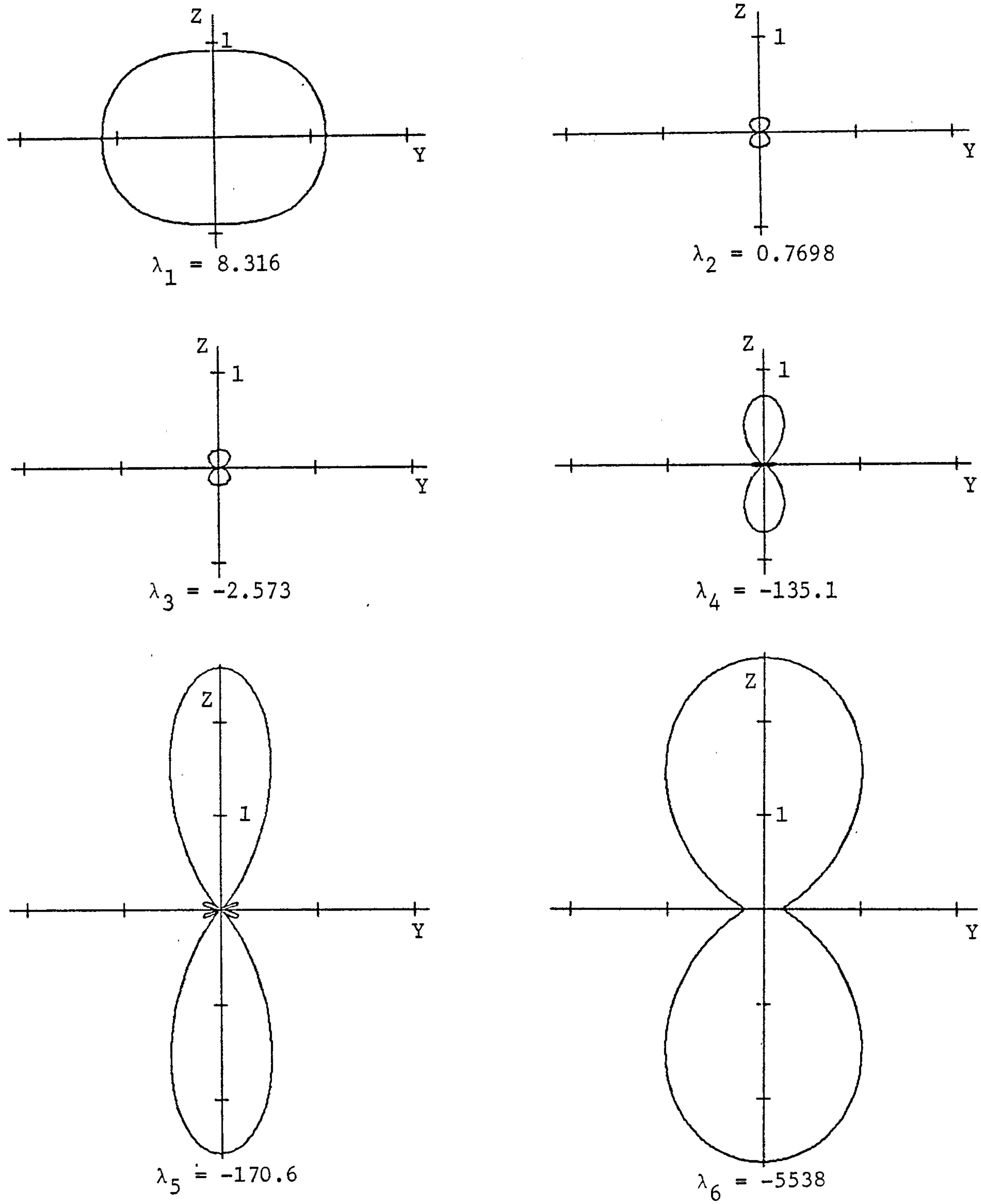

Figure 4. Calcomp plots of the lowest-order mode gain patterns $G_\phi = |E_\phi|^2$ in the x=0 plane for the bent wire of Fig. 1.

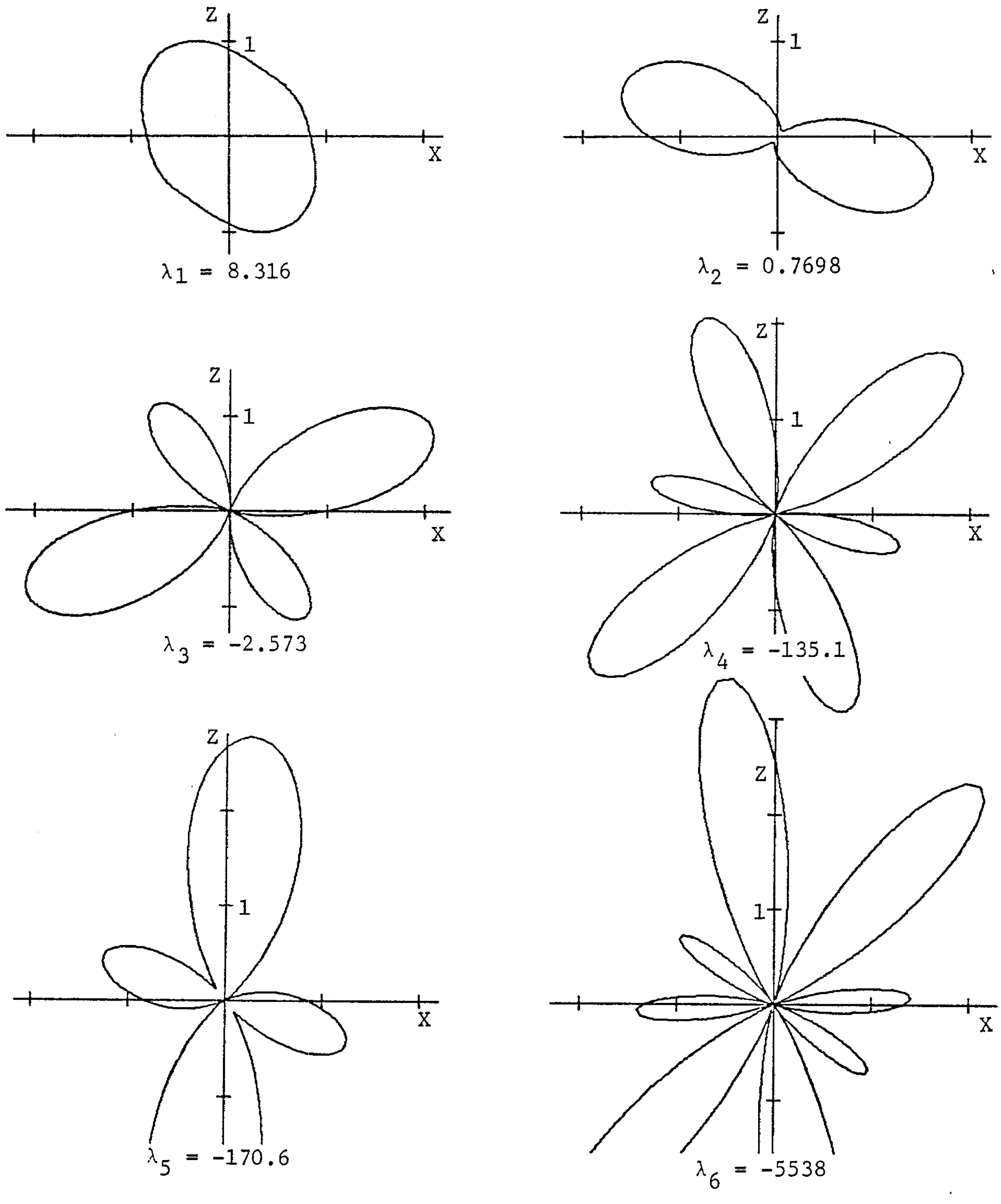

Figure 5. Calcomp plots of the lowest-order mode gain patterns $G_\theta = |E_\theta|^2$ in the y=0 plane for the bent wire of Fig. 1.

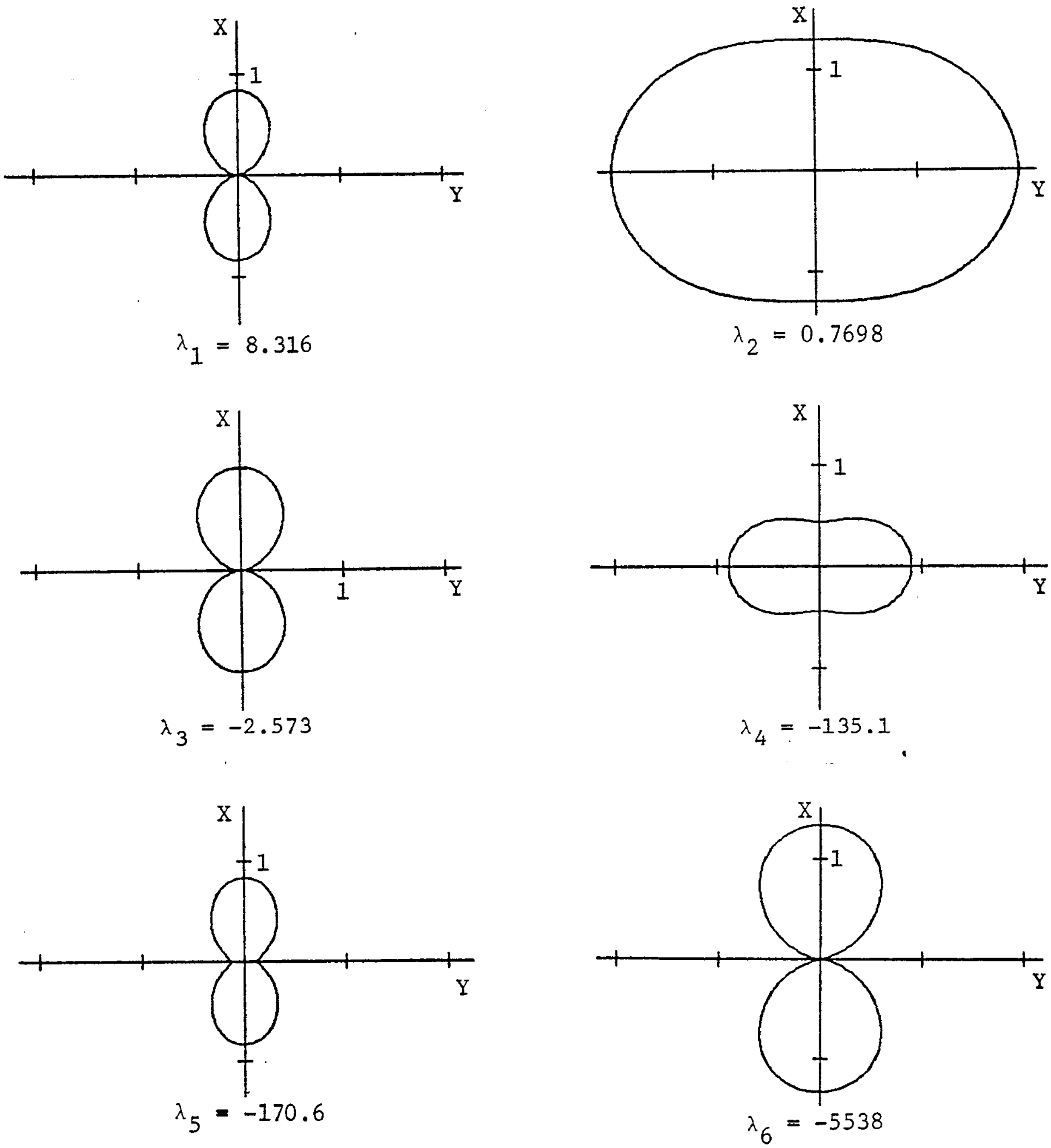

Figure 6. Calcomp plots of the lowest-order mode gain patterns $G_\theta = |E_\theta|^2$ in the z=0 plane for the bent wire of Fig. 1.

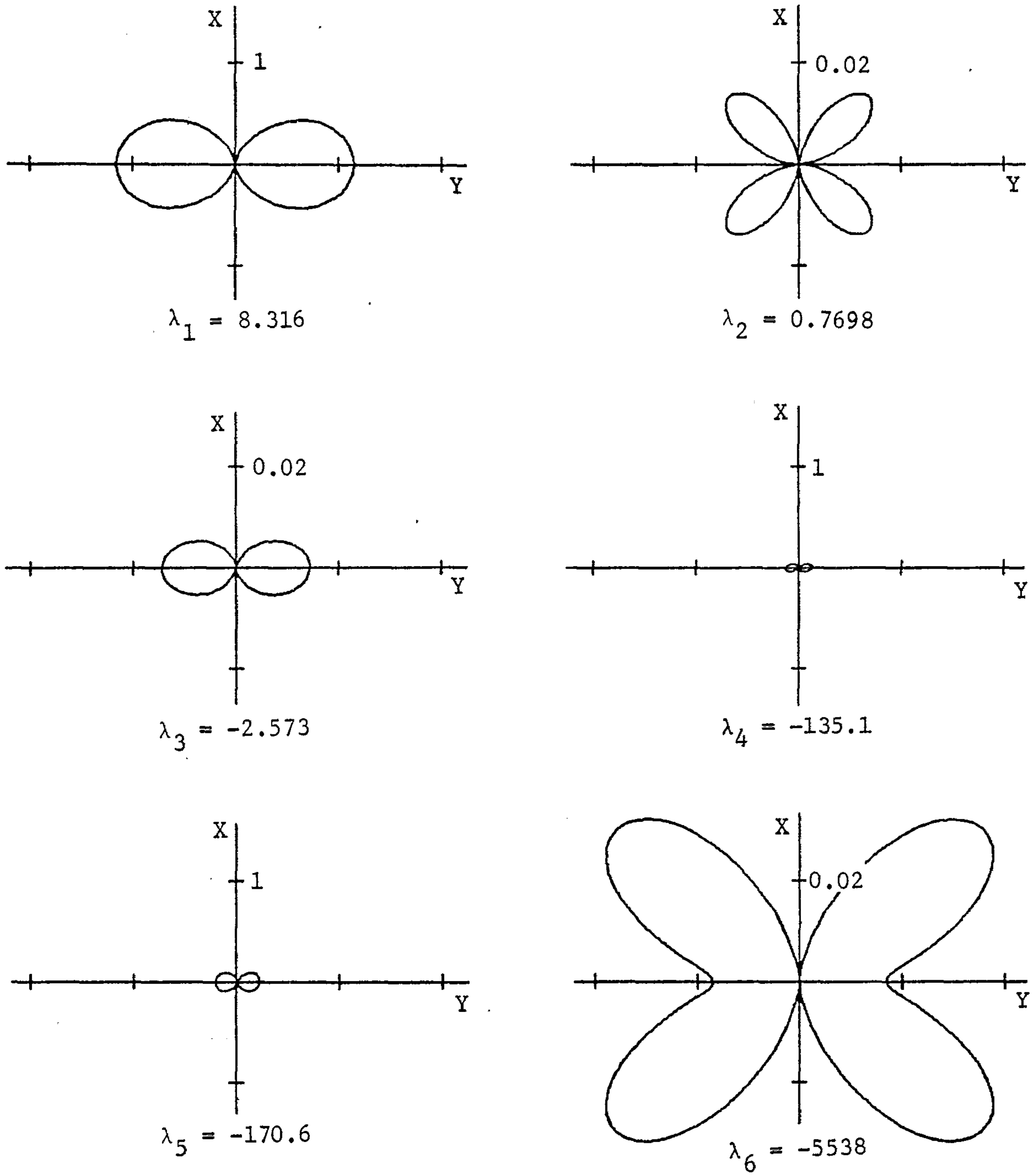

Figure 7. Calcomp plots of the lowest-order mode gain patterns $G_\phi = |E_\phi|^2$ in the z=0 plane for the bent wire of Fig. 1.

DO loop 69 obtains the horizontal and vertical coordinates X and Y for plotting the first and (NC+2)th patterns of σ/λ^2. DO loop 31 obtains the horizontal and vertical coordinates TX and TY of the (L4(J) + 1)th and the (L4(J) + NC + 2)th patterns. In DO loop 65, K4 = 1 corresponds to the (L4(J) + 1)th pattern and K4 = 2 corresponds to the (L4(J) + NC + 2)th pattern. The (L4(J) + 1)th pattern for the ϕ polarization in the x=0 plane is plotted using the symbols □ while the (L4(J) + NC + 2)th pattern for the θ polarization in the y=0 plane is plotted using the symbols x. Both the first pattern for the ϕ polarization in the x=0 plane and the (NC + 2)th pattern for the θ polarization in the y=0 plane are plotted as straight lines. Figure 8 gives Calcomp plots showing convergence of the modal solution to the matrix inversion solution (solid) as modes are added in the order of increasing $|\lambda|$.

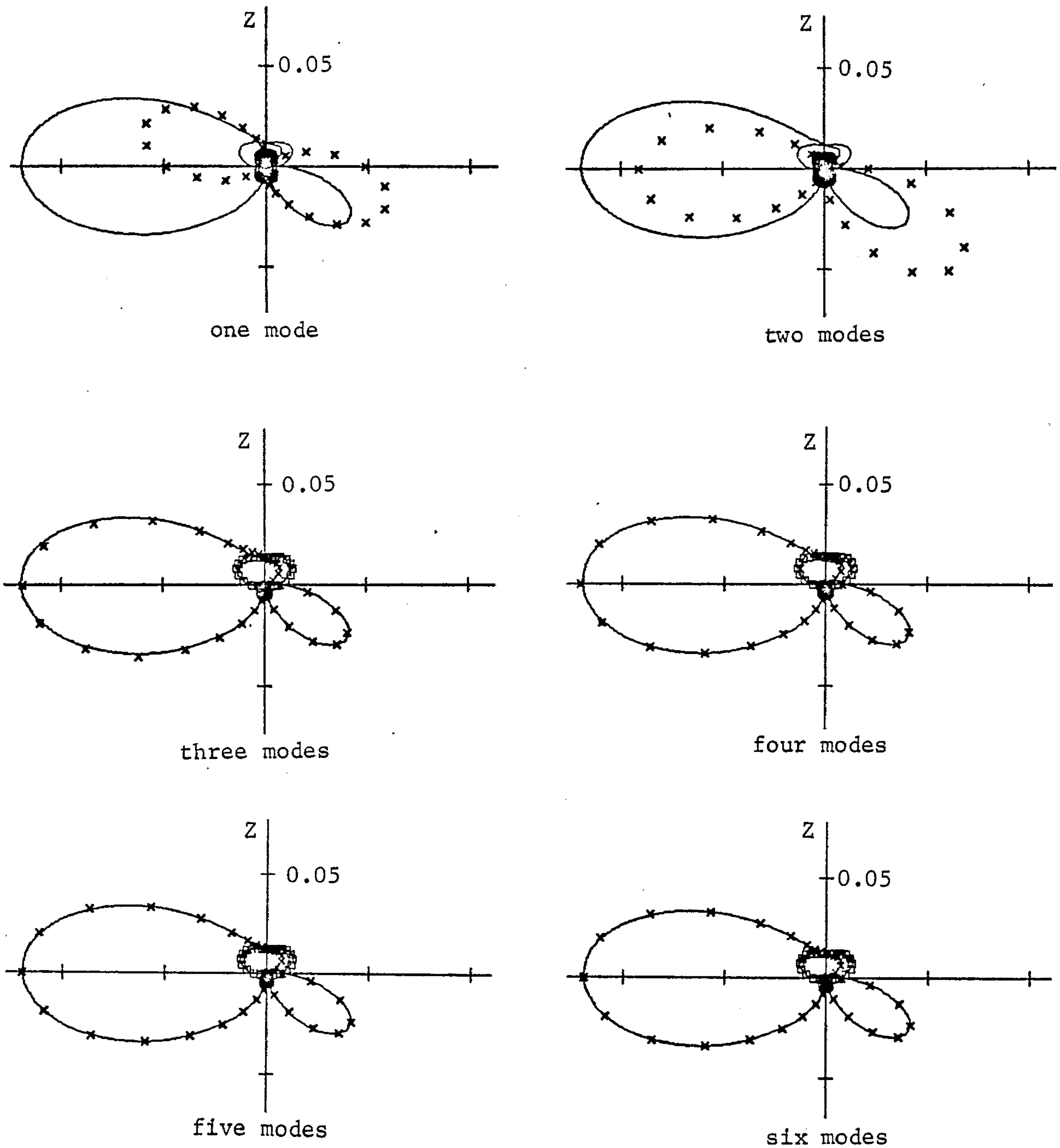

Figure 8. Calcomp plots showing convergence of the modal solution to the matrix inversion solution (solid) for bistatic radar scattering from the bent wire of Fig. 1. Incident wave is x-polarized and z-traveling. Symbols ◘ denote the modal solution for σ_ϕ/λ^2 in the x=0 plane, symbols × denote σ_θ/λ^2 in the y=0 plane.

Listing of Program #5

```
//              (0034,EE,5,2,,60),'MAUTZ,JOE',MSGLEVEL=1
// MSG H, HOLD THE PLOT FOR BLACK INDIA INK UNDER SUPERVISION D.PRIDMORE
// EXEC FORTGCLG,PARM.FORT='MAP'
//FORT.SYSIN DD *
      DIMENSION FI(5000),G(5000),SIG(5000),TX(145),TY(145),X(290),Y(290)
      DIMENSION SNN(145),CSN(145),XP(4),YP(4),AREA(400),L4(50),SN(145)
      DIMENSION CS(145),MD1(60),MD2(60),SCAL(50)
      EQUIVALENCE (FI(1),G(1),SIG(1))
      CALL PLOTS(AREA,400)
      REWIND 6
      PI=3.141593
      NTT=0
      READ(1,11) MD5
   11 FORMAT(20I3)
      WRITE(3,73) MD5
   73 FORMAT('1MD5'/1X,I3)
      READ(1,11)(MD1(I),I=1,MD5)
      WRITE(3,74)(MD1(I),I=1,MD5)
   74 FORMAT('0MD1'/(1X,20I3))
      READ(1,11)(MD2(I),I=1,MD5)
      WRITE(3,75)(MD2(I),I=1,MD5)
   75 FORMAT('0MD2'/(1X,20I3))
      DO 80 KAP=1,MD5
      J2=MD1(KAP)
      MD4=MD2(KAP)
      J1=IABS(J2)
      IF(J2) 82,83,84
   82 DO 85 J=1,J1
      BACKSPACE 6
   85 CONTINUE
      GO TO 83
   84 DO 86 J=1,J1
      READ(6)
   86 CONTINUE
   83 GO TO (87,88,89),MD4
   87 READ(1,11) N,NE,NE1
      WRITE(3,35) N,NE,NE1
   35 FORMAT('0  N NE NE1'/1X,3I3)
      READ(1,11)(L4(I),I=1,NE1)
      WRITE(3,37)(L4(I),I=1,NE1)
   37 FORMAT('0L4'/(1X,20I3))
      READ(1,13)(X(I),I=1,N)
   13 FORMAT(7E11.4)
      WRITE(3,36)(X(I),I=1,N)
   36 FORMAT('0X'/(1X,7E11.4))
      NZ=N*NE
      READ(6)(FI(I),I=1,NZ)
      WRITE(3,17)(FI(J),J=1,20)
   17 FORMAT('0FI'/(1X,10E11.4))
      DO 15 J=1,N
      X(J)=X(J)+1.
   15 CONTINUE
      XP(1)=1.
      XP(2)=6.
      YP(1)=5.
      YP(2)=5.
      XP(3)=1.
      XP(4)=1.
      YP(3)=3.
      YP(4)=7.
```

```
      DO 16 J=1,NE1
      J1=(L4(J)-1)*N
      DO 18 I=1,N
      J2=J1+I
      Y(I)=FI(J2)+5.
   18 CONTINUE
      CALL LINE(XP(1),YP(1),2,1,0,0)
      DO 96 K=1,5
      S1=7-K
      CALL SYMBOL(S1,5.,.14,13,0.,-1)
   96 CONTINUE
      CALL LINE(XP(3),YP(3),2,1,0,0)
      DO 97 K=1,5
      S1=8-K
      CALL SYMBOL(1.,S1,.14,13,90.,-1)
   97 CONTINUE
      CALL NUMBER(.64,3.93,.14,-1.,0.,-1)
      CALL NUMBER(.76,4.93,.14,0.,0.,-1)
      CALL NUMBER(.76,5.93,.14,1.,0.,-1)
      CALL LINE(X(1),Y(1),N,1,4,1)
      CALL PLOT(7.,0.,-3)
   16 CONTINUE
      GO TO 80
   88 READ(1,11) LP,LP1,NT
      WRITE(3,38) LP,LP1,NT
   38 FORMAT('0LP LP1 NT'/1X,3I3)
      READ(1,11)(L4(I),I=1,LP1)
      WRITE(3,39)(L4(I),I=1,LP1)
   39 FORMAT('0L4'/(1X,20I3))
      READ(1,13)(SCAL(I),I=1,LP1)
      WRITE(3,27)(SCAL(I),I=1,LP1)
   27 FORMAT('0SCAL'/(1X,7E11.4))
      N7=(NT-1)/2+1
      NZ=N7*LP
      READ(6)(G(I),I=1,N7)
      WRITE(3,25)(G(I),I=1,20)
   25 FORMAT('0G'/(1X,10E11.4))
      XP(1)=2.
      XP(2)=8.
      YP(1)=5.
      YP(2)=5.
      IF(NT-NTT) 1,2,1
    1 DEL=2.*PI/(NT-1)
      DO 21 J=1,NT
      S1=(J-1)*DEL
      SN(J)=SIN(S1)
      CS(J)=COS(S1)
   21 CONTINUE
    2 NTT=NT
      DO 20 J=1,LP1
      SCL=SCAL(J)
   47 S5=1./SCL
      WRITE(3,92) S5
   92 FORMAT('0ONE INCH CORRESPONDS TO A GAIN OF',E11.4)
      K1=(L4(J)-1)*N7
      DO 52 I=1,N7
      K2=K1+I
      S5=G(K2)*SCL
      X(I)=5.+S5*SN(I)
      Y(I)=5.+S5*CS(I)
```

```
      K3=I+N7-1
      X(K3)=10.-X(I)
      Y(K3)=10.-Y(I)
   52 CONTINUE
      CALL LINE(X(1),Y(1),NT,1,0,0)
   57 CALL LINE(XP(1),YP(1),2,1,0,0)
      DO 23 K=1,7
      S1=9-K
      CALL SYMBOL(S1,5.,.14,13,0.,-1)
   23 CONTINUE
      CALL LINE(YP(1),XP(1),2,1,0,0)
      DO 24 K=1,7
      S1=9-K
      CALL SYMBOL(5.,S1,.14,13,90.,-1)
   24 CONTINUE
      CALL PLOT(7.,0.,-3)
   20 CONTINUE
      GO TO 80
   89 READ(1,11) NC,NC1,NT,NS
      WRITE(3,3) NC,NC1,NT,NS
    3 FORMAT('ONC NC1 NT NS'/1X,4I3)
      READ(1,11)(L4(I),I=1,NC1)
      WRITE(3,4)(L4(I),I=1,NC1)
    4 FORMAT('OL4'/(1X,20I3))
      XP(1)=2.
      XP(2)=8.
      YP(1)=5.
      YP(2)=5.
      IF(NT-NTT) 5,6,5
    5 DEL=2.*PI/(NT-1)
      DO 7 J=1,NT
      S1=(J-1)*DEL
      SN(J)=SIN(S1)
      CS(J)=COS(S1)
    7 CONTINUE
    6 NTT=NT
      N8=(NT-1)/NS+1
      N4=NT+N8*NC
      NZ=N4*2
      READ(6)(SIG(I),I=1,NZ)
      WRITE(3,22)(SIG(I),I=1,20)
   22 FORMAT('OSIG'/(1X,10E11.4))
      DO 30 J=1,NT
      J2=J+N4
      J1=J+NT
      X(J)=SIG(J)
      X(J1)=SIG(J2)
   30 CONTINUE
      N9=NT*2
      S2=0.
      DO 28 J=1,N9
      IF(X(J).GT.S2) S2=X(J)
   28 CONTINUE
      J1=10+ALOG10(S2)
      S3=.1**(J1-10)
      S4=S2*S3
      IF(S4-1.5) 110,110,111
  110 SCL=2.*S3
      GO TO 112
  111 IF(S4-3.) 113,113,114
```

```
  113 SCL=S3
      GO TO 112
  114 IF(S4-6.) 115,115,116
  115 SCL=.5*S3
      GO TO 112
  116 SCL=.2*S3
  112 S5=1./SCL
      WRITE(3,109) S5
  109 FORMAT('OONE INCH CORRESPONDS TO',E11.4)
      DO 29 J=1,NT
      SNN(J)=SCL*SN(J)
      CSN(J)=SCL*CS(J)
   29 CONTINUE
      DO 69 I=1,NT
      Y(I)=5.+X(I)*CSN(I)
      X(I)=5.+X(I)*SNN(I)
      K2=NT+I
      Y(K2)=5.+X(K2)*CSN(I)
      X(K2)=5.+X(K2)*SNN(I)
   69 CONTINUE
      NT1=NT+1
      DO 31 J=1,NC1
      CALL LINE(XP(1),YP(1),2,1,0,0)
      DO 33 K=1,7
      S1=9-K
      CALL SYMBOL(S1,5.,.14,13,0.,-1)
   33 CONTINUE
      CALL LINE(YP(1),XP(1),2,1,0,0)
      DO 34 K=1,7
      S1=9-K
      CALL SYMBOL(5.,S1,.14,13,90.,-1)
   34 CONTINUE
      J4=NT+(L4(J)-1)*N8
      DO 65 K4=1,2
      K7=(K4-1)*4
      J1=J4+(K4-1)*N4
      J2=0
      DO 70 I=1,NT,NS
      J2=J2+1
      J3=J1+J2
      TX(J2)=5.+SIG(J3)*SNN(I)
      TY(J2)=5.+SIG(J3)*CSN(I)
   70 CONTINUE
      DO 72 K=1,J2
      CALL SYMBOL(TX(K),TY(K),.07,K7,0.,-1)
   72 CONTINUE
   65 CONTINUE
      CALL LINE(X,Y,NT,1,0,0)
      CALL LINE(X(NT1),Y(NT1),NT,1,0,0)
      CALL PLOT(7.,0.,-3)
   31 CONTINUE
   80 CONTINUE
      CALL PLOT(6.,0.,-3)
      STOP
      END
/*
//GO.FT06F001 DD DSNAME=SURC0677.ZNEW,DISP=OLD,UNIT=2314,                   X
//               VOLUME=SER=SH0005,DCB=(RECFM=V,BLKSIZE=2596,LRECL=2592)
//GO.SYSIN DD *
  3
```

```
60  0  0
 1  2  3
26  7  7
 1  2  3  4  5  6  7
0.1923E+00 0.3846E+00 0.5769E+00 0.7692E+00 0.9615E+00 0.1154E+01 0.1346E+01
0.1538E+01 0.1731E+01 0.1923E+01 0.2115E+01 0.2308E+01 0.2500E+01 0.2692E+01
0.2885E+01 0.3077E+01 0.3269E+01 0.3462E+01 0.3654E+01 0.3846E+01 0.4038E+01
0.4231E+01 0.4423E+01 0.4615E+01 0.4808E+01 0.5000E+01
42 35145
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20
21 29 30 31 32 33 34 35 36 37 38 39 40 41 42
0.5000E+02 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01
0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01
0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01
0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01 0.1000E+01
0.1000E+01 0.5000E+02 0.5000E+02 0.1000E+01 0.1000E+01 0.5000E+02 0.1000E+01
 7  7145  4
 1  2  3  4  5  6  7
/*
//
```

Sample Output

```
MD5
  1

MD1
 59

MD2
  1

 NP NT NS LS NW        BK
 55145  4  1  1 0.1396263E 00

PX
 -4.6587 -4.3999 -4.1411 -3.8823 -3.6235 -3.3646 -3.1058 -2.8470 -2.5882 -2.3294
 -2.0706 -1.8117 -1.5529 -1.2941 -1.0353 -0.7765 -0.5176 -0.2588  0.0     0.2588
  0.5176  0.7765  1.0353  1.2941  1.5529  1.8117  2.0706  2.3294  2.5882  2.8470
  3.1058  3.3646  3.6235  3.8823  4.1411  4.3999  4.6587  4.9175  5.1764  5.4352
  5.6940  5.9528  6.2117  6.4705  6.7293  6.9881  7.2469  7.5058  7.7646  8.0234
  8.2822  8.5410  8.7998  9.0587  9.3175

PY
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0     0.0
  0.0     0.0     0.0     0.0     0.0

PZ
 17.3867 16.4207 15.4548 14.4889 13.5230 12.5570 11.5911 10.6252  9.6593  8.6933
  7.7274  6.7615  5.7956  4.8296  3.8637  2.8978  1.9319  0.9659  0.0     0.9659
  1.9319  2.8978  3.8637  4.8296  5.7956  6.7615  7.7274  8.6933  9.6593 10.6252
 11.5911 12.5570 13.5230 14.4889 15.4548 16.4207 17.3867 18.3526 19.3185 20.2844
 21.2504 22.2163 23.1822 24.1481 25.1141 26.0800 27.0459 28.0118 28.9778 29.9437
 30.9096 31.8755 32.8415 33.8074 34.7733

LL
  1

RAD
 0.4500000E 00

WIRE LENGTH VARIABLE
 0.1923E 00 0.3846E 00 0.5769E 00 0.7692E 00 0.9615E 00 0.1154E 01 0.1346E 01
 0.1538E 01 0.1731E 01 0.1923E 01 0.2115E 01 0.2308E 01 0.2500E 01 0.2692E 01
 0.2885E 01 0.3077E 01 0.3269E 01 0.3462E 01 0.3654E 01 0.3846E 01 0.4038E 01
 0.4231E 01 0.4423E 01 0.4615E 01 0.4808E 01 0.5000E 01
```

NE	NC	INC	MD3	MD4
7	7	-1	0	1

AMD

0.8315513E 01 0.7697922E 02-0.2573003E 01-0.1351125E 03-0.1706178E 03
-0.5537559E 04-0.1320306E 05

LR

2 3 1 4 5 6 7

ZL

0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0	0.0	0.0	0.0	0.0
0.0	0.0	0.0				

SCATTERED FIELD AND SCATTERING CS/W2
INCIDENCE FROM θ=180.
THIS PATTERN IS IN THE PLANE X=0

Ø	REAL(EØ)	IMAG(EØ)	SØ/(LAM)**2
0.0	0.8517E-01	-0.6966E-01	0.1211E-01
10.0	0.8414E-01	-0.7264E-01	0.1236E-01
20.0	0.8027E-01	-0.8130E-01	0.1305E-01
30.0	0.7146E-01	-0.9453E-01	0.1404E-01
40.0	0.5506E-01	-0.1095E 00	0.1501E-01
50.0	0.2963E-01	-0.1208E 00	0.1548E-01
60.0	-0.2406E-02	-0.1221E 00	0.1492E-01
70.0	-0.3394E-01	-0.1089E 00	0.1300E-01
80.0	-0.5509E-01	-0.8278E-01	0.9886E-02
90.0	-0.5925E-01	-0.5284E-01	0.6303E-02
100.0	-0.4809E-01	-0.3056E-01	0.3246E-02
110.0	-0.3061E-01	-0.2272E-01	0.1453E-02
120.0	-0.1693E-01	-0.2793E-01	0.1067E-02
130.0	-0.1225E-01	-0.3934E-01	0.1698E-02
140.0	-0.1576E-01	-0.5018E-01	0.2766E-02
150.0	-0.2334E-01	-0.5709E-01	0.3805E-02
160.0	-0.3101E-01	-0.6013E-01	0.4577E-02
170.0	-0.3629E-01	-0.6091E-01	0.5026E-02
180.0	-0.3813E-01	-0.6097E-01	0.5171E-02
190.0	-0.3629E-01	-0.6091E-01	0.5026E-02
200.0	-0.3101E-01	-0.6013E-01	0.4577E-02
210.0	-0.2334E-01	-0.5709E-01	0.3805E-02
220.0	-0.1576E-01	-0.5018E-01	0.2766E-02
230.0	-0.1225E-01	-0.3934E-01	0.1698E-02
240.0	-0.1693E-01	-0.2793E-01	0.1067E-02
250.0	-0.3061E-01	-0.2272E-01	0.1453E-02
260.0	-0.4809E-01	-0.3056E-01	0.3246E-02
270.0	-0.5925E-01	-0.5284E-01	0.6303E-02
280.0	-0.5509E-01	-0.8278E-01	0.9886E-02
290.0	-0.3394E-01	-0.1089E 00	0.1300E-01
300.0	-0.2406E-02	-0.1221E 00	0.1492E-01
310.0	0.2963E-01	-0.1208E 00	0.1548E-01
320.0	0.5506E-01	-0.1095E 00	0.1501E-01
330.0	0.7146E-01	-0.9453E-01	0.1404E-01
340.0	0.8027E-01	-0.8130E-01	0.1305E-01
350.0	0.8414E-01	-0.7264E-01	0.1236E-01
360.0	0.8517E-01	-0.6966E-01	0.1211E-01

plus 11 more pages

REFERENCES

[1] R. J. Garbacz, "A Generalized Expansion for Radiated and Scattered Fields," Ph.D. dissertation, The Ohio State University, 1968.

[2] R. F. Harrington and J. R. Mautz, "Theory and Computation of Characteristic Modes for Conducting Bodies," Report AFCRL-70-0657, Contract F19628-68-C-0180 between Syracuse University and Air Force Cambridge Research Laboratories, December 1970.

[3] H. H. Chao and B. J. Strait, "Computer Programs for Radiation and Scattering by Arbitrary Configurations of Bent Wires," Report AFCRL-70-0374, Contract F19628-68-C-0180 between Syracuse University and Air Force Cambridge Research Laboratories, September 1970. pp. 13-14 and 67-70.

[4] R. F. Harrington, "Matrix Methods for Field Problems," Proc. IEEE, vol. 55, February 1967, pp. 136-149.

[5] IBM System/360 Scientific Subroutine Package (360A-CM-03X) Version III, Programmer's Manual.

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Syracuse University Electrical Engineering Department Syracuse, New York 13210	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b. GROUP

3. REPORT TITLE

COMPUTER PROGRAMS FOR CHARACTERISTIC MODES OF WIRE OBJECTS

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Scientific, Interim

5. AUTHOR(S) (First name, middle initial, last name)

Joseph R. Mautz
Roger F. Harrington

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
March 1971	71	5

8a. CONTRACT OR GRANT NO. F19628-68-C-0180 b. PROJECT, TASK, AND WORK UNIT NO. 5635-06-01 c. DOD ELEMENT 61102F d. DOD SUBELEMENT 681305	9a. ORIGINATOR'S REPORT NUMBER(S) Scientific Report No. 11 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report) AFCRL-71-0174

10. DISTRIBUTION STATEMENT

1 - This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
TECH, OTHER	Air Force Cambridge Research Laboratories (LZ) L. G. Hanscom Field Bedford, Massachusetts 01730

13. ABSTRACT

Computer programs for calculating the characteristic modes of wire objects of arbitrary shape are given. A program for computing the generalized impedance matrix of wire objects is included. It is valid for systems of N wires of arbitrary shape, using triangle functions for both expansion and testing. A program for using the characteristic modes in plane-wave scattering problems, showing convergence of the modal solution, is also given. Programs for making Calcomp plots of the characteristic currents, gain patterns and modal solutions are included. This report gives program descriptions, operating instructions, listings, and sample input-output data for each program.

UNCLASSIFIED
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Bent wire						
Characteristic currents						
Characteristic modes						
Computer programs						
Modal solutions						
Radiation from wires						
Scattering from wires						
Wire antennas						
Wire scatterers						

UNCLASSIFIED
Security Classification