Interaction Notes

Note 203

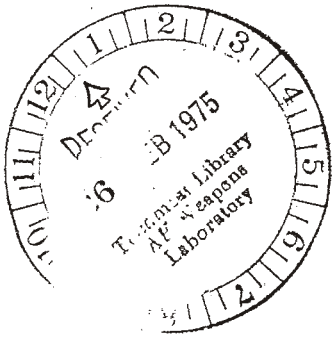
June 1974

Computer Program for Thin-Wire Structures in a Homogeneous Conducting Medium

J. H. Richmond
Ohio State University

ABSTRACT

A computer program is presented for thin-wire antennas and scatterers in a homogenous conducting medium. The analysis is performed in the real or complex frequency domain. The program handles insulated and bare wires with finite conductivity and lumped loads. The output data includes the current distribution, impedance, radiation efficiency, gain, absorption cross section, scattering cross section, echo area and the polarization scattering matrix. The program uses sinusoidal bases and Galerkin's method.



CONTENTS

I. INTRODUCTION

Reference 1 presents the electromagnetic theory for a thin-wire structure in a homogeneous conducting medium, and this report presents the corresponding computer program. The program performs a frequency-domain analysis of thin-wire antennas and scatterers. The wire configuration is a generalized polygon assembled from straight wire segments. The program has been tested extensively with simple structures (linear dipoles, V dipoles, coupled dipoles, square loops, octagonal loops, multiturn loops and coupled loops) and complicated configurations including wire-grid models of plates, spheres, cones, aircraft and ships. Although the air-earth or air-water interface is not considered, the program is applicable in many problems involving buried or submerged antennas or targets. It is useful in locating the poles of the admittance or scattering function for wire structures in the complex frequency domain.

A piecewise-sinusoidal expansion is used for the current distribution. The matrix equation $ZI = V$ is generated by enforcing reaction tests with a set of sinusoidal dipoles located in the interior region of the wire. Since the test dipoles have the same current distribution as the expansion modes, this may be regarded as an application of Galerkin's method. Rumsey's reaction concept was most helpful in this development, and therefore the formulation is known as the "sinusoidal reaction technique."

The current is assumed to vanish at the endpoints (if any) of the wire, and Kirchhoff's current law is enforced everywhere on the structure. The input data specify the frequency, wire radius, wire conductivity, the parameters of the exterior medium, coordinates of points to describe the shape and size of the wire configuration, and a list of the wire segments. If some or all of the wire segments are insulated, the radius and permittivity of the insulating sleeve are indicated.

Coordinates are required for wire endpoints, corners, junctions and terminals. For accuracy, the longest wire segment should not greatly exceed one-quarter wavelength. Longer segments should be subdivided by defining additional current-sampling points. The program automatically defines a set of N sinusoidal dipole modes on the wire structure and computes the mutual impedance matrix for these modes. The elements in the matrix are generated by numerical integration when appropriate, or from closed-form expressions in terms of exponential integrals. The computer program uses certain approximations which yield a symmetric matrix even when the wire structure has finite conductivity, lumped loads and insulating sleeves.

In antenna problems, the output data includes the current distribution, impedance, radiation efficiency, gain, patterns and near-zone field. In bistatic scattering problems, the output includes the echo

area and the complex elements of the polarization scattering matrix. In backscatter situations, the output includes also the absorption, scattering and extinction cross sections.

If the wire has finite conductivity or dielectric sleeves, it is assumed that the frequency is real. This restriction can readily be removed if the user will specify the surface impedance of the wire and the complex permittivities of the dielectric sleeves and the ambient medium appropriate for complex frequencies.

The user may make a tradeoff between accuracy and computation costs by specifying the input variable INT. A large value increases the accuracy and the cost. For most problems, the recommended value is INT = 4.

The program was run on an IBM 370/165 computer to determine the broadside backscatter for a wire-grid square plate with edge length L. With a five-by-five grid, there are 60 segments, 36 points and 84 simultaneous equations. With INT = 4, calculations were made for L/λ = 0.3, 0.4, 0.5, 0.6 and 0.7. The execution time was 100 seconds. This averages to 20 seconds for each value of L/λ. The wire structure was perfectly conducting, uninsulated and located in free space. No advantage was taken of the target symmetries.

The next section presents the thin-wire computer program, instructions for the user, typical input and output data and tables of the mutual impedance of sinusoidal dipoles. Appendicies list the computer subroutines and explain their functions.

II. THE THIN-WIRE COMPUTER PROGRAM

Fig. 1 is a Fortran listing of the thin-wire computer program. Near the beginning of this program, the DIMENSION statements reserve storage for a wire structure with up to 50 segments, 55 points and 60 dipole modes. Quantities with the same or related dimensions are grouped together on the same line or consecutive lines.

NM denotes the actual number of monopoles (segments), INM is the corresponding dimension, and the dimension for CG, VG and ZLD is twice INM. The second subscript for MD always has a dimension of 4.

N denotes the number of simultaneous linear equations and ICJ is the corresponding dimension. The dimension for C is (ICJ*ICJ + ICJ)/2.

The DO LOOP ending at statement 15 sets ISC(J) = 0 for all the segments. This indicates that the wires are bare or uninsulated. If some or all of the segments are insulated, the user may set ISC(J) = 1 for the appropriate segment numbers J.

```
      COMPLEX EP2,EP3,ETA,GAM,Y11,Z11,ZS                                0001
      COMPLEX EPPS,EPTS,ETPS,ETTS,EX,EY,EZ                              0002
      COMPLEX C(1830),CJ(60),EP(60),ET(60),EPP(60),ETT(60)              0003
      DIMENSION I1(60),I2(60),I3(60),JA(60),JB(60)                      0004
      COMPLEX CGD(50),SGD(50),CG(100),VG(100),ZLD(100)                  0005
      DIMENSION D(50),IA(50),IB(50),ISC(50),MD(50,4),ND(50)             0006
      DIMENSION X(55),Y(55),Z(55)                                       0007
      DATA PI,TP/3.14159,6.28318/                                       0008
      DATA E0,U0/8.854E-12,1.2566E-6/                                   0009
    2 FORMAT(1X,8F15.7)                                                 0010
    3 FORMAT(1X,4F15.7/)                                                0011
    4 FORMAT(1X,I15,8F14.6)                                             0012
    5 FORMAT(1H0)                                                       0013
    6 FORMAT(1X,6F15.7/)                                                0014
    7 FORMAT(8F10.5)                                                    0015
    8 FORMAT(1X,1I4,1315)                                               0016
    9 FORMAT(3X,'MAX = ',I5,3X,'MIN = ',I5,3X,'N = ',I5)                0017
      ICJ=60                                                            0018
      INM=50                                                            0019
      DO 15 J=1,INM                                                     0020
   15 ISC(J)=0                                                          0021
      READ(5,7)BM,ER2,SIG2,TD2                                          0022
      WRITE(6,2)BM,ER2,SIG2,TD2                                         0023
      READ(5,7)AM,CMM,ER3,SIG3,TD3                                      0024
      WRITE(6,2)AM,CMM,ER3,SIG3,TD3                                     0025
      READ(5,8)IBISC,IGAIN,INEAR,ISCAT,IWR,NGEN,NM,NP                   0026
      WRITE(6,8)IBISC,IGAIN,INEAR,ISCAT,IWR,NGEN,NM,NP                  0027
      READ(5,7)FMC,PHA,THA,PHI,THI,PHS,THS                              0028
      WRITE(6,2)FMC,PHA,THA,PHI,THI,PHS,THS                             0029
      DO 22 J=1,NM                                                      0030
      READ(5,8)IA(J),IB(J)                                              0031
   22 WRITE(6,8)J,IA(J),IB(J)                                           0032
      DO 40 I=1,NP                                                      0033
      READ(5,7)X(I),Y(I),Z(I)                                           0034
   40 WRITE(6,4)I,X(I),Y(I),Z(I)                                        0035
      READ(5,7)XP,YP,ZP                                                 0036
      FHZ=FMC*1.E6                                                      0037
      OMEGA=TP*FHZ                                                      0038
      IF(SIG2.LT.0.)EP2=ER2*E0*CMPLX(1.,-TD2)                           0039
      IF(TD2.LT.0.)EP2=CMPLX(ER2*E0,-SIG2/OMEGA)                        0040
      IF(SIG3.LT.0.)EP3=ER3*E0*CMPLX(1.,-TD3)                           0041
      IF(TD3.LT.0.)EP3=CMPLX(ER3*E0,-SIG3/OMEGA)                        0042
      ETA=CSQRT(U0/EP3)                                                 0043
      GAM=OMEGA*CSQRT(-U0*EP3)                                          0044
      CALL SORT(IA,IB,I1,I2,I3,JA,JB,MD,ND,NM,NP,N,MAX,MIN,ICJ,INM)     0045
      WRITE(6,5)                                                        0046
      WRITE(6,9)MAX,MIN,N                                               0047
      WRITE(6,5)                                                        0048
      IF(MAX.GT.4 .OR. MIN.LT.1 .OR. N.GT.ICJ)GO TO 800                 0049
      INT=4                                                             0050
      I12=1                                                             0051
      DO 60 J=1,NM                                                      0052
      VG(J)=(.0,.0)                                                     0053
      ZLD(J)=(.0,.0)                                                    0054
      JJ=J+NM                                                           0055
      VG(JJ)=(.0,.0)                                                    0056
   60 ZLD(JJ)=(.0,.0)                                                   0057
      IF(NGEN.GT.0)VG(NGEN)=(1.,0.)                                     0058
      CALL        SGANT(IA,IB,INM,INT,ISC,I1,I2,I3,JA,JB,MD,N,ND,NM,NP  0059
     2,AM,BM,C,CGD,CMM,D,EP2,EP3,ETA,FHZ,GAM,SGD,X,Y,Z,ZLD,ZS)          0060
      IF(N.LE.0)GO TO 800                                               0061
      IF(NGEN.LE.0)GO TO 400                                            0062
```

Fig. 1a. The thin-wire computer program.

```
      CALL GANT1(IA,IB,INM,IWR,I1,I2,I3,I12,JA,JB,MD,N,ND,NM,AM                0063
     2,C,CJ,CG,CMM,D,EFF,GAM,GG,CGD,SGD,VG,Y11,Z11,ZLD,ZS)                    0064
      WRITE(6,3)EFF,GG,Z11                                                     0065
  200 IF(INEAR.LE.0)GO TO 300                                                  0066
      CALL        GNFLD(IA,IB,INM,I1,I2,I3,MD,N,ND,NM,AM,CGD,SGD,ETA,GAM     0067
     2,CJ,D,X,Y,Z,XP,YP,ZP,EX,EY,EZ)                                          0068
      WRITE(6,3)XP,YP,ZP                                                       0069
      WRITE(6,6)EX,EY,EZ                                                       0070
  300 IF(IGAIN.LE.0)GO TO 400                                                  0071
      INC=0                                                                    0072
      PH=PHA                                                                   0073
      TH=THA                                                                   0074
      CALL        GFFLD(IA,IB,INC,INM,IWR,I1,I2,I3,I12,MD,N,ND,NM,AM         0075
     2,ACSP,ACST,C,CGD,CG,CJ,CMM,D,ECSP,ECST,EP,ET,EPP,ETT,EPPS,EPTS          0076
     3,ETPS,ETTS,GG,GPP,GTT,PH,SGD,SCSP,SCST,SPPM,SPTM,STPM,STTM,TH           0077
     4,X,Y,Z,ZLD,ZS,ETA,GAM)                                                  0078
      WRITE(6,3)PH,TH,GPP,GTT                                                  0079
  400 IF(ISCAT.LE.0)GO TO 600                                                  0080
      INC=1                                                                    0081
      PH=PHI                                                                   0082
      TH=THI                                                                   0083
      CALL        GFFLD(IA,IB,INC,INM,IWR,I1,I2,I3,I12,MD,N,ND,NM,AM         0084
     2,ACSP,ACST,C,CGD,CG,CJ,CMM,D,ECSP,ECST,EP,ET,EPP,ETT,EPPS,EPTS          0085
     3,ETPS,ETTS,GG,GPP,GTT,PH,SGD,SCSP,SCST,SPPM,SPTM,STPM,STTM,TH           0086
     4,X,Y,Z,ZLD,ZS,ETA,GAM)                                                  0087
      WRITE(6,6)PH,TH,SPPM,SPTM,STPM,STTM                                      0088
      WRITE(6,6)ACSP,ACST,ECSP,ECST,SCSP,SCST                                  0089
  500 IF(IBISC.LE.0)GO TO 600                                                  0090
      INC=2                                                                    0091
      PH=PHS                                                                   0092
      TH=THS                                                                   0093
      CALL        GFFLD(IA,IB,INC,INM,IWR,I1,I2,I3,I12,MD,N,ND,NM,AM         0094
     2,ACSP,ACST,C,CGD,CG,CJ,CMM,D,ECSP,ECST,EP,ET,EPP,ETT,EPPS,EPTS          0095
     3,ETPS,ETTS,GG,GPP,GTT,PH,SGD,SCSP,SCST,SPPM,SPTM,STPM,STTM,TH           0096
     4,X,Y,Z,ZLD,ZS,ETA,GAM)                                                  0097
      WRITE(6,6)PH,TH,SPPM,SPTM,STPM,STTM                                      0098
  600 CONTINUE                                                                 0099
  800 CALL EXIT                                                                0100
      END
```

Fig. 1b. The thin-wire computer program.

The first READ statement inputs the following parameters for the dielectric insulation:

BM	outer radius in meters
ER2	dielectric constant relative to free space
SIG2	conductivity in mhos per meter
TD2	loss tangent

The program will use SIG2 or TD2 but not both. The user determines which one will be used by assigning the other a negative value. For an uninsulated wire structure, the program will not use any of the data from the first READ statement.

The second READ statement inputs the following parameters for the wire and the exterior medium:

AM	wire radius in meters
CMM	wire conductivity in megamhos per meter
ER3	dielectric constant relative to free space
SIG3	conductivity in mhos per meter
TD3	loss tangent

The parameters ER3, SIG3 and TD3 are those of the homogeneous ambient medium. Again, the program will use SIG3 or TD3 but not both.

The third READ statement inputs the following data:

IBISC	indicator for bistatic scattering calculations
IGAIN	indicator for antenna gain calculations
INEAR	indicator for near-zone field calculations
ISCAT	indicator for backscatter calculations
IWR	indicator for writeout of current distributions
NGEN	indicator for antenna calculations
NM	number of monopoles (segments)
NP	number of points

For each indicator, a positive value means the calculation or writeout is desired while a zero or negative value means it is not desired.

The fourth READ statement inputs the following data:

FMC	frequency in megahertz
PHA,THA	far-field angle for antenna gain
PHI,THI	incidence angle for plane-wave scattering
PHS,THS	scattering angle for bistatic scattering

The above angles are given in degrees, and they denote values of the angular coordinates in the spherical system (r,θ,ϕ) widely used in antenna and scattering literature.

The fifth READ statement (in the DO LOOP ending with statement 22) inputs the endpoints IA(J) and IB(J) of segment J. Thus, IA and IB are the index numbers of the two points which are joined by segment J.

The sixth READ statement (in the DO LOOP ending with statement 40) inputs the coordinates X(I), Y(I) and Z(I) of point I in meters. The seventh and last READ statement inputs the coordinates XP, YP and ZP (in meters) of the observation point for near-zone field calculations.

Some of the quantities used in the program are defined as follows:

FHZ	frequency in Hertz
OMEGA	angular frequency
EP2	complex permittivity of insulation
EP3	complex permittivity of ambient medium
ETA	intrinsic impedance of ambient medium
GAM	intrinsic propagation constant of ambient medium
ZS	surface impedance of wire

For an uninsulated wire with perfect conductivity, one may specify complex values for ETA and GAM and delete the following input data and calculations: BM, ER2, SIG2, TD2, ER3, SIG3, TD3, FMC, FHZ, OEMGA, EP2 and EP3.

After reading the input data, the program calls subroutine SORT. This subroutine defines a set of dipole modes on the wire structure. N denotes the total number of dipole modes, the number of simultaneous linear equations, and the size of the impedance matrix Z_{ij}. Since this matrix is symmetric, only the upper-right triangular portion (including the entire principal diagonal) is calculated and stored in C(K). SORT calculates N, but the user may predict N as follows to reserve adequate storage. If m wire segments intersect at a point, this point is defined as a junction of order m and degree n = m - 1. There will be n dipole modes with terminals at this junction. N is determined by summing the degrees of all the junctions. For an example, an endpoint of a dipole is a junction of order m = 1 and degree n = 0. The vertex of a V dipole is a junction of order 2 and degree 1. NP denotes the number of points on the wire structure, and each of these points is considered to be a junction.

Mode I is a two-segment V dipole with a sinusoidal current distributed over the intersecting segments JA(I) and JB(I). The dipole has endpoints I1(I) and I3(I) and terminals at I2(I). The reference direction for positive current on dipole I is from I1 to I2 to I3.

A wire segment may be shared by as many as four dipole modes, or as few as one. In the output of subroutine SORT, ND(J) denotes the number of dipoles sharing segment J. The extreme values of ND(J) are MAX and MIN. If MIN is less than one, the wire structure has an unconnected segment and the computation is aborted. (An isolated wire

must have at least two segments and three points.) If N exceeds ICJ, the dimensions are inadequate and the run is aborted.

INT specifies the number of intervals for calculating the elements in the impedance matrix with Simpson's-rule integration. A large value for INT improves the accuracy at the expense of greater execution time. For most problems a suitable combination of speed and accuracy is obtained with INT = 4. A larger value is recommended if one wire passes close to another as in the helix or the multiturn loop. If in doubt, one may set INT = 0 to choose the rigorous closed-form impedance expressions in terms of exponential integrals.

The DO LOOP ending with statement 60 sets all the lumped load impedances and generator voltages to zero. If the wire structure has lumped loads, one may insert a READ command after statement 60 to input a list of complex load impedances ZLD(J). For a wire antenna with just one generator, the program inserts a unit voltage generator with VG(NGEN) = (1.,0.). If the antenna or array has several generators, one may insert a READ command after statement 60 to input a list of complex voltages VG(J).

Generators or lumped loads may be inserted at either end or both ends of segment J. First consider a load impedance inserted in the middle of segment J. Now slide the load along the segment and let it approach the endpoint IA(J). This load is represented by ZLD(J). Next insert another load in segment J and slide it to approach the endpoint IB(J). This load is designated ZLD(JJ) where JJ = J + NM. The same convention is employed for the voltage generators VG(J) and VG(JJ). A generator voltage VG(J) is considered positive if it tends to force a current flow in the direction from IA(J) to IB(J).

Subroutine SGANT calculates the elements of the impedance matrix Z_{ij} and stores them in C(K) where K = (I-1)*N - (I*I - I)/2 + J. This subroutine will set N = 0 and the run will abort if the wire radius is zero or negative, the shortest segment length is less than the wire diameter, the wire radius is electrically large, or the longest segment is too long.

Subroutine GANT1 considers the thin-wire structure as an antenna and solves for the current distribution CG(J), radiation efficiency EFF, time-average power input GG and complex power input Y11. In the current distribution, CG(J) is the current on segment J as one approaches the endpoint IA(J) and CG(JJ) is the current at the other end IB(J). The reference direction for positive current is from IA to IB. Thus, the conventions are the same for the branch currents CG and the branch voltages VG.

If the antenna has only one voltage generator with VG(NGEN) = (1.,0.), then Y11 is the antenna admittance and Z11 is the impedance.

The radiation efficiency EFF is calculated from the time-average power input to the antenna and the time-average power dissipated in the wire and the lumped loads. If the antenna is insulated, the power dissipated in the insulation is neglected. If the wire has perfect conductivity and the loads are purely reactive, the calculated efficiency will be 100 per cent.

The near-field subroutine GNFLD calculates the electric field intensity (EX,EY,EZ) at the observation point (XP,YP,ZP). In the calling parameters, CJ denotes the current distribution on the wire. (The loop currents are stored in CJ(I) and the branch currents in CG(J)). Thus, the currents must be calculated before GNFLD is called. Fig. 1 illustrates the use of GNFLD to calculate the near-zone field in an antenna problem. This subroutine can be called again just above statement 500 to calculate the near-zone scattered field for a wire target. In the calling parameters, CJ is replaced with EP or ET to obtain the near-zone field with a phi-polarized or theta-polarized incident plane wave. Reference 1 describes the more sophisticated techniques required when the observation point is extremely close to the wire structure.

The far-field subroutine GFFLD calculates antenna gain if INC = 0, backscattering if INC = 1, and bistatic scattering if INC = 2. If INC = 0, PH and TH denote the spherical coordinates ϕ and θ of the distant observation point and the output from GFFLD is defined as follows. EPPS and ETTS denote the phi-polarized and theta-polarized components of the electric field intensity. For example,

$$(1) \qquad EPPS = r\, e^{\gamma r} E_{\phi}$$

where r is the distance from the origin to the observation point. GPP and GTT denote the power gains associated with the phi and theta polarizations. Appendix 14 defines GPP and GTT more precisely.

If INC = 1, PH and TH denote the incidence angles ϕ_i and θ_i. These are also the spherical coordinates of the distant source.[1] In this backscattering situation, the output data from GFFLD are defined as follows:

ACSP,ACST	absorption cross sections for ϕ and θ polarizations
ECSP,ECST	extinction cross sections for ϕ and θ polarizations
EP,ET	loop currents induced by ϕ and θ polarized waves
EPPS	scattered electric field $E_{\phi\phi}$
EPTS	scattered electric field $E_{\phi\theta}$
ETPS	scattered electric field $E_{\theta\phi}$
ETTS	scattered electric field $E_{\theta\theta}$
SCSP,SCST	scattering cross sections for ϕ and θ polarizations
SPPM	echo area $\sigma_{\phi\phi}$

SPTM	echo area $\sigma_{\phi\theta}$
STPM	echo area $\sigma_{\theta\phi}$
STTM	echo area $\sigma_{\theta\theta}$

The echo areas are given in square meters. For the doubly-subscripted quantities such as $E_{\phi\phi}$ and $\sigma_{\phi\phi}$, the first and second subscripts specify the polarizations of the incident and scattered waves, respectively. The complex numbers EPPS, EPTS, ETPS and ETTS are the elements of the polarization scattering matrix.

If INC = 2, PH and TH denote the scattering angles ϕ_s and θ_s. These are the spherical coordinates of the distant observer. In this bistatic scattering situation, the only outputs from GFFLD are the polarization scattering matrix and the echo areas.

To obtain antenna patterns, backscattering patterns or bistatic patterns, one may insert DO LOOPS in the program to increment the angles PH and TH. The DO LOOP will begin just above the call to GFFLD and terminate just below this call. To obtain the near-zone field distribution along a given probing path, one may insert a DO LOOP beginning just above the call to GNFLD and terminating just below this call.

When the calculations have been completed for one problem, one may GO TO a point just above CALL GANT1 if only the generator voltages are to be changed. One may GO TO a point just below CALL SORT if there is a change in the wire radius or conductivity, the insulation, ambient medium, frequency, load impedances or the coordinates (X,Y,Z). If there is a change in NM, NP, IA or IB, one should GO TO a point above CALL SORT.

Consider an array of three center-fed dipoles, and suppose we desire the 3 x 3 admittance matrix for the array. Let each dipole be divided into four segments with segments 1 through 4 on dipole 1, 5 through 8 on dipole 2 and 9 through 12 on dipole 3. The three-port admittance matrix can be obtained by inserting a DO LOOP beginning just above CALL GANT1 and terminating just below this call. GANT1 will be called three times with all the voltages VG set to zero except for a single one-volt generator. On the first, second and third calls, let NGEN = 3, 7 and 11 to represent a generator at the center of dipole 1, 2 and 3, respectively. After the first call, set Y11 = CG(3), Y12 = CG(7) and Y13 = CG(11). Set Y22 = CG(7) and Y23 = CG(11) after the second call and Y33 = CG(11) after the third call.

For extremely small antennas, quasi-static or double-precision subroutines are required.

The wire radius must exceed zero, but there is no difficulty with small radii. If the radius exceeds 0.007λ, the thin-wire assumptions are questionable and the accuracy and convergence deteriorate. The length ratio of the longest and shortest segments should not exceed 100. It is

assumed that the wire length exceeds the wire diameter by a factor of at least 30. We are not aware of any lower limit on the segment length, however.

If a wire is bent sharply to form a small acute angle (less than 30 degrees), the thin-wire model is questionable. It is assumed that the wire conductivity greatly exceeds the conductivity of the ambient medium. For insulated wires, the dielectric layer is assumed to be electrically thin.

For each thin-wire problem, calculations should be repeated several times with the wire divided progressively into shorter segments. There is no assurance of accuracy until the output data converge. For a moderately thick wire (with radius $a = 0.007\ \lambda$ or larger), the susceptance may diverge with the delta-gap model. This difficulty may be alleviated or eliminated with the magnetic-frill model and the techniques of Imbriale and Ingerson [2].

Tables 1, 2 and 3 list input and output data for three simple examples of uninsulated wire structures. Each table includes a sketch of the wire configuration with labels to indicate the numbering system for the points and segments. In these examples there are no lumped loads.

In the sinusoidal-reaction formulation, a basic function is the mutual impedance between two sinusoidal filamentary electric dipoles. One dipole is a test source located on the axis of the wire structure, and the other is an expansion mode on the wire surface. In view of the importance of this mutual impedance, short tables are presented next for a few simple cases. The data can be reproduced with the program in Fig. 1 with appropriate input data for uninsulated wires with perfect conductivity and no lumped loads in free space. The data were obtained with the closed-form expressions (INT = 0) by writing out the quantities C(K) just below the call to subroutine SGANT. Double precision was used for these calculations.

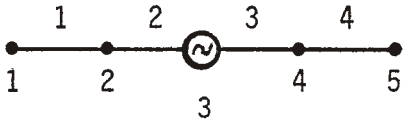
Table 4 lists the self impedance of a two-segment sinusoidal V dipole with radius $a = 0.001\ \lambda$. Subroutine SGANT calculates this quantity by setting up one filamentary dipole on the wire axis and another identical dipole on the wire surface. These dipoles lie in parallel planes separated by a distance equal to the wire radius.

In Table 5, dipoles 1 and 2 have terminals at vertices 1 and 2, respectively, and they share the middle segment. Again these dipoles lie in parallel planes separated by a distance equal to the wire radius. For a one-turn planar polygon wire loop, subroutine SGANT would generate the data in Table 4 for the diagonal elements Z_{ij} and the data in Table 5 for the next elements.

TABLE 1
Input and Output Data for Straight Wire

Input Data

0.002	2.56	-1.0	0.0005				
0.001	1.00	1.0	-1.0	0.0			
1	1	1	1	0	3	4	5
300.	0.	90.	0.	90.	45.	45.	
1	2						
2	3						
3	4						
4	5						
0.	0.	-0.250					
0.	0.	-0.125					
0.	0.	0.					
0.	0.	0.125					
0.	0.	0.250					
1.	1.	1.					



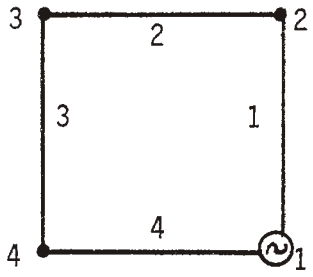
Output Data

98.18	0.0095	82.97	43.26		
-.091	0.080	-0.091	0.080	0.224	-0.096
0.0	90.0	0.0	1.615		
0.0	90.0	0.0	0.0	0.0	0.608
0.0	0.0069	0.0	0.377	0.0	0.370
45.0	45.0	0.0	0.0	0.0	0.239

TABLE 2
Input and Output Data for Square Loop

Input Data

0.002	2.56	-1.0	0.0005				
0.001	1.0	1.0	-1.0	0.0			
1	1	1	1	0	1	4	4
300.	0.0	90.0	0.0	90.0	45.	45.	
1	2						
2	3						
3	4						
4	1						
0.05	-0.05	0.0					
0.05	0.05	0.0					
-0.05	0.05	0.0					
-0.05	-0.05	0.0					
1.0	1.0	1.0					



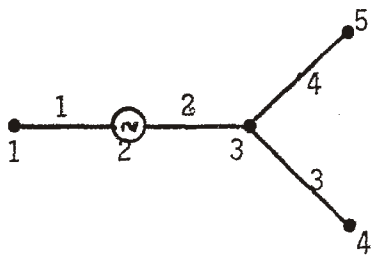
Output Data

73.10	.243E-4	62.94	1609.8		
-.0078	.0027	.0057	.0029	-.0010	-.0056
0.0	90.0	.8066	.0		
0.0	90.0	.0002	.0	.0	.0
.126E-4	0.0	.936E-4	.0	.810E-4	.0
45.0	45.0	.106E-3	.265E-4	.0	.0

TABLE 3
Input and Output Data for Y Antenna

Input Data

```
0.002       2.56        -1.0        0.0005
0.001       1.0          1.0       -1.0          0.0
1           1            1          1            0          2     4     5
300.        0.0         90.0        0.0         90.0       45.   45.
1           2
2           3
3           4
3           5
0.0        -0.30        0.0
0.0        -0.15        0.0
0.0         0.0         0.0
0.1         0.1         0.0
-0.1        0.1         0.0
1.0         1.0         1.0
```



Output Data

```
97.88       0.013       75.53       -0.572
-.124       0.081        0.260      -0.064      -0.126      0.070
0.0        90.0          1.535       0.0
0.0        90.0          0.748       0.0          0.0        0.0
0.0103      0.0          0.487       0.0          0.477      0.0
45.0       45.0          0.360       0.170        0.0        0.0
```

TABLE 4

Self Impedance of Two-Segment V Dipole Shown in Fig. 2
Radius: a = 0.001λ

ψ	h = 0.10λ	h = 0.15λ	h = 0.20λ	h = 0.25λ
30°	0.59 - j 481	1.4 - j 314	3.1 - j 186	6.1 - j 61
60	2.15 - j 547	5.3 - j 337	11.0 - j 177	21.3 - j 21
90	4.22 - j 572	10.4 - j 340	21.1 - j 163	40.0 + j 9
120	6.31 - j 583	15.3 - j 338	30.9 - j 151	57.7 + j 28
150	7.81 - j 587	18.9 - j 335	37.7 - j 144	69.3 + j 39
180	8.33 - j 589	20.1 - j 335	39.9 - j 142	73.1 + j 42

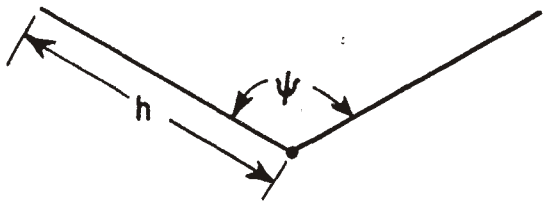


Fig. 2. Symmetric center-fed V dipole.

TABLE 5

Mutual Impedance Between Overlapping V Dipoles in Fig. 3
Radius: a = 0.001λ

ψ	h = 0.10λ	h = 0.15λ	h = 0.20λ	h = 0.25λ
60°	-0.96 + j 338	-2.08 + j 285	-3.45 + j 275	- 4.8 + j 298
90	0.19 + j 322	1.03 + j 276	3.57 + j 271	10.1 + j 297
120	3.29 + j 336	8.40 + j 290	17.86 + j 285	35.3 + j 309
150	6.61 + j 346	15.61 + j 299	30.00 + j 291	52.9 + j 309
180	8.01 + j 349	18.47 + j 301	34.35 + j 292	58.2 + j 308

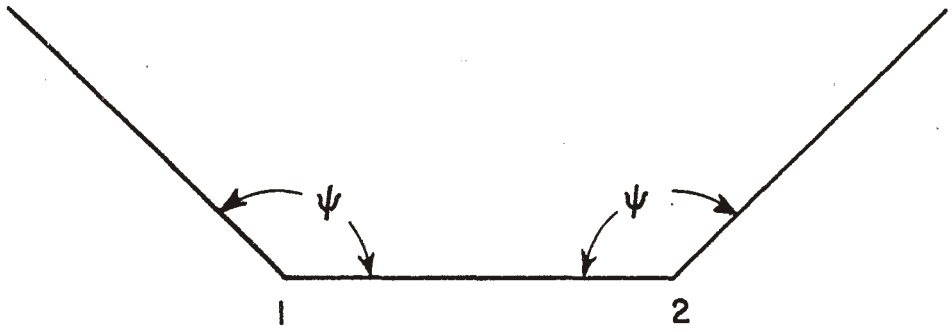


Fig. 3. Overlapping V dipoles share the middle segment.

Tables 6, 7, and 8 list the mutual impedance for other configurations. In all these tables, the data apply to two-segment center-fed sinusoidal dipoles with identical segment lengths h.

TABLE 6
Mutual Impedance Between Overlapping V Dipoles in Fig. 4
Radius: a = 0.001λ

α	h = 0.10λ	h = 0.15λ	h = 0.20λ	h = 0.25λ
30°	6.74 - j 314	16.24 - j 167	32.17 - j 56	58.7 + j 49.6
60	3.16 - j 291	7.68 - j 169	15.47 - j 76	28.8 + j 14.2
90	0.06 - j 278	0.31 - j 172	1.15 - j 92	3.5 - j 12.2
120	-1.01 - j 256	-2.39 - j 168	-4.47 - j 101	-7.6 - j 35.5
150	-0.48 - j 207	-1.20 - j 146	-2.40 - j 98	-4.5 - j 50.7

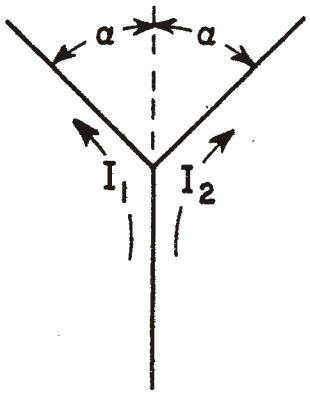


Fig. 4. Overlapping V dipoles share the bottom segment in a planar Y configuration.

TABLE 7
Mutual Impedance Between the Coplanar-Skew Linear Dipoles in Fig. 5
Displacement: $d = \lambda$

θ	$h = 0.10\lambda$	$h = 0.15\lambda$	$h = 0.20\lambda$	$h = 0.25\lambda$
0°	0.337 + j 1.952	0.880 + j 4.759	1.932 + j 9.547	4.011 + j 17.7
15	0.322 + j 1.884	0.831 + j 4.585	1.799 + j 9.180	3.671 + j 17.0
30	0.281 + j 1.684	0.700 + j 4.082	1.448 + j 8.128	2.800 + j 15.0
45	0.220 + j 1.369	0.521 + j 3.301	1.000 + j 6.519	1.745 + j 11.9
60	0.149 + j 0.964	0.333 + j 2.310	0.579 + j 4.524	0.860 + j 8.1
75	0.075 + j 0.497	0.159 + j 1.187	0.252 + j 2.308	0.305 + j 4.1
90	0.0 + j 0.0	0.0 + j 0.0	0.0 + j 0.0	0.0 + j 0.0

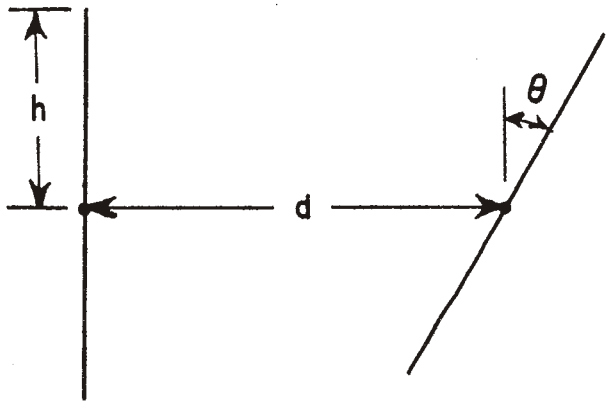


Fig. 5. Center-fed coplanar-skew linear dipoles.

TABLE 8
Mutual Impedance Between the Nonplanar-Skew Linear Dipoles in Fig. 6
Displacement: $d = \lambda$

ϕ	$h = 0.10\lambda$	$h = 0.15\lambda$	$h = 0.20\lambda$	$h = 0.25\lambda$
0°	0.337 + j 1.952	0.880 + j 4.759	1.932 + j 9.547	4.011 + j 17.74
15	0.326 + j 1.886	0.850 + j 4.596	1.867 + j 9.222	3.877 + j 17.14
30	0.292 + j 1.691	0.762 + j 4.121	1.675 + j 8.269	3.482 + j 15.37
45	0.238 + j 1.380	0.622 + j 3.365	1.369 + j 6.752	2.850 + j 12.55
60	0.169 + j 0.976	0.440 + j 2.380	0.969 + j 4.775	2.020 + j 8.88
75	0.087 + j 0.505	0.228 + j 1.232	0.502 + j 2.472	1.047 + j 4.60
90	0.0 + j 0.0	0.0 + j 0.0	0.0 + j 0.0	0.0 + j 0.0

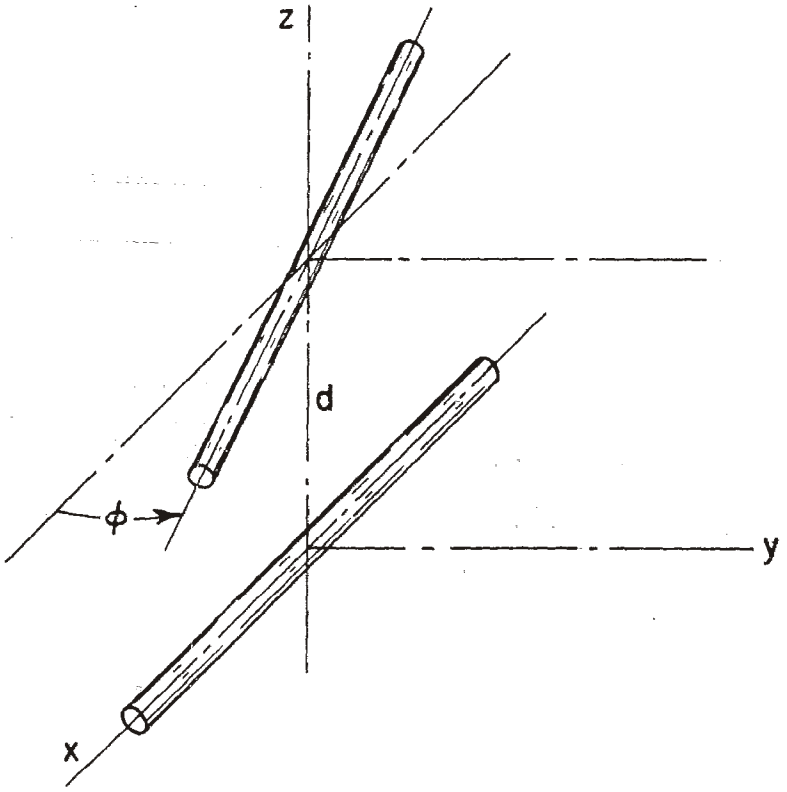


Fig. 6. Center-fed nonplanar-skew linear dipoles.

III. SUMMARY

This report presents the sinusoidal-reaction computer program for thin-wire antennas and scatterers, instructions for the user, typical input and output data and mutual-impedance tables for sinusoidal dipoles. Appendices list the computer subroutines and explain their functions.

REFERENCES

1. Richmond, J. H., "Radiation and Scattering by Thin-Wire Structures in the Complex Frequency Domain", Interaction Note 202, May 1974.

2. Imbriale, W.A., and Ingerson, P.G., "On numerical convergence of moment solutions of moderately thick wire antennas using sinusoidal basis functions," IEEE Trans., Vol. AP-21, May 1973, pp. 363-366.

3. Abramowitz, M., and Stegun, I.A., "Handbook of mathematical functions with formulas, graphs, and mathematical tables," National Bureau of Standards, Applied Mathematics Series AMS-55, 1964, Chapter 5.

4. Faddeev, D.K., and Faddeeva, V.N., Computational Methods of Linear Algebra, W. H. Freeman and Company, San Francisco, 1963, pp. 144-147.

APPENDIX 1. Subroutine SORT

Subroutine SORT, listed in Fig. 7, defines a set of dipole mode currents on the wire structure. The input data IA, IB, NM, NP, ICJ and INM have been defined already. The output data are defined as follows:

N	total number of dipole modes
I1(I)	endpoint of dipole I
I2(I)	terminal point of dipole I
I3(I)	endpoint of dipole I
JA(I)	first segment of dipole I
JB(I)	second segment of dipole I
MD(J,K)	list of dipoles sharing segment J
ND(J)	total number of dipoles sharing segment J
MAX,MIN	extreme values of ND(J)

At completion of the DO LOOP ending with statement 20, NJK denotes the number of segments intersecting at point K, and JSP is a list of these segments. In the DO LOOP ending with statement 22, the computer sets up the appropriate number MOD of dipoles modes with terminals at poink K.

APPENDIX 2. Subroutine SGANT

Subroutine SGANT, listed in Fig. 8, calculates the mutual impedances Z_{ij} and stores them in C(K). The input data for SGANT have been defined already. The output data are defined as follows:

C(K)	open-circuit impedance matrix
CGD(J)	cosh γd for segment J
SGD(J)	sinh γd for segment J
D(J)	length of segment J
ZS	surface impedance of the wire

The surface impedance is calculated just above statement 12. B01 denotes J_0/J_1 where J_0 and J_1 are the Bessel functions of order zero and one with complex argument ZARG. It is assumed that all the wire segments have the same radius, conductivity and surface impedance.

In the DO LOOP ending with statement 20, SGANT calculates the segment lengths D(J). DMIN and DMAX denote the lengths of the shortest and longest segments. If the wire radius or the segment lengths are clearly beyond the range of thin-wire theory, N is set to zero at statement 25 followed by RETURN to the main program to abort the calculation.

At statement 30, the program selects a segment K, and a few statements below this it selects another segment L. K is a segment of test dipole I, and L is a segment of expansion mode J. The mutual impedance between segments K and L is obtained by calling subroutine GGS or GGMM.

```
      SUBROUTINE SORT(IA,IB,I1,I2,I3,JA,JB,MD,ND,NM,NP,N,MAX,MIN      0001
     2,ICJ,INM)                                                       0002
      DIMENSION JSP(20)                                               0003
      DIMENSION I1(1),I2(1),I3(1),JA(1),JB(1)                         0004
      DIMENSION IA(1),IB(1),ND(1),MD(INM,4)                           0005
      I=0                                                             0006
      DO 24 K=1,NP                                                    0007
      NJK=0                                                           0008
      DO 20 J=1,NM                                                    0009
      IND=(IA(J)-K)*(IB(J)-K)                                         0010
      IF(IND.NE.0)GO TO 20                                            0011
      NJK=NJK+1                                                       0012
      JSP(NJK)=J                                                      0013
   20 CONTINUE                                                        0014
      MOD=NJK-1                                                       0015
      IF(MOD.LE.0)GO TO 24                                            0016
      DO 22 IMD=1,MOD                                                 0017
      I=I+1                                                           0018
      IF(I.GT.ICJ)GO TO 22                                            0019
      IPD=IMD+1                                                       0020
      JAI=JSP(IMD)                                                    0021
      JA(I)=JAI                                                       0022
      JBI=JSP(IPD)                                                    0023
      JB(I)=JBI                                                       0024
      I1(I)=IA(JAI)                                                   0025
      IF(IA(JAI).EQ.K)I1(I)=IB(JAI)                                   0026
      I2(I)=K                                                         0027
      I3(I)=IA(JBI)                                                   0028
      IF(IA(JBI).EQ.K)I3(I)=IB(JBI)                                   0029
   22 CONTINUE                                                        0030
   24 CONTINUE                                                        0031
      N=I                                                             0032
      DO 30 J=1,NM                                                    0033
      ND(J)=0                                                         0034
      DO 30 K=1,4                                                     0035
   30 MD(J,K)=0                                                       0036
      III=N                                                           0037
      IF(N.GT.ICJ)III=ICJ                                             0038
      DO 40 I=1,III                                                   0039
      J=JA(I)                                                         0040
      DO 38 L=1,2                                                     0041
      ND(J)=ND(J)+1                                                   0042
      K=1                                                             0043
      M=0                                                             0044
   32 MJK=MD(J,K)                                                     0045
      IF(MJK.NE.0)GO TO 34                                            0046
      M=1                                                             0047
      MD(J,K)=I                                                       0048
   34 K=K+1                                                           0049
      IF(K.GT.4)GO TO 38                                              0050
      IF(M.EQ.0)GO TO 32                                              0051
   38 J=JB(I)                                                         0052
   40 CONTINUE                                                        0053
      MIN=100                                                         0054
      MAX=0                                                           0055
      DO 46 J=1,NM                                                    0056
      NDJ=ND(J)                                                       0057
      IF(NDJ.GT.MAX)MAX=NDJ                                           0058
   46 IF(NDJ.LT.MIN)MIN=NDJ                                           0059
      RETURN                                                          0060
      END                                                             0061
```

Fig. 7. Subroutine SORT

```
      SUBROUTINE SGANT(IA,IB,INM,INT,ISC,I1,I2,I3,JA,JB,MD,N,ND,NM,NP      0001
     2,AM,BM,C,CGD,CMM,D,EP2,EP3,ETA,FHZ,GAM,SGD,X,Y,Z,ZLD,ZS)            0002
      COMPLEX ZG,ZH,ZS,EGD,GD,CGDS,SGDS,SGDT,B01                          0003
      COMPLEX P11,P12,P21,P22,Q11,Q12,Q21,Q22,EP2,EP,ETA,GAM,EP3          0004
      COMPLEX EPSILA,CWEA,BETA,ZARG                                       0005
      COMPLEX P(2,2),Q(2,2),CGD(1),SGD(1),C(1),ZLD(1)                     0006
      DIMENSION X(1),Y(1),Z(1),D(1),IA(1),IB(1),MD(INM,4)                 0007
      DIMENSION I1(1),I2(1),I3(1),JA(1),JB(1),ND(1),ISC(1)                0008
      DATA E0,TP,U0/8.854E-12,6.28318,1.2566E-6/                          0009
    2 FORMAT(3X,'AM = ',E10.3,3X,'DMAX = ',E10.3,3X,'DMIN = ',E10.3)      0010
      EP=EP3                                                              0011
      ICC=(N*N+N)/2                                                       0012
      DO 10 I=1,ICC                                                       0013
   10 C(I)=(.0,.0)                                                        0014
      ZS=(.0,.0)                                                          0015
      IF(CMM.LE.0.)GO TO 12                                               0016
      OMEGA=TP*FHZ                                                        0017
      EPSILA=CMPLX(E0,-CMM*1.E6/OMEGA)                                    0018
      CWEA=(.0,1.)*OMEGA*EPSILA                                           0019
      BETA=OMEGA*SQRT(U0)*CSQRT(EPSILA-EP)                                0020
      ZARG=BETA*AM                                                        0021
      CALL CBES(ZARG,B01)                                                 0022
      ZS=BETA*B01/CWEA                                                    0023
   12 ZH=ZS/(TP*AM*GAM)                                                   0024
      DMIN=1.E30                                                          0025
      DMAX=.0                                                             0026
      DO 20 J=1,NM                                                        0027
      K=IA(J)                                                             0028
      L=IB(J)                                                             0029
      D(J)=SQRT((X(K)-X(L))**2+(Y(K)-Y(L))**2+(Z(K)-Z(L))**2)             0030
      IF(D(J).LT.DMIN)DMIN=D(J)                                           0031
      IF(D(J).GT.DMAX)DMAX=D(J)                                           0032
      EGD=CEXP(GAM*D(J))                                                  0033
      CGD(J)=(EGD+1./EGD)/2.                                              0034
   20 SGD(J)=(EGD-1./EGD)/2.                                              0035
      IF(DMIN.LT.2.*AM)GO TO 25                                           0036
      IF(CABS(GAM*AM).GT.0.06)GO TO 25                                    0037
      IF(CABS(GAM*DMAX).GT.3.)GO TO 25                                    0038
      IF(AM.GT.0.)GO TO 30                                                0039
   25 N=0                                                                 0040
      WRITE(6,2)AM,DMAX,DMIN                                              0041
      RETURN                                                              0042
   30 DO 200 K=1,NM                                                       0043
      NDK=ND(K)                                                           0044
      KA=IA(K)                                                            0045
      KB=IB(K)                                                            0046
      DK=D(K)                                                             0047
      CGDS=CGD(K)                                                         0048
      SGDS=SGD(K)                                                         0049
      DO 200 L=1,NM                                                       0050
      NDL=ND(L)                                                           0051
      LA=IA(L)                                                            0052
      LB=IB(L)                                                            0053
      DL=D(L)                                                             0054
      SGDT=SGD(L)                                                         0055
      NIL=0                                                               0056
      DO 200 II=1,NDK                                                     0057
      I=MD(K,II)                                                          0058
      MM=(I-1)*N-(I*I-I)/2                                                0059
      FI=1.                                                               0060
      IF(KB.EQ.I2(I))GO TO 36                                             0061
      IF(KB.EQ.I1(I))FI=-1.                                               0062
```

Fig. 8a. Subroutine SGANT

```
      IS=1                                                              0063
      GO TO 40                                                          0064
   36 IF(KA.EQ.I3(I))FI=-1.                                             0065
      IS=2                                                              0066
   40 DO 200 JJ=1,NDL                                                   0067
      J=MD(L,JJ)                                                        0068
      MMM=MM+J                                                          0069
      IF(I.GT.J)GO TO 200                                               0070
      FJ=1.                                                             0071
      IF(LB.EQ.I2(J))GO TO 46                                           0072
      IF(LB.EQ.I1(J))FJ=-1.                                             0073
      JS=1                                                              0074
      GO TO 50                                                          0075
   46 IF(LA.EQ.I3(J))FJ=-1.                                             0076
      JS=2                                                              0077
   50 IF(NIL.NE.0)GO TO 168                                             0078
      NIL=1                                                             0079
      IF(K.EQ.L)GO TO 120                                               0080
      IND=(LA-KA)*(LB-KA)*(LA-KB)*(LB-KB)                               0081
      IF(IND.EQ.0)GO TO 80                                              0082
C     SEGMENTS K AND L SHARE NO POINTS                                  0083
      CALL GGS(X(KA),Y(KA),Z(KA),X(KB),Y(KB),Z(KB),X(LA),Y(LA),Z(LA)    0084
     2,X(LB),Y(LB),Z(LB),AM,DK,CGDS,SGDS,DL,SGDT,INT,ETA,GAM            0085
     3,P(1,1),P(1,2),P(2,1),P(2,2))                                     0086
      GO TO 168                                                         0087
C     SEGMENTS K AND L SHARE ONE POINT (THEY INTERSECT)                 0088
   80 KG=0                                                              0089
      JM=KB                                                             0090
      JC=KA                                                             0091
      KF=1                                                              0092
      IND=(KB-LA)*(KB-LB)                                               0093
      IF(IND.NE.0)GO TO 82                                              0094
      JC=KB                                                             0095
      KF=-1                                                             0096
      JM=KA                                                             0097
      KG=3                                                              0098
   82 LG=3                                                              0099
      JP=LA                                                             0100
      LF=-1                                                             0101
      IF(LB.EQ.JC)GO TO 83                                              0102
      JP=LB                                                             0103
      LF=1                                                              0104
      LG=0                                                              0105
   83 SGN=KF*LF                                                         0106
      CPSI=((X(JP)-X(JC))*(X(JM)-X(JC))+(Y(JP)-Y(JC))*(Y(JM)-Y(JC))     0107
     2+(Z(JP)-Z(JC))*(Z(JM)-Z(JC)))/(DK*DL)                             0108
      CALL GGMM(.0,DK,.0,DL,AM,CGDS,SGDS,SGDT,CPSI,ETA,GAM              0109
     2,Q(1,1),Q(1,2),Q(2,1),Q(2,2))                                     0110
      DO 98 KK=1,2                                                      0111
      KP=IABS(KK-KG)                                                    0112
      DO 98 LL=1,2                                                      0113
      LP=IABS(LL-LG)                                                    0114
      P(KP,LP)=SGN*Q(KK,LL)                                             0115
   98 CONTINUE                                                          0116
      GO TO 168                                                         0117
C     K=L  (SELF REACTION OF SEGMENT K)                                 0118
  120 Q11=(.0,.0)                                                       0119
      Q12=(.0,.0)                                                       0120
      IF(CMM.LE.0.)GO TO 150                                            0121
      GD=GAM*DK                                                         0122
      ZG=ZH/(SGDS**2)                                                   0123
      Q11=ZG*(SGDS*CGDS-GD)/2.                                          0124
```

Fig. 8b. Subroutine SGANT

```
      Q12=ZG*(GD*CGDS-SGDS)/2.                                          0125
  150 ISCK=ISC(K)                                                       0126
      P11=(.0,.0)                                                       0127
      P12=(.0,.0)                                                       0128
      IF(ISCK.EQ.0)GO TO 155                                            0129
      IF(BM.LE.AM)GO TO 155                                             0130
      CALL        DSHELL(AM,BM,DK,CGDS,SGDS,EP2,EP,ETA,GAM,P11,P12)     0131
  155 Q11=P11+Q11                                                       0132
      Q12=P12+Q12                                                       0133
      CALL GGMM(.0,DK,.0,DK,AM,CGDS,SGDS,SGDS,1.                        0134
     2,ETA,GAM,P11,P12,P21,P22)                                         0135
      Q11=P11+Q11                                                       0136
      Q12=P12+Q12                                                       0137
      P(1,1)=Q11                                                        0138
      P(1,2)=Q12                                                        0139
      P(2,1)=Q12                                                        0140
      P(2,2)=Q11                                                        0141
      IF(KA.NE.LA)GO TO 160                                             0142
      GO TO 168                                                         0143
  160 P(1,1)=-Q12                                                       0144
      P(1,2)=-Q11                                                       0145
      P(2,1)=-Q11                                                       0146
      P(2,2)=-Q12                                                       0147
  168 C(MMM)=C(MMM)+FI*FJ*P(IS,JS)                                      0148
  200 CONTINUE                                                          0149
      DO 220 I=1,N                                                      0150
      IJ=(I-1)*N-(I*I-I)/2+I                                            0151
      J1=JA(I)                                                          0152
      IF(I2(I).EQ.IB(J1))J1=J1+NM                                       0153
      J2=JB(I)                                                          0154
      IF(I2(I).EQ.IB(J2))J2=J2+NM                                       0155
  220 C(IJ)=C(IJ)+ZLD(J1)+ZLD(J2)                                       0156
      RETURN                                                            0157
      END                                                               0158
```

Fig. 8c. Subroutine SGANT

In statement 168, this impedance is lumped into C(MMM). The mutual impedance Z_{ij} between dipoles I and J is the sum of four segment-segment impedances.

In SGANT, segment K has endpoints KA and KB, and segment L has endpoints LA and LB. It is convenient to think of KA and KB as points 1 and 2 on segment K, and LA and LB as points 1 and 2 on L. Now we define four segment-segment impedances P(IS,JS). The first subscript IS refers to the terminal point on segment K, and the second subscript JS refers to the terminal point on L. Thus IS = 1 or 2 if dipole I has its terminal point I2(I) at KA (point 1) or KB (point 2), respectively. Similarly, JS = 1 or 2 if mode J has its terminal point I2(J) at LA or LB. The impedances P(IS,JS) are defined with the following reference directions for current flow: from point 1 toward point 2 on each segment. If dipole I has this same reference direction on segment K, we set FI = 1; otherwise FI = -1. Similarly FJ = 1 or -1 in accordance with the reference direction for mode J on segment L. In statement 168, P(IS,JS) is multiplied by FI and FJ before its contribution is added to Z_{ij}.

Subroutine GGMM calculates the impedances Q(KK,LL) which are like the P(IS,JS) but have different conventions for reference directions and subscript meaning. The transformation from the Q impedances to the P impedances is accomplished in the DO LOOP ending with statement 98.

If the wire has finite conductivity, the appropriate modification is applied to the impedance matrix just above statement 150. (See Eqs. 27 through 29 in Reference 1.) The terms arising from the dielectric shell on an insulated segment are obtained from subroutine DSHELL just above statement 155. Finally, the lumped loads ZLD are added to the diagonal elements of the impedance matrix in statement 220.

The impedance matrix could be calculated in a different order as follows. Select modes I and J, calculate ZIJ, and then increment I or J. Instead, SGANT selects segments K and L, calculates ZKL, adds ZKL to all the appropriate elements ZIJ, and then increments K or L. This minimizes the calls to GGS and GGMM and presumably improves the computational efficiency.

K is a segment of test dipole I, and L is a segment of expansion mode J. When the segment numbers K and L are equal, SGANT calls GGMM to obtain the mutual impedance between two filamentary electric monopoles. These monopoles are parallel and have the same length. Monopole K is positioned on the axis of the wire segment, and monopole L is on the surface of the same wire segment. Thus, the displacement is equal to the wire radius. The two monopoles are side-by-side with no stagger.

When segments K and L intersect, SGANT again calls GGMM for the mutual impedance between the two filamentary monopoles. Monopole K is

situated on the axis of wire segment K, and monopole L is on the surface of wire segment L. The axes of segments K and L define a plane P, and monopole K lies in this plane. Monopole L is parallel with plane P and is displaced from it by a distance equal to the wire radius.

APPENDIX 3. Subroutine CBES

Subroutine CBES, listed in Fig. 9, calculates the quantity BO1 = $J_0(z)/J_1(z)$ where z is complex and J_0 and J_1 denote the Bessel functions of order zero and one.

APPENDIX 4. Subroutine DSHELL

Subroutine DSHELL, listed in Fig. 10, calculates the mutual impedance term contributed by the dielectric insulation on the surface of a thin wire. This subroutine uses Eq. 35 of Reference 1.

APPENDIX 5. Subroutine GGS

Subroutine GGS, listed in Fig. 11, calculates the mutual impedance between two filamentary monopoles with sinusoidal current distributions. (The dipole-dipole mutual impedance in Eq. 20 of Reference 1 is the sum of four monopole-monopole mutual impedances.) The endpoints of the axial test monopole s are (XA,YA,ZA) and (XB,YB,ZB), and the endpoints of the expansion monopole t are (X1,Y1,Z1) and (X2,Y2,Z2). DS and DT denote the lengths of monopoles s and t, respectively. CAS, CBS and CGS are the direction cosines of monopole s, and CA, CB and CG are the direction cosines of monopole t.

If INT = 0, GGS calls GGMM for the closed-form impedance calculations. Otherwise GGS calculates the mutual impedance via Simpson's-rule integration with the following number of sample points: IP = INT + 1. If the monopoles are parallel with small displacement, GGS calls GGMM to avoid the difficulties of numerical integration.

For the fields of the test monopole, GGS uses Eqs. 75 and 76 of Reference 1. The current distribution on the expansion monopole is given by Eq. 74 of Reference 1. With an origin at (X1,Y1,Z1), the coordinate T measures distance along the expansion monopole. Thus T is the integration variable.

Let the coordinate s measure distance along the test monopole with origin at (XA,YA,ZA). From any point T on monopole t, construct a line to the test monopole such that the line is perpendicular to the test monopole. SZ denotes the s coordinate of the intersection of this line with the test monopole. The length of the line is the radial coordinate ρ, and RS denotes ρ^2. R1 and R2 are the distances from (XA,YA,ZA) and (XB,YB,ZB) to the point T. C1 is the current at T for the mode with terminals at (X1,Y1,Z1), and C2 is the current at T for the other mode with terminals at (X2,Y2,Z2). C denotes the Simpson's-rule weighting coefficient.

```
      SUBROUTINE CBES(Z,BO1)                                            0001
      COMPLEX ARG,CC,CS,EX                                              0002
      COMPLEX BO1,Z,TERMJ,TERMN,MZ24,JN(2)                              0003
      DATA PI/3.14159/                                                  0004
      IF(CABS( Z ).GE.12.0) GO TO10                                     0005
      FACTOR=0.0                                                        0006
      TERMN=(0.,0.)                                                     0007
      MZ24=-0.25*Z*Z                                                    0008
      TERMJ=(1.0,0.0)                                                   0009
      DO 1 NP=1,2                                                       0010
      N=NP-1                                                            0011
      JN(NP)=TERMJ                                                      0012
      M=0                                                               0013
    2 M=M+1                                                             0014
      TERMJ=TERMJ*MZ24/FLOAT(M*(N+M))                                   0015
      JN(NP)=JN(NP)+TERMJ                                               0016
      IF(NP.NE.1) GO TO 3                                               0017
      FACTOR=FACTOR+1.0/FLOAT(M)                                        0018
      TERMN=TERMN+TERMJ*FACTOR                                          0019
    3 ERROR=CABS(TERMJ)                                                 0020
      IF(ERROR.GT.1.0E-10) GO TO 2                                      0021
    1 TERMJ=0.5*Z                                                       0022
      BO1=JN(1)/JN(2)                                                   0023
      RETURN                                                            0024
   10 Y=AIMAG(Z)                                                        0025
      IF(ABS(Y).GT.20.)GO TO 20                                         0026
      ARG=(.0,1.)*Z                                                     0027
      EX=CEXP(ARG)                                                      0028
      CC=EX+1./EX                                                       0029
      CS=(.0,-1.)*(EX-1./EX)                                            0030
      BO1=(CS+CC)/(CS-CC)                                               0031
      RETURN                                                            0032
   20 BO1=(.0,-1.)                                                      0033
      IF(Y.LT.0.)BO1=(.0,1.)                                            0034
      RETURN                                                            0035
      END                                                               0036
```

Fig. 9. Subroutine CBES

```
      SUBROUTINE DSHELL(AM,BM,DK,CGDS,SGDS,EP2,EP,ETA,GAM,P11,P12)      0001
      COMPLEX CGDS,SGDS,EP2,EP,ETA,GAM,P11,P12,GD,CST                   0002
      DATA PI/3.14159/                                                  0003
      GD=GAM*DK                                                         0004
      CST=(EP2-EP )*ETA*ALOG(BM/AM)/(4.*PI*EP2*SGDS*SGDS)               0005
      P11=-CST*(GD+SGDS*CGDS)                                           0006
      P12=CST*(GD*CGDS+SGDS)                                            0007
      RETURN                                                            0008
      END                                                               0009
```

Fig. 10. Subroutine DSHELL.

```
      SUBROUTINE GGS(XA,YA,ZA,XB,YB,ZB,X1,Y1,Z1,X2,Y2,Z2,AM                0001
     2,DS,CGDS,SGDS,DT,SGDT,INT,ETA,GAM,P11,P12,P21,P22)                  0002
      COMPLEX P11,P12,P21,P22,EJA,EJB,EJ1,EJ2,ETA,GAM,C1,C2,CST           0003
      COMPLEX EGD,CGDS,SGDS,SGDT,ER1,ER2,ET1,ET2                          0004
      DATA FP/12.56637/                                                   0005
      CA=(X2-X1)/DT                                                       0006
      CB=(Y2-Y1)/DT                                                       0007
      CG=(Z2-Z1)/DT                                                       0008
      CAS=(XB-XA)/DS                                                      0009
      CBS=(YB-YA)/DS                                                      0010
      CGS=(ZB-ZA)/DS                                                      0011
      CC=CA*CAS+CB*CBS+CG*CGS                                             0012
      IF(ABS(CC).GT..997)GO TO 200                                        0013
   20 SZ=(X1-XA)*CAS+(Y1-YA)*CBS+(Z1-ZA)*CGS                              0014
      IF(INT.LE.0)GO TO 300                                               0015
      INS=2*(INT/2)                                                       0016
      IF(INS.LT.2)INS=2                                                   0017
      IP=INS+1                                                            0018
      DELT=DT/INS                                                         0019
      T=.0                                                                0020
      DSZ=CC*DELT                                                         0021
      P11=(.0,.0)                                                         0022
      P12=(.0,.0)                                                         0023
      P21=(.0,.0)                                                         0024
      P22=(.0,.0)                                                         0025
      AMS=AM*AM                                                           0026
      SGN=-1.                                                             0027
      DO 100 IN=1,IP                                                      0028
      ZZ1=SZ                                                              0029
      ZZ2=SZ-DS                                                           0030
      XXZ=X1+T*CA-XA-SZ*CAS                                               0031
      YYZ=Y1+T*CB-YA-SZ*CBS                                               0032
      ZZZ=Z1+T*CG-ZA-SZ*CGS                                               0033
      RS=XXZ**2+YYZ**2+ZZZ**2                                             0034
      R1=SQRT(RS+ZZ1**2)                                                  0035
      EJA=CEXP(-GAM*R1)                                                   0036
      EJ1=EJA/R1                                                          0037
      R2=SQRT(RS+ZZ2**2)                                                  0038
      EJB=CEXP(-GAM*R2)                                                   0039
      EJ2=EJB/R2                                                          0040
      ER1=EJA*SGDS+ZZ1*EJ1*CGDS-ZZ2*EJ2                                   0041
      ER2=-EJB*SGDS+ZZ2*EJ2*CGDS-ZZ1*EJ1                                  0042
      FAC=.0                                                              0043
      IF(RS.GT.AMS)FAC=(CA*XXZ+CB*YYZ+CG*ZZZ)/RS                          0044
      ET1=CC*(EJ2-EJ1*CGDS)+FAC*ER1                                       0045
      ET2=CC*(EJ1-EJ2*CGDS)+FAC*ER2                                       0046
      C=3.+SGN                                                            0047
      IF(IN.EQ.1 .OR. IN.EQ.IP)C=1.                                       0048
      EGD=CEXP(GAM*(DT-T))                                                0049
      C1=C*(EGD-1./EGD)/2.                                                0050
      EGD=CEXP(GAM*T)                                                     0051
      C2=C*(EGD-1./EGD)/2.                                                0052
      P11=P11+ET1*C1                                                      0053
      P12=P12+ET1*C2                                                      0054
      P21=P21+ET2*C1                                                      0055
      P22=P22+ET2*C2                                                      0056
      T=T+DELT                                                            0057
      SZ=SZ+DSZ                                                           0058
  100 SGN=-SGN                                                            0059
      CST=-ETA*DELT/(3.*FP*SGDS*SGDT)                                     0060
      P11=CST*P11                                                         0061
      P12=CST*P12                                                         0062
```

Fig. 11a. Subroutine GGS

```
      P21=CST*P21                                                          0063
      P22=CST*P22                                                          0064
      RETURN                                                               0065
200   SZ1=(X1-XA)*CAS+(Y1-YA)*CBS+(Z1-ZA)*CGS                              0066
      RH1=SQRT((X1-XA-SZ1*CAS)**2+(Y1-YA-SZ1*CBS)**2+(Z1-ZA-SZ1*CGS)**2)   0067
      SZ2=SZ1+DT*CC                                                        0068
      RH2=SQRT((X2-XA-SZ2*CAS)**2+(Y2-YA-SZ2*CBS)**2+(Z2-ZA-SZ2*CGS)**2)   0069
      DDD=(RH1+RH2)/2.                                                     0070
      IF(DDD.GT.20.*AM .AND. INT.GT.0)GO TO 20                             0071
      IF(DDD.LT.AM)DDD=AM                                                  0072
      CALL GGMM(.0,DS,SZ1,SZ2,DDD,CGDS,SGDS,SGDT,1.                       0073
     2,ETA,GAM,P11,P12,P21,P22)                                            0074
      RETURN                                                               0075
300   SS=SQRT(1.-CC*CC)                                                    0076
      CAD=(CGS*CB-CBS*CG)/SS                                               0077
      CBD=(CAS*CG-CGS*CA)/SS                                               0078
      CGD=(CBS*CA-CAS*CB)/SS                                               0079
      DK=(X1-XA)*CAD+(Y1-YA)*CBD+(Z1-ZA)*CGD                               0080
      DK=ABS(DK)                                                           0081
      IF(DK.LT.AM)DK=AM                                                    0082
      XZ=XA+SZ*CAS                                                         0083
      YZ=YA+SZ*CBS                                                         0084
      ZZ=ZA+SZ*CGS                                                         0085
      XP1=X1-DK*CAD                                                        0086
      YP1=Y1-DK*CBD                                                        0087
      ZP1=Z1-DK*CGD                                                        0088
      CAP=CBS*CGD-CGS*CBD                                                  0089
      CBP=CGS*CAD-CAS*CGD                                                  0090
      CGP=CAS*CBD-CBS*CAD                                                  0091
      P1=CAP*(XP1-XZ)+CBP*(YP1-YZ)+CGP*(ZP1-ZZ)                            0092
      T1=P1/SS                                                             0093
      S1=T1*CC-SZ                                                          0094
      CALL GGMM(S1,S1+DS,T1,T1+DT,DK,CGDS,SGDS,SGDT,CC,ETA,GAM             0095
     2,P11,P12,P21,P22)                                                    0096
      RETURN                                                               0097
      END                                                                  0098
```

Fig. 11b. Subroutine GGS

Below statement 300, GGS performs some analytic geometry in preparation for calling GGMM. The remaining part of this Appendix concerns this last part of subroutine GGS.

Let $\hat{s}$ denote a unit vector in the direction from (XA,YA,ZA) toward (XB,YB,ZB). Also let $\hat{t}$ denote a unit vector from (X1,Y1,Z1) toward (X2,Y2,Z2). Then $\hat{s} \cdot \hat{t} = \cos\theta = CC$ where θ is the angle formed by the axes of the two monopoles. Let monopole s lie in one plane P_s and monopole t lie in another parallel plane P_t. CAD, CBD and CGD are the direction cosines of the unit vector $\hat{d} = \hat{t} \times \hat{s} / \sin\theta$ which is perpendicular to both planes. To obtain the distance DK between the two planes, we construct a vector $\underline{R}_{11}$ from (XA,YA,ZA) to (X1,Y1,Z1) and take $DK = \underline{R}_{11} \cdot \hat{d}$.

Construct a line from (X1,Y1,Z1) to the test monopole, such that the line is perpendicular to the test monopole. SZ denotes the s coordinate of the intersection of this line with the test monopole, and the cartesian coordinates of this intersection are XZ, YZ and ZZ. The direction cosines of $\hat{s} \times \hat{d}$ are CAP, CBP and CGP.

From the point (X1,Y1,Z1) in plane P_t, construct a perpendicular line to the point (XP1,YP1,ZP1) in the plane P_s. This line is parallel with $\hat{d}$ and has length DK. Let $\underline{R}$ represent a vector from (XZ,YZ,ZZ) to (XP1,YP1,ZP1). P1 denotes $\underline{R} \cdot (\hat{s} \times \hat{d})$. S1 and T1 are defined in the next Appendix.

APPENDIX 6. Subroutine GGMM

Subroutine GGMM calculates the mutual impedance between two filamentary monopoles with sinusoidal current distributions. The dipole-dipole mutual impedance in Eq. 20 of Reference 1 is the sum of four monopole-monopole mutual impedances. The monopole impedances are calculated by GGS with Simpson's rule or by GGMM with closed-form expressions in terms of exponential integrals.

To explain the input data for GGMM, reference is made to Fig. 12. Subroutine GGMM is listed in Fig. 13. If the monopoles are parallel, let the z axis be parallel with both monopoles. The coordinate origin may be selected arbitrarily. S1 and S2 denote the z coordinates of the endpoints of the test monopole, T1 and T2 are the z coordinates of the endpoints of the expansion monopole, and D is the perpendicular distance (displacement) between the monopoles. The mutual impedance of parallel monopoles is calculated in the last part of GGMM below statement 110.

For skew monopoles, let the test monopole s lie in the xy plane and the expansion monopole t in the plane z = D. (D is the perpendicular distance between the parallel planes.) If the monopoles are viewed along a line of sight parallel with the z axis as in Fig. 12, the extended axes of the two monopoles will appear to intersect at a point on the xy plane. Let s measure the distance along the axis of

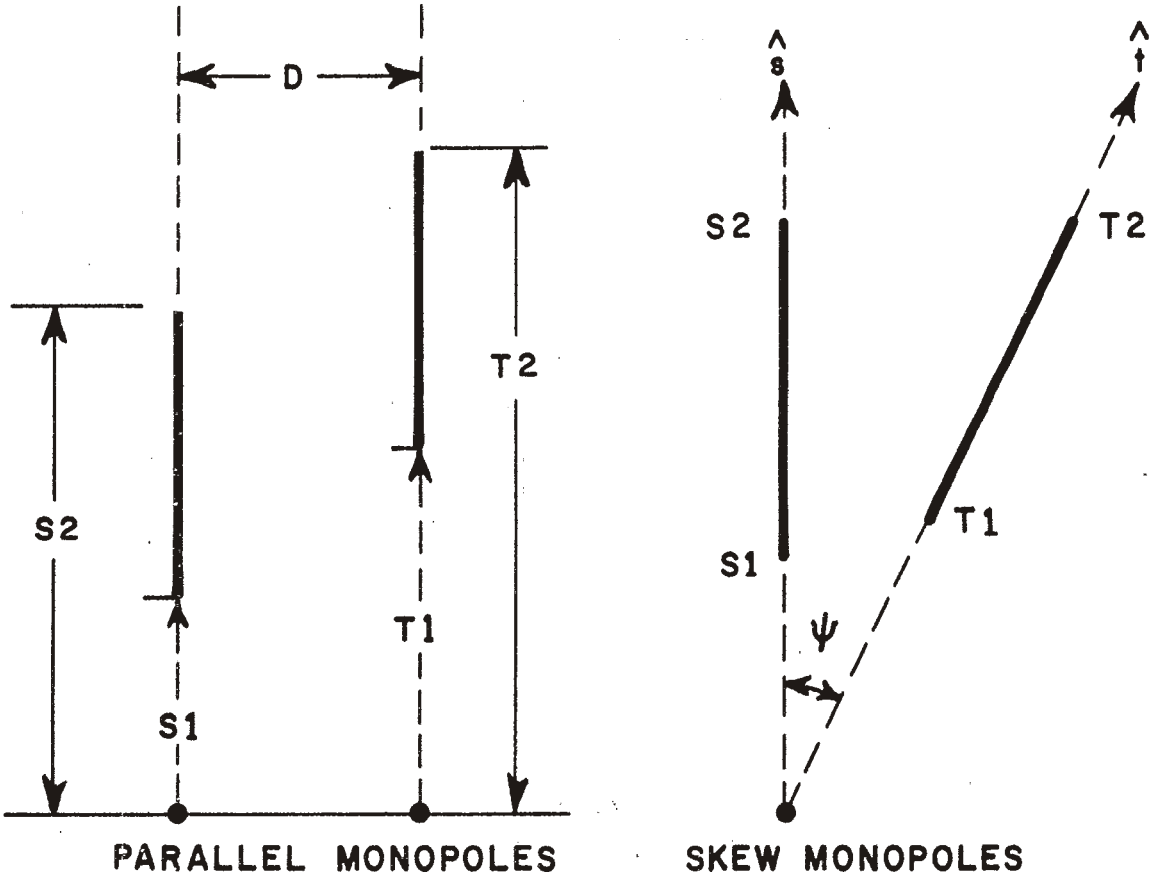


Fig. 12. Coordinates for parallel and skew monopoles in subroutine GGMM.

```
      SUBROUTINE GGMM(S1,S2,T1,T2,D,CGDS,SGD1,SGD2,CPSI,ETA,GAM          0001
     2,P11,P12,P21,P22)                                                 0002
      DOUBLE PRECISION R1,R2,DPQ,SIS,TS1,TS2,ST1,ST2,CD,BD,CPSS,SK      0003
     2,TL1,TL2,TD1,TD2,SDI,DPSI,DD,ZD                                   0004
      COMPLEX CGDS,SGDS,SGDT,SGD1,SGD2,ETA,GAM,P11,P12,P21,P22          0005
      COMPLEX CST,EB,EC,EK,EL,EKL,EGZI,ES1,ES2,ET1,ET2,EXPA,EXPB        0006
      COMPLEX E(2,2),F(2,2)                                             0007
      COMPLEX EGZ(2,2),GM(2),GP(2)                                      0008
      DATA PI/3.14159/                                                  0009
      DSQ=D*D                                                           0010
      SGDS=SGD1                                                         0011
      IF(S2.LT.S1)SGDS=-SGD1                                            0012
      SGDT=SGD2                                                         0013
      IF(T2.LT.T1)SGDT=-SGD2                                            0014
      IF(ABS(CPSI).GT..997)GO TO 110                                    0015
      ES1=CEXP(GAM*S1)                                                  0016
      ES2=CEXP(GAM*S2)                                                  0017
      ET1=CEXP(GAM*T1)                                                  0018
      ET2=CEXP(GAM*T2)                                                  0019
      DD=D                                                              0020
      DPSI=CPSI                                                         0021
      TD1=T1                                                            0022
      TD2=T2                                                            0023
      CPSS=DPSI*DPSI                                                    0024
      CD=DD/DSQRT(1.DO-CPSS)                                            0025
      C=CD                                                              0026
      BD=CD*DPSI                                                        0027
      B=BD                                                              0028
      EB=CEXP(GAM*CMPLX(.0,B))                                          0029
      EC=CEXP(GAM*CMPLX(.0,C))                                          0030
      DO 10 K=1,2                                                       0031
      DO 10 L=1,2                                                       0032
10    E(K,L)=(.0,.0)                                                    0033
      TS1=TD1*TD1                                                       0034
      TS2=TD2*TD2                                                       0035
      DPQ=DD*DD                                                         0036
      SI=S1                                                             0037
      DO 100 I=1,2                                                      0038
      FI=(-1)**I                                                        0039
      SDI=SI                                                            0040
      SIS=SDI*SDI                                                       0041
      ST1=2.*SDI*TD1*DPSI                                               0042
      ST2=2.*SDI*TD2*DPSI                                               0043
      R1=DSQRT(DPQ+SIS+TS1-ST1)                                         0044
      R2=DSQRT(DPQ+SIS+TS2-ST2)                                         0045
      EK=EB                                                             0046
      DO 50 K=1,2                                                       0047
      FK=(-1)**K                                                        0048
      SK=FK*SDI                                                         0049
      EL=EC                                                             0050
      DO 40 L=1,2                                                       0051
      FL=(-1)**L                                                        0052
      EKL=EK*EL                                                         0053
      XX=FK*BD+FL*CD                                                    0054
      TL1=FL*TD1                                                        0055
      TL2=FL*TD2                                                        0056
      RR1=R1+SK+TL1                                                     0057
      RR2=R2+SK+TL2                                                     0058
      CALL EXPJ(GAM*CMPLX(RR1,-XX),GAM*CMPLX(RR2,-XX),EXPA)             0059
      CALL EXPJ(GAM*CMPLX(RR1,XX),GAM*CMPLX(RR2,XX),EXPB)               0060
      E(K,L)=E(K,L)+FI*(EXPA*EKL+EXPB/EKL)                              0061
40    EL=1./EC                                                          0062
```

Fig. 13a. Subroutine GGMM

```
50  EK=1./EB                                                          0063
    ZD=SDI*DPSI                                                       0064
    ZC=ZD                                                             0065
    EGZI=CEXP(GAM*ZC)                                                 0066
    RR1=R1+ZD-TD1                                                     0067
    RR2=R2+ZD-TD2                                                     0068
    CALL EXPJ(GAM*RR1,GAM*RR2,EXPB)                                   0069
    RR1=R1-ZD+TD1                                                     0070
    RR2=R2-ZD+TD2                                                     0071
    CALL EXPJ(GAM*RR1,GAM*RR2,EXPA)                                   0072
    F(I,1)=2.*SGDS*EXPA/EGZI                                          0073
    F(I,2)=2.*SGDS*EXPB*EGZI                                          0074
100 SI=S2                                                             0075
    CST=ETA/(16.*PI*SGDS*SGDT)                                        0076
    P11=CST*(( F(1,1)+E(2,2)*ES2-E(1,2)/ES2)*ET2                      0077
   A         +(-F(1,2)-E(2,1)*ES2+E(1,1)/ES2)/ET2)                    0078
    P12=CST*((-F(1,1)-E(2,2)*ES2+E(1,2)/ES2)*ET1                      0079
   B         +( F(1,2)+E(2,1)*ES2-E(1,1)/ES2)/ET1)                    0080
    P21=CST*((-F(2,1)-E(2,2)*ES1+E(1,2)/ES1)*ET2                      0081
   C         +( F(2,2)+E(2,1)*ES1-E(1,1)/ES1)/ET2)                    0082
    P22=CST*(( F(2,1)+E(2,2)*ES1-E(1,2)/ES1)*ET1                      0083
   D         +(-F(2,2)-E(2,1)*ES1+E(1,1)/ES1)/ET1)                    0084
    RETURN                                                            0085
110 IF(CPSI.LT.0.)GO TO 120                                           0086
    TA=T1                                                             0087
    TB=T2                                                             0088
    GO TO 130                                                         0089
120 TA=-T1                                                            0090
    TB=-T2                                                            0091
    SGDT=-SGDT                                                        0092
130 SI=S1                                                             0093
    DO 150 I=1,2                                                      0094
    TJ=TA                                                             0095
    DO 140 J=1,2                                                      0096
    ZIJ=TJ-SI                                                         0097
    R=SQRT(DSQ+ZIJ*ZIJ)                                               0098
    W=R+ZIJ                                                           0099
    IF(ZIJ.LT.0.)W=DSQ/(R-ZIJ)                                        0100
    V=R-ZIJ                                                           0101
    IF(ZIJ.GT.0.)V=DSQ/(R+ZIJ)                                        0102
    IF(J.EQ.1)V1=V                                                    0103
    IF(J.EQ.1)W1=W                                                    0104
    EGZ(I,J)=CEXP(GAM*ZIJ)                                            0105
140 TJ=TB                                                             0106
    CALL EXPJ(GAM*V1,GAM*V,GP(I))                                     0107
    CALL EXPJ(GAM*W1,GAM*W,GM(I))                                     0108
150 SI=S2                                                             0109
    CST=-ETA/(8.*PI*SGDS*SGDT)                                        0110
    P11=CST*(GM(2)*EGZ(2,2)+GP(2)/EGZ(2,2)                            0111
   2-CGDS*(GM(1)*EGZ(1,2)+GP(1)/EGZ(1,2)))                            0112
    P12=CST*(-GM(2)*EGZ(2,1)-GP(2)/EGZ(2,1)                           0113
   2+CGDS*(GM(1)*EGZ(1,1)+GP(1)/EGZ(1,1)))                            0114
    P21=CST*(GM(1)*EGZ(1,2)+GP(1)/EGZ(1,2)                            0115
   2-CGDS*(GM(2)*EGZ(2,2)+GP(2)/EGZ(2,2)))                            0116
    P22=CST*(-GM(1)*EGZ(1,1)-GP(1)/EGZ(1,1)                           0117
   2+CGDS*(GM(2)*EGZ(2,1)+GP(2)/EGZ(2,1)))                            0118
    RETURN                                                            0119
    END                                                               0120
```

Fig. 13b. Subroutine GGMM

the test monopole with origin at the apparent intersection. S1 and S2 denote the s coordinates of the endpoints of the test monopole. Similarly, let t measure distance along the axis of the expansion monopole with origin at the apparent intersection. T1 and T2 denote the t coordinates of the endpoints of the expansion monopole. Let $\hat{s}$ and $\hat{t}$ be unit vectors parallel with the positive s and t axes, respectively. Then CPSI = $\hat{s} \cdot \hat{t} = \cos \psi$. The monopole lengths are d_s and d_t, and the remaining input data are defined as follows:

CGDS	$\cosh \gamma d_s$
SGD1	$\sinh \gamma d_s$
SGD2	$\sinh \gamma d_t$

GGMM calls EXPJ for the exponential integrals.

The output data from GGMM are the impedances P11, P12, P21, and P22. In defining these impedances, the reference direction is from S1 to S2 for the current on monopole s, and from T1 to T2 for the current on monopole t. In the impedance P_{ij}, the first subscript is 1 or 2 if the test dipole has terminals at S1 or S2 on monopole s. The second subscript is 1 or 2 if the expansion dipole has terminals at T1 or T2 on monopole t. The endpoint coordinates S1, S2, T1 and T2 may be positive or negative. The monopole lengths d_s and d_t are assumed positive in defining the input data CGDS, SGD1 and SGD2.

For parallel monopoles, CPSI = 1 or -1. S1, S2, T1 and T2 are cartesian coordinates for parallel monopoles and spherical coordinates for skew monopoles. For skew monopoles, the radial coordinates S1, S2, T1 and T2 tend to infinity as the angle ψ tends to zero or π. Therefore, if the monopoles are within 4.5° of being parallel, they are approximated by parallel dipoles.

APPENDIX 7. Subroutine EXPJ

Subroutine EXPJ, listed in Fig. 14, evaluates the exponential integral defined as follows:

$$(2) \qquad W12 = \int_{V1}^{V2} \frac{e^{-v}\, dv}{v} = E_1(V1) - E_1(V2) + j\, 2n\pi$$

where the integration path is the straight line from V1 to V2 on the complex v plane and

$$(3) \qquad E_1(z) = \int_{z}^{\infty} \frac{e^{-t}\, dt}{t}$$

```
      SUBROUTINE EXPJ(V1,V2,W12)                                        0001
      COMPLEX EC,E15,S,T,UC,VC,V1,V2,W12,Z                              0002
      DIMENSION V(21),W(21),D(16),E(16)                                 0003
      DATA V/   0.22284667E 00,                                         0004
     20.11889321E 01,0.29927363E 01,0.57751436E 01,0.98374674E 01,      0005
     20.15982874E 02,0.93307812E-01,0.49269174E 00,0.12155954E 01,      0006
     20.22699495E 01,0.36676227E 01,0.54253366E 01,0.75659162E 01,      0007
     20.10120228E 02,0.13130282E 02,0.16654408E 02,0.20776479E 02,      0008
     20.25623894E 02,0.31407519E 02,0.38530683E 02,0.48026086E 02/      0009
      DATA W/   0.45896460E 00,                                         0010
     20.41700083E 00,0.11337338E 00,0.10399197E-01,0.26101720E-03,      0011
     20.89854791E-06,0.21823487E 00,0.34221017E 00,0.26302758E 00,      0012
     20.12642582E 00,0.40206865E-01,0.85638778E-02,0.12124361E-02,      0013
     20.11167440E-03,0.64599267E-05,0.22263169E-06,0.42274304E-08,      0014
     20.39218973E-10,0.14565152E-12,0.14830270E-15,0.16005949E-19/      0015
      DATA D/   0.22495842E 02,                                         0016
     2 0.74411568E 02,-0.41431576E 03,-0.78754339E 02, 0.11254744E 02,  0017
     2 0.16021761E 03,-0.23862195E 03,-0.50094687E 03,-0.68487854E 02,  0018
     2 0.12254778E 02,-0.10161976E 02,-0.47219591E 01, 0.79729681E 01,  0019
     2-0.21069574E 02, 0.22046490E 01, 0.89728244E 01/                  0020
      DATA E/   0.21103107E 02,                                         0021
     2-0.37959787E 03,-0.97489220E 02, 0.12900672E 03, 0.17949226E 02,  0022
     2-0.12910931E 03,-0.55705574E 03, 0.13524801E 02, 0.14696721E 03,  0023
     2 0.17949528E 02,-0.32981014E 00, 0.31028836E 02, 0.81657657E 01,  0024
     2 0.22236961E 02, 0.39124892E 02, 0.81636799E 01/                  0025
      Z=V1                                                              0026
      DO 100 JIM=1,2                                                    0027
      X=REAL(Z)                                                         0028
      Y=AIMAG(Z)                                                        0029
      E15=(.0,.0)                                                       0030
      AB=CABS(Z)                                                        0031
      IF(AB.EQ.0.)GO TO 90                                              0032
      IF(X.GE.0. .AND. AB.GT.10.)GO TO 80                               0033
      YA=ABS(Y)                                                         0034
      IF(X.LE.0. .AND. YA.GT.10.)GO TO 80                               0035
      IF(YA-X.GE.17.5.OR.YA.GE.6.5.OR.X+YA.GE.5.5.OR.X.GE.3.)GO TO 20   0036
      IF(X.LE.-9.)GO TO 40                                              0037
      IF(YA-X.GE.2.5)GO TO 50                                           0038
      IF(X+YA.GE.1.5)GO TO 30                                           0039
   10 N=6.+3.*AB                                                        0040
      E15=1./(N-1.)-Z/N**2                                              0041
   15 N=N-1.                                                            0042
      E15=1./(N-1.)-Z*E15/N                                             0043
      IF(N.GE.3)GO TO 15                                                0044
      E15=Z*E15-CMPLX(.577216+ALOG(AB),ATAN2(Y,X))                      0045
      GO TO 90                                                          0046
   20 J1=1                                                              0047
      J2=6                                                              0048
      GO TO 31                                                          0049
   30 J1=7                                                              0050
      J2=21                                                             0051
   31 S=(.0,.0)                                                         0052
      YS=Y*Y                                                            0053
      DO 32 I=J1,J2                                                     0054
      XI=V(I)+X                                                         0055
      CF=W(I)/(XI*XI+YS)                                                0056
   32 S=S+CMPLX(XI*CF,-YA*CF)                                           0057
      GO TO 54                                                          0058
   40 T3=X*X-Y*Y                                                        0059
      T4=2.*X*YA                                                        0060
      T5=X*T3-YA*T4                                                     0061
      T6=X*T4+YA*T3                                                     0062
```

Fig. 14a. Subroutine EXPJ

```
     UC=CMPLX(D(11)+D(12)*X+D(13)*T3+T5-E(12)*YA-E(13)*T4,                0063
    2          E(11)+E(12)*X+E(13)*T3+T6+D(12)*YA+D(13)*T4)               0064
     VC=CMPLX(D(14)+D(15)*X+D(16)*T3+T5-E(15)*YA-E(16)*T4,               0065
    2          E(14)+E(15)*X+E(16)*T3+T6+D(15)*YA+D(16)*T4)               0066
     GO TO 52                                                           0067
50   T3=X*X-Y*Y                                                         0068
     T4=2.*X*YA                                                         0069
     T5=X*T3-YA*T4                                                      0070
     T6=X*T4+YA*T3                                                      0071
     T7=X*T5-YA*T6                                                      0072
     T8=X*T6+YA*T5                                                      0073
     T9=X*T7-YA*T8                                                      0074
     T10=X*T8+YA*T7                                                     0075
     UC=CMPLX(D(1)+D(2)*X+D(3)*T3+D(4)*T5+D(5)*T7+T9-(E(2)*YA+E(3)*T4   0076
    2+E(4)*T6+E(5)*T8),E(1)+E(2)*X+E(3)*T3+E(4)*T5+E(5)*T7+T10+         0077
    3(D(2)*YA+D(3)*T4+D(4)*T6+D(5)*T8))                                 0078
     VC=CMPLX(D(6)+D(7)*X+D(8)*T3+D(9)*T5+D(10)*T7+T9-(E(7)*YA+E(8)*T4  0079
    2+E(9)*T6+E(10)*T8),E(6)+E(7)*X+E(8)*T3+E(9)*T5+E(10)*T7+T10+       0080
    3(D(7)*YA+D(8)*T4+D(9)*T6+D(10)*T8))                                0081
52   EC=UC/VC                                                           0082
     S=EC/CMPLX(X,YA)                                                   0083
54   EX=EXP(-X)                                                         0084
     T=EX*CMPLX(COS(YA),-SIN(YA))                                       0085
     E15=S*T                                                            0086
56   IF(Y.LT.0.)E15=CONJG(E15)                                          0087
     GO TO 90                                                           0088
80   E15=.409319/(Z+.193044)+.421831/(Z+1.02666)+.147126/(Z+2.56788)+   0089
    2.206335E-1/(Z+4.90035)+.107401E-2/(Z+8.18215)+.158654E-4/(Z+       0090
    312.7342)+.317031E-7/(Z+19.3957)                                    0091
     E15=E15*CEXP(-Z)                                                   0092
90   IF(JIM.EQ.1)W12=E15                                                0093
100  Z=V2                                                               0094
     Z=V2/V1                                                            0095
     TH=ATAN2(AIMAG(Z),REAL(Z))-ATAN2(AIMAG(V2),REAL(V2))               0096
    2+ATAN2(AIMAG(V1),REAL(V1))                                         0097
     AB=ABS(TH)                                                         0098
     IF(AB.LT.1.)TH=.0                                                  0099
     IF(TH.GT.1.)TH=6.2831853                                           0100
     IF(TH.LT.-1.)TH=-6.2831853                                         0101
     W12=W12-E15+CMPLX(.0,TH)                                           0102
     RETURN                                                             0103
     END                                                                0104
```

Fig. 14b. Subroutine EXPJ

The exponential integral $E_1(z)$ is defined in Reference 3. To generate W12, subroutine EXPJ calculates $E_1(V1)$, subtracts $E_1(V2)$ and adds $j2n\pi$. The term $j2n\pi$ is determined by the requirement that W12 vanish in the limit as V1 approaches V2. The integer n may assume values of -1, 0 or +1. If the integration path does not cross the negative real axis in the v plane, n is zero. The term $j2n\pi$ is calculated below statement 100.

APPENDIX 8. Subroutine GANT1

Subroutine GANT1, listed in Fig. 15, considers the wire structure as an antenna. In the input data, VG(J) is the voltage of a generator at point IA(J) of segment J. VG(JJ) is the voltage of a generator at point IB(J) of segment J. The DO LOOP ending with statement 50 uses the delta-gap model to determine the excitation voltages CJ(I) for all the dipole modes. These are also stored temporarily in CG(I). Then subroutine SQROT is called to obtain a solution of the simultaneous linear equations. SQROT stores the solution (the loop currents) in CJ(I).

In the DO LOOP ending at statement 80, the complex power input is calculated and stored in Y11. GG denotes the time-average power input and is the real part of Y11. If the antenna has only one voltage generator (with unit voltage and zero phase angle), then Y11 also denotes the antenna admittance and Z11 is the antenna impedance at that port.

Subroutine RITE is called to make the transformation from the loop currents CJ(I) to the branch currents CG(J). If IWR is a positive integer, RITE will write out the list of branch currents.

Finally, GANT1 calculates the radiation efficiency EFF. PIN denotes the time-average power input. Subroutine GDISS is called to obtain the time-average power dissipated. DISS is the total power dissipated in the lumped loads and the imperfectly-conducting wire. PRAD is the time-average power radiated, defined by the difference between PIN and DISS. If the antenna has perfect conductivity and purely reactive loads, the radiation efficiency is considered to be 100 per cent.

APPENDIX 9. Subroutine SQROT

Subroutine SQROT is listed in Fig. 16. This subroutine considers the matrix equation ZI = V which represents a system of simultaneous linear equations. If the square matrix Z is symmetric, SQROT is useful for obtaining the solution I with V given. NEQ denotes the number of simultaneous equations and the size of the matrix Z.

On entry to SQROT, S is the excitation column V. On exit, the solution I is stored in S. Let Z(I,J) denote the symmetric square

```
      SUBROUTINE GANT1(IA,IB,INM,IWR,I1,I2,I3,I12,JA,JB,MD,N,ND,NM,AM      0001
     2,C,CJ,CG,CMM,D,EFF,GAM,GG,CGD,SGD,VG,Y11,Z11,ZLD,ZS)                0002
      COMPLEX C(1),CJ(1),CGD(1),SGD(1),VG(1),ZLD(1),Y11,Z11,ZS,GAM,CG(1)  0003
      DIMENSION D(1),IA(1),IB(1),JA(1),JB(1)                              0004
      DIMENSION I1(1),I2(1),I3(1),MD(INM,4),ND(1)                         0005
2     FORMAT(1X,1I5,8F10.2)                                               0006
5     FORMAT(1H0)                                                         0007
      DO 50 I=1,N                                                         0008
      CJ(I)=(.0,.0)                                                       0009
      K=JA(I)                                                             0010
      DO 40 KK=1,2                                                        0011
      KA=IA(K)                                                            0012
      KB=IB(K)                                                            0013
      JJ=K                                                                0014
      FI=1.                                                               0015
      IF(KB.EQ.I2(I))GO TO 36                                             0016
      IF(KB.EQ.I1(I))FI=-1.                                               0017
      CJ(I)=CJ(I)+FI*VG(JJ)                                               0018
      GO TO 40                                                            0019
36    IF(KA.EQ.I3(I))FI=-1.                                               0020
      JJ=K+NM                                                             0021
      CJ(I)=CJ(I)+FI*VG(JJ)                                               0022
40    K=JB(I)                                                             0023
50    CONTINUE                                                            0024
      DO 55 I=1,N                                                         0025
55    CG(I)=CJ(I)                                                         0026
      CALL SQROT(C,CJ,0,I12,N)                                            0027
      I12=2                                                               0028
      Y11=(.0,.0)                                                         0029
      DO 80 I=1,N                                                         0030
80    Y11=Y11+CJ(I)*CONJG(CG(I))                                          0031
      CALL RITE(IA,IB,INM,IWR,I1,I2,I3,MD,ND,NM,CJ,CG)                    0032
      GG=REAL(Y11)                                                        0033
      Z11=1./Y11                                                          0034
      PIN=GG                                                              0035
      CALL       GDISS(AM,CG,CMM,D,DISS,GAM,NM,SGD,ZLD,ZS)                0036
      PRAD=PIN-DISS                                                       0037
      EFF=100.*PRAD/PIN                                                   0038
      RETURN                                                              0039
      END                                                                 0040
```

Fig. 15. Subroutine GANT1

```
      SUBROUTINE SQROT(C,S,IWR,I12,NEQ)                                 0001
      COMPLEX C(1),S(1),SS                                              0002
2     FORMAT(1X,1I5,1F10.3,1F15.7,1F10.0,2F15.6)                        0003
3     FORMAT(1H0)                                                       0004
      N=NEQ                                                             0005
      IF(I12.EQ.2)GO TO 20                                              0006
      C(1)=CSQRT(C(1))                                                  0007
      DO 4 K=2,N                                                        0008
4     C(K)=C(K)/C(1)                                                    0009
      DO 10 I=2,N                                                       0010
      IMO=I-1                                                           0011
      IPO=I+1                                                           0012
      ID=(I-1)*N-(I*I-I)/2                                              0013
      II=ID+I                                                           0014
      DO 5 L=1,IMO                                                      0015
      LI=(L-1)*N-(L*L-L)/2+I                                            0016
5     C(II)=C(II)-C(LI)*C(LI)                                           0017
      C(II)=CSQRT(C(II))                                                0018
      IF(IPO.GT.N)GO TO 10                                              0019
      DO 8 J=IPO,N                                                      0020
      IJ=ID+J                                                           0021
      DO 6 M=1,IMO                                                      0022
      MD=(M-1)*N-(M*M-M)/2                                              0023
      MI=MD+I                                                           0024
      MJ=MD+J                                                           0025
    6 C(IJ)=C(IJ)-C(MJ)*C(MI)                                           0026
8     C(IJ)=C(IJ)/C(II)                                                 0027
10    CONTINUE                                                          0028
20    S(1)=S(1)/C(1)                                                    0029
      DO 30 I=2,N                                                       0030
      IMO=I-1                                                           0031
      DO 25 L=1,IMO                                                     0032
      LI=(L-1)*N-(L*L-L)/2+I                                            0033
25    S(I)=S(I)-C(LI)*S(L)                                              0034
      II=(I-1)*N-(I*I-I)/2+I                                            0035
30    S(I)=S(I)/C(II)                                                   0036
      NN=((N+1)*N)/2                                                    0037
      S(N)=S(N)/C(NN)                                                   0038
      NMO=N-1                                                           0039
      DO 40 I=1,NMO                                                     0040
      K=N-I                                                             0041
      KPO=K+1                                                           0042
      KD=(K-1)*N-(K*K-K)/2                                              0043
      DO 35 L=KPO,N                                                     0044
      KL=KD+L                                                           0045
35    S(K)=S(K)-C(KL)*S(L)                                              0046
      KK=KD+K                                                           0047
40    S(K)=S(K)/C(KK)                                                   0048
      IF(IWR.LE.0) GO TO 100                                            0049
      CNOR=.0                                                           0050
      DO 50 I=1,N                                                       0051
      SA=CABS(S(I))                                                     0052
50    IF(SA.GT.CNOR)CNOR=SA                                             0053
      IF(CNOR.LE.0.)CNOR=1.                                             0054
      DO 60 I=1,N                                                       0055
      SS=S(I)                                                           0056
      SA=CABS(SS)                                                       0057
      SNOR=SA/CNOR                                                      0058
      PH=.0                                                             0059
      IF(SA.GT.0.)PH=57.29578*ATAN2(AIMAG(SS),REAL(SS))                 0060
60    WRITE(6,2)I,SNOR,SA,PH,SS                                         0061
      WRITE(6,3)                                                        0062
100   RETURN                                                            0063
      END                                                               0064
```

Fig. 16. Subroutine SQROT

matrix. On entry to SQROT, the upper-right triangular portion of Z(I,J) is stored by rows in C(K) with

(4) K = (I - 1)*NEQ - (I*I - I) / 2 + J

If I12 = 1, SQROT will transform the symmetric matrix into the auxiliary matrix (implicit inverse), store the result in C(K) and use the auxiliary matrix to solve the simultaneous equations. If I12 = 2, this indicates that C(K) already contains the auxiliary matrix.

The transformation from the symmetric matrix to the auxiliary matrix is programmed above statement 10, and the solution of the simultaneous equations is programmed in statements 20 to 40. If IWR is positive, the program below statement 40 will write the solution.

SQROT uses the square root method described in Reference 4. The original symmetric matrix Z and the upper triangular auxiliary matrix A are related by

(5) Z = A' A

where A' is the transpose of A.

In the thin-wire application, SQROT must be called with I12 = 1 before it is called with I12 = 2. With a large matrix, the execution time in SQROT is much smaller with I12 = 2 than with I12 = 1.

APPENDIX 10. Subroutine RITE

Subroutine RITE is listed in Fig. 17. Given the list of loop currents CJ(I), this subroutine generates a list of branch currents CG(J). CG(J) and CG(JJ) denote the currents at IA(J) and IB(J), respectively, on the wire segment J, where JJ = J + NM. If IWR is a positive integer, the program below statement 110 writes a list of the branch currents. The symbols in this list are defined as follows:

K	the segment number
ACJ	normalized current magnitude at IA(K)
BCJ	normalized current magnitude at IB(K)
PA	phase of current at IA(K)
PB	phase of current at IB(K)
CJA	complex current at IA(K)
CJB	complex current at IB(K)

The phase angles PA and PB are in degrees. Even if IWR is negative, RITE generates the branch-current list for use in subroutine GDISS.

```
      SUBROUTINE RITE(IA,IB,INM,IWR,I1,I2,I3,MD,ND,NM,CJ,CG)                0001
      COMPLEX CJ(1),CG(1),CJA,CJB                                           0002
      DIMENSION IA(1),IB(1),I1(1),I2(1),I3(1),MD(INM,4),ND(1)               0003
2     FORMAT(1X,1I5,2F10.3,2F10.0,4F15.6)                                   0004
5     FORMAT(1H0)                                                           0005
      AMAX=.0                                                               0006
      DO 100 K=1,NM                                                         0007
      KA=IA(K)                                                              0008
      KB=IB(K)                                                              0009
      CJA=(.0,.0)                                                           0010
      CJB=(.0,.0)                                                           0011
      NDK=ND(K)                                                             0012
      DO 40 II=1,NDK                                                        0013
      I=MD(K,II)                                                            0014
      FI=1.                                                                 0015
      IF(KB.EQ.I2(I))GO TO 36                                               0016
      IF(KB.EQ.I1(I))FI=-1.                                                 0017
      CJA=CJA+FI*CJ(I)                                                      0018
      GO TO 40                                                              0019
36    IF(KA.EQ.I3(I))FI=-1.                                                 0020
      CJB=CJB+FI*CJ(I)                                                      0021
40    CONTINUE                                                              0022
      CG(K)=CJA                                                             0023
      KK=K+NM                                                               0024
      CG(KK)=CJB                                                            0025
      ACJ=CABS(CJA)                                                         0026
      BCJ=CABS(CJB)                                                         0027
      IF(ACJ.GT.AMAX)AMAX=ACJ                                               0028
      IF(BCJ.GT.AMAX)AMAX=BCJ                                               0029
100   CONTINUE                                                              0030
      IF(IWR.GT.0)GO TO 110                                                 0031
      RETURN                                                                0032
110   IF(AMAX.LE.0.)AMAX=1.                                                 0033
      DO 200 K=1,NM                                                         0034
      CJA=CG(K)                                                             0035
      KK=K+NM                                                               0036
      CJB=CG(KK)                                                            0037
      ACJ=CABS(CJA)/AMAX                                                    0038
      BCJ=CABS(CJB)/AMAX                                                    0039
      PA=57.29578*ATAN2(AIMAG(CJA),REAL(CJA))                               0040
      PB=57.29578*ATAN2(AIMAG(CJB),REAL(CJB))                               0041
200   WRITE(6,2)K,ACJ,BCJ,PA,PB,CJA,CJB                                     0042
      WRITE(6,5)                                                            0043
      RETURN                                                                0044
      END                                                                   0045
```

Fig. 17. Subroutine RITE

APPENDIX 11. Subroutine GDISS

Subroutine GDISS is listed in Fig. 18. This subroutine uses Eq. 50 of Reference 1 to calculate the time-average power dissipated in the imperfectly conducting wire. This is accomplished in the DO LOOP terminating at statement 100. The power dissipated in the lumped loads is calculated in the DO LOOP terminating with statement 140. DISS denotes the time-average power dissipated in the wire and the loads.

APPENDIX 12. Subroutine GNFLD

Subroutine GNFLD, listed in Fig. 19, inputs the loop currents CJ(I), calls GNF for the near-zone field of each wire segment, and sums over all the segments to obtain the near-zone field of the wire antenna or the near-zone scattered field of the wire scatterer. EX, EY and EZ denote the cartesian components of this field at the observation point (XP,YP,ZP). This calculated field does not include the incident fields of the magnetic frills or loops associated with generators on the antenna. It also does not include the radiation from the polarization currents in the dielectric insulation.

This subroutine could be simplified and speeded by inputting the branch currents CG(J) instead of the loop currents CJ(I). However, this would increase the storage requirements because the far-field subroutine GFFLD would have to store the branch currents induced by the phi-polarized and theta-polarized incident waves.

APPENDIX 13. Subroutine GNF

Subroutine GNF, listed in Fig. 20, uses Eqs. 75 and 76 of Reference 1 to calculate the near-zone electric field of a sinusoidal electric monopole with endpoints at (XA,YA,ZA) and (XB,YB,ZB). The observation point is at (X,Y,Z). EX1, EY1 and EZ1 are the components of the field generated by the mode with unit current at (XA,YA,ZA). EX2, EY2 and EZ2 denote the field generated by the mode with unit current at (XB,YB,ZB). GNF is similar to GGS, and Appendix 5 defines many of the symbols used in both subroutines.

APPENDIX 14. Subroutine GFFLD

The far-field subroutine GFFLD, listed in Fig. 21, is discussed in section II. In antenna gain calculations with INC = 0, the loop currents CJ(I) are employed by GFFLD to calculate the far-zone field. The field of each segment is obtained by calling GFF, and a summation over all the segments yields the field of the antenna.

In a bistatic scattering situation with INC = 2, the input data include the loop currents EP and ET induced by phi-polarized and theta-polarized incident waves. These currents were calculated by GFFLD in a

```
      SUBROUTINE GDISS(AM,CG,CMM,D,DISS,GAM,NM,SGD,ZLD,ZS)                0001
      COMPLEX CG(1),SGD(1),ZLD(1),CJA,CJB,GAM,ZS                          0002
      DIMENSION D(1)                                                      0003
      DATA PI/3.14159/                                                    0004
      DISS=.0                                                             0005
      IF(CMM.LE.0.)GO TO 120                                              0006
      ALPH=REAL(GAM)                                                      0007
      BETA=AIMAG(GAM)                                                     0008
      RH=REAL(ZS)/(4.*PI*AM)                                              0009
      DO 100 K=1,NM                                                       0010
      DK=D(K)                                                             0011
      DEN=CABS(SGD(K))**2                                                 0012
      EAD=EXP(ALPH*DK)                                                    0013
      CAD=(EAD+1./EAD)/2.                                                 0014
      CBD=COS(BETA*DK)                                                    0015
      SAD=DK                                                              0016
      IF(ALPH.NE.0.)SAD=(EAD-1./EAD)/(2.*ALPH)                            0017
      SBD=DK                                                              0018
      IF(BETA.NE.0.)SBD=SIN(BETA*DK)/BETA                                 0019
      FA=RH*(SAD*CAD-SBD*CBD)/DEN                                         0020
      FB=2.*RH*(CAD*SBD-SAD*CBD)/DEN                                      0021
      CJA=CG(K)                                                           0022
      L=K+NM                                                              0023
      CJB=CG(L)                                                           0024
  100 DISS=DISS+FA*(CABS(CJA)**2+CABS(CJB)**2)                            0025
     2+FB*(REAL(CJA)*REAL(CJB)+AIMAG(CJA)*AIMAG(CJB))                     0026
  120 DO 140 J=1,NM                                                       0027
      K=J+NM                                                              0028
  140 DISS=DISS+REAL(ZLD(J))*(CABS(CG(J))**2)                             0029
     2+REAL(ZLD(K))*(CABS(CG(K))**2)                                      0030
      RETURN                                                              0031
      END                                                                 0032
```

Fig. 18. Subroutine GDISS

```
      SUBROUTINE GNFLD(IA,IB,INM,I1,I2,I3,MD,N,ND,NM,AM,CGD,SGD,ETA,GAM      0001
     2,CJ,D,X,Y,Z,XP,YP,ZP,EX,EY,EZ)                                        0002
      COMPLEX EX,EY,EZ,EX1,EY1,EZ1,EX2,EY2,EZ2,ETA,GAM                       0003
      COMPLEX CJ(1),CGD(1),SGD(1)                                            0004
      DIMENSION IA(1),IB(1),I1(1),I2(1),I3(1),D(1),X(1),Y(1),Z(1)            0005
      DIMENSION MD(INM,4),ND(1)                                              0006
      DATA PI,TP/3.14159,6.28318/                                            0007
      EX=(.0,.0)                                                             0008
      EY=(.0,.0)                                                             0009
      EZ=(.0,.0)                                                             0010
      DO 140 K=1,NM                                                          0011
      KA=IA(K)                                                               0012
      KB=IB(K)                                                               0013
      CALL GNF(X(KA),Y(KA),Z(KA),X(KB),Y(KB),Z(KB),XP,YP,ZP,AM,D(K)          0014
     2,CGD(K),SGD(K),ETA,GAM,EX1,EY1,EZ1,EX2,EY2,EZ2)                        0015
      NDK=ND(K)                                                              0016
      DO 140 II=1,NDK                                                        0017
      I=MD(K,II)                                                             0018
      FI=1.                                                                  0019
      IF(KB.EQ.I2(I))GO TO 136                                               0020
      IF(KB.EQ.I1(I))FI=-1.                                                  0021
      EX=EX+FI*EX1*CJ(I)                                                     0022
      EY=EY+FI*EY1*CJ(I)                                                     0023
      EZ=EZ+FI*EZ1*CJ(I)                                                     0024
      GO TO 140                                                              0025
  136 IF(KA.EQ.I3(I))FI=-1.                                                  0026
      EX=EX+FI*EX2*CJ(I)                                                     0027
      EY=EY+FI*EY2*CJ(I)                                                     0028
      EZ=EZ+FI*EZ2*CJ(I)                                                     0029
  140 CONTINUE                                                               0030
      RETURN                                                                 0031
      END                                                                    0032
```

Fig. 19. Subroutine GNFLD

```
      SUBROUTINE GNF(XA,YA,ZA,XB,YB,ZB,X,Y,Z,AM,DS,CGDS,SGDS,ETA,GAM      0001
     2,EX1,EY1,EZ1,EX2,EY2,EZ2)                                           0002
      COMPLEX EJA,EJB,EJ1,EJ2,ER1,ER2,ES1,ES2,SGDS,GAM,CST,CGDS,ETA       0003
      COMPLEX EX1,EY1,EZ1,EX2,EY2,EZ2                                     0004
      DATA PI/3.14159/                                                    0005
      CAS=(XB-XA)/DS                                                      0006
      CBS=(YB-YA)/DS                                                      0007
      CGS=(ZB-ZA)/DS                                                      0008
      SZ=(X-XA)*CAS+(Y-YA)*CBS+(Z-ZA)*CGS                                 0009
      ZZ1=SZ                                                              0010
      ZZ2=SZ-DS                                                           0011
      XXZ=X-XA-SZ*CAS                                                     0012
      YYZ=Y-YA-SZ*CBS                                                     0013
      ZZZ=Z-ZA-SZ*CGS                                                     0014
      RS=XXZ**2+YYZ**2+ZZZ**2                                             0015
      R1=SQRT(RS+ZZ1**2)                                                  0016
      EJA=CEXP(-GAM*R1)                                                   0017
      EJ1=EJA/R1                                                          0018
      R2=SQRT(RS+ZZ2**2)                                                  0019
      EJB=CEXP(-GAM*R2)                                                   0020
      EJ2=EJB/R2                                                          0021
      ES1=EJ2-EJ1*CGDS                                                    0022
      ES2=EJ1-EJ2*CGDS                                                    0023
      ER1=(.0,.0)                                                         0024
      ER2=(.0,.0)                                                         0025
      AMS=AM*AM                                                           0026
      IF(RS.LT.AMS)GO TO 80                                               0027
      CTH1=ZZ1/R1                                                         0028
      CTH2=ZZ2/R2                                                         0029
      ER1=( EJA*SGDS+EJA*CGDS*CTH1-EJB*CTH2)/RS                           0030
      ER2=(-EJB*SGDS+EJB*CGDS*CTH2-EJA*CTH1)/RS                           0031
   80 CST=ETA/(4.*PI*SGDS)                                                0032
      EX1=CST*(ES1*CAS+ER1*XXZ)                                           0033
      EY1=CST*(ES1*CBS+ER1*YYZ)                                           0034
      EZ1=CST*(ES1*CGS+ER1*ZZZ)                                           0035
      EX2=CST*(ES2*CAS+ER2*XXZ)                                           0036
      EY2=CST*(ES2*CBS+ER2*YYZ)                                           0037
      EZ2=CST*(ES2*CGS+ER2*ZZZ)                                           0038
      RETURN                                                              0039
      END                                                                 0040
```

Fig. 20. Subroutine GNF

```
      SUBROUTINE GFFLD(IA,IB,INC,INM,IWR,I1,I2,I3,I12,MD,N,ND,NM,AM      0001
     2,ACSP,ACST,C,CGD,CG,CJ,CMM,D,ECSP,ECST,EP,ET,EPP,ETT,EPPS,EPTS    0002
     3,ETPS,ETTS,GG,GPP,GTT,PH,SGD,SCSP,SCST,SPPM,SPTM,STPM,STTM,TH     0003
     4,X,Y,Z,ZLD,ZS,ETA,GAM)                                            0004
      COMPLEX CJI,ET1,ET2,EP1,EP2,EPPS,ETTS,EPTS,ETPS,ZS,VP,VT          0005
      COMPLEX C(1),CJ(1),EP(1),ET(1),EPP(1),ETT(1),ZLD(1)               0006
      COMPLEX ETA,GAM,CGD(1),SGD(1),CG(1)                               0007
      DIMENSION IA(1),IB(1),I1(1),I2(1),I3(1),ND(1),MD(INM,4)           0008
      DIMENSION D(1),X(1),Y(1),Z(1)                                     0009
      DATA PI,TP/3.14159,6.28318/                                       0010
      CJI=-4.*PI/(ETA*GAM)                                              0011
      GGG=REAL(1./ETA)                                                  0012
      THR=.0174533*TH                                                   0013
      CTH=COS(THR)                                                      0014
      STH=SIN(THR)                                                      0015
      PHR=.0174533*PH                                                   0016
      CPH=COS(PHR)                                                      0017
      SPH=SIN(PHR)                                                      0018
      DO 130 I=1,N                                                      0019
      ETT(I)=(.0,.0)                                                    0020
  130 EPP(I)=(.0,.0)                                                    0021
      DO 140 K=1,NM                                                     0022
      KA=IA(K)                                                          0023
      KB=IB(K)                                                          0024
      CALL GFF(X(KA),Y(KA),Z(KA),X(KB),Y(KB),Z(KB),D(K)                 0025
     2,CGD(K),SGD(K),CTH,STH,CPH,SPH,GAM,ETA,ET1,ET2,EP1,EP2)           0026
      NDK=ND(K)                                                         0027
      DO 140 II=1,NDK                                                   0028
      I=MD(K,II)                                                        0029
      FI=1.                                                             0030
      IF(KB.EQ.I2(I))GO TO 136                                          0031
      IF(KB.EQ.I1(I))FI=-1.                                             0032
      EPP(I)=EPP(I)+FI*EP1                                              0033
      ETT(I)=ETT(I)+FI*ET1                                              0034
      GO TO 140                                                         0035
  136 IF(KA.EQ.I3(I))FI=-1.                                             0036
      EPP(I)=EPP(I)+FI*EP2                                              0037
      ETT(I)=ETT(I)+FI*ET2                                              0038
  140 CONTINUE                                                          0039
      EPPS=(.0,.0)                                                      0040
      ETTS=(.0,.0)                                                      0041
      IF(INC.EQ.0)GO TO 200                                             0042
      IF(INC.EQ.2)GO TO 170                                             0043
      DO 150 I=1,N                                                      0044
      ET(I)=ETT(I)*CJI                                                  0045
  150 EP(I)=EPP(I)*CJI                                                  0046
      CALL SQROT(C,EP,0,I12,N)                                          0047
      I12=2                                                             0048
      CALL SQROT(C,ET,0,I12,N)                                          0049
      CALL RITE(IA,IB,INM,IWR,I1,I2,I3,MD,ND,NM,EP,CG)                  0050
      CALL GDISS(AM,CG,CMM,D,PDIS,GAM,NM,SGD,ZLD,ZS)                    0051
      CALL RITE(IA,IB,INM,IWR,I1,I2,I3,MD,ND,NM,ET,CG)                  0052
      CALL GDISS(AM,CG,CMM,D,TDIS,GAM,NM,SGD,ZLD,ZS)                    0053
      ACSP=PDIS/GGG                                                     0054
      ACST=TDIS/GGG                                                     0055
      PIN=.0                                                            0056
      TIN=.0                                                            0057
      DO 164 I=1,N                                                      0058
      VP=CJI*EPP(I)                                                     0059
      VT=CJI*ETT(I)                                                     0060
      PIN=PIN+REAL(VP*CONJG(EP(I)))                                     0061
  164 TIN=TIN+REAL(VT*CONJG(ET(I)))                                     0062
```

Fig. 21a. Subroutine GFFLD

```
      ECSP=PIN/GGG                                                      0063
      ECST=TIN/GGG                                                      0064
      SCSP=ECSP-ACSP                                                    0065
      SCST=ECST-ACST                                                    0066
170   EPTS=(.0,.0)                                                      0067
      ETPS=(.0,.0)                                                      0068
      DO 180 I=1,N                                                      0069
      EPPS=EPPS+EP(I)*EPP(I)                                            0070
      EPTS=EPTS+EP(I)*ETT(I)                                            0071
      ETTS=ETTS+ET(I)*ETT(I)                                            0072
180   ETPS=ETPS+ET(I)*EPP(I)                                            0073
      SPPM=2.*TP*(CABS(EPPS)**2)                                        0074
      SPTM=2.*TP*(CABS(EPTS)**2)                                        0075
      STPM=2.*TP*(CABS(ETPS)**2)                                        0076
      STTM=2.*TP*(CABS(ETTS)**2)                                        0077
      RETURN                                                            0078
200   DO 260 I=1,N                                                      0079
      ETTS=ETTS+CJ(I)*ETT(I)                                            0080
260   EPPS=EPPS+CJ(I)*EPP(I)                                            0081
      APP=CABS(EPPS)                                                    0082
      ATT=CABS(ETTS)                                                    0083
      GPP=4.*PI*APP*APP*GGG/GG                                          0084
      GTT=4.*PI*ATT*ATT*GGG/GG                                          0085
      RETURN                                                            0086
      END                                                               0087
```

Fig. 21b. Subroutine GFFLD

previous call for the backscattering situation with INC = 1. Thus, a bistatic call must be preceded by a backscatter call.

EPP(I) and ETT(I) denote the phi-polarized and theta-polarized far-zone fields of dipole mode I with unit terminal current. In a backscattering situation, the excitation voltages EP(I) and ET(I) are obtained by multiplying EPP and ETT by the constant CJI. (See Eqs. 38, 39 and 40 in Reference 1.) Then calls are made to SQROT which stores the solution (the induced loop currents) in EP(I) and ET(I). RITE is called for the branch currents CG(J), and GDISS is called for the time-average power dissipated in the imperfectly conducting wire and the lumped loads. This power is denoted PDIS and TDIS for phi-polarized and theta-polarized incident waves, respectively.

In scattering problems, the incident plane wave has unit electric field intensity at the coordinate origin. GGG denotes the time-average power density of the incident wave at the origin. ACSP and ACST denote the absorption cross sections for the phi and theta polarizations.

PIN and TIN denote the time-average power input to the wire structure, delivered by the equivalent voltage generators VP and VT at the terminals. PIN and TIN apply for the phi and theta polarizations, respectively. The time-average power input is regarded as the sum of the time-average power dissipated (in the wire and the lumped loads) and the time-average power radiated or scattered by the wire. ECSP and ECST denote the extinction cross sections and SCSP and SCST are the scattering cross sections.

The distant field is calculated in the DO LOOP ending with statement 180 for scattering situations, and in the DO LOOP ending with statement 260 for the antenna situation. In these fields, the range dependence is suppressed as in Eq. (1).

The radar cross sections (echo areas) SPPM, SPTM, STPM and STTM are defined as in Eq. 72 of Reference 1 with the incident power density (S_i or GGG) evaluated at the coordinate origin. The user selects the location of the origin when supplying the input data for the coordinates of all the points on the wire.

For an antenna, the following definition is employed for the power gain:

$$(6) \qquad G_p(\theta,\phi) = \lim_{r \to \infty} \; 4\pi r^2 \, e^{2\alpha r} \, S(r,\theta,\phi) \; / \; P_i$$

where P_i (or GG in the program) denotes the time-average power input and $S(r,\theta,\phi)$ is the time-average power density in the radiated field. For an antenna in a lossless medium, α vanishes and Eq. (6) reduces to the standard definition of power gain. Without the factor $e^{2\alpha r}$ in Eq. (6), the power gain would vanish for a finite antenna in a conducting medium. GPP and GTT denote the power gains associated with the phi-polarized and theta-polarized components of the field, respectively.

APPENDIX 15. Subroutine GFF

Subroutine GFF, listed in Fig. 22, uses the equations in Appendix 2 of Reference 1 to calculate the far-zone field of a sinusoidal electric monopole. The monopole has endpoints (XA,YA,ZA) and (XB,YB,ZB). EP1 and ET1 denote E_ϕ and E_θ for the mode with unit current at (XA,YA,ZA). EP2 and ET2 denote the fields for the mode with unit current at (XB,YB,ZB). The range dependence is suppressed as in Eq. (1). The far field vanishes in the endfire direction where GK = 0.

```
      SUBROUTINE GFF(XA,YA,ZA,XB,YB,ZB,D,                               0001
     2CGD,SGD,CTH,STH,CPH,SPH,                                          0002
     2GAM,ETA,ET1,ET2,EP1,EP2)                                          0003
      COMPLEX ET1,ET2,EP1,EP2,GAM,ETA                                   0004
      COMPLEX GD,CGD,SGD,EGD                                            0005
      COMPLEX EGFA,EGFB,EGGD,ESA,ESB                                    0006
      COMPLEX CST                                                       0007
      FP=12.56637                                                       0008
      XAB=XB-XA                                                         0009
      YAB=YB-YA                                                         0010
      ZAB=ZB-ZA                                                         0011
      CA=XAB/D                                                          0012
      CB=YAB/D                                                          0013
      CG=ZAB/D                                                          0014
      G=(CA*CPH+CB*SPH)*STH+CG*CTH                                      0015
      GK=1.-G*G                                                         0016
      ET1=(.0,.0)                                                       0017
      ET2=(.0,.0)                                                       0018
      EP1=(.0,.0)                                                       0019
      EP2=(.0,.0)                                                       0020
      IF(GK.LT..001)GO TO 200                                           0021
      FA=(XA*CPH+YA*SPH)*STH+ZA*CTH                                     0022
      FB=(XB*CPH+YB*SPH)*STH+ZB*CTH                                     0023
      EGFA=CEXP(GAM*FA)                                                 0024
      EGFB=CEXP(GAM*FB)                                                 0025
      EGGD=CEXP(GAM*G*D)                                                0026
      CST=ETA/(GK*SGD*FP)                                               0027
      ESA=CST*EGFA*(EGGD-G*SGD-CGD)                                     0028
      ESB=CST*EGFB*(1./EGGD+G*SGD-CGD)                                  0029
      T=(CA*CPH+CB*SPH)*CTH-CG*STH                                      0030
      P=-CA*SPH+CB*CPH                                                  0031
      ET1=T*ESA                                                         0032
      ET2=T*ESB                                                         0033
      EP1=P*ESA                                                         0034
      EP2=P*ESB                                                         0035
  200 CONTINUE                                                          0036
      RETURN                                                            0037
      END                                                               0038
```

Fig. 22. Subroutine GFF