5322-5

NRL Report 7826

Computer Analysis of Conformal Phased Arrays

J. K. Hsiao and J. B. L. Rao

Search Radar Branch
Radar Division

November 15, 1974

NAVAL RESEARCH LABORATORY
Washington, D.C.

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER NRL Report 7826	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) COMPUTER ANALYSIS OF CONFORMAL PHASED ARRAYS		5. TYPE OF REPORT & PERIOD COVERED Interim report on a continuing NRL problem.
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) J.K. Hsiao and J.B.L. Rao		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Research Laboratory Washington, D.C. 20375		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS NRL Problem R02-24 Project WF 099-05-020
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Air Systems Command Arlington, VA 22217		12. REPORT DATE November 15, 1974
		13. NUMBER OF PAGES 41
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release, distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Conformal array
Phased array
Antenna pattern
Conformal-array computer program

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A systematic method of analyzing conformal arrays on a general surface is described. A computer program is included in this report along with the necessary subroutines. By using proper input data cards and the subroutines, the program can be used for the arrays on a general conformal surface or well-defined surface like that of a circular cylinder, circular torus, cone, or a plane. The computer output includes a printout and a plot of the radiation patterns. Crosspolarization, if any, will be plotted on the same curve.

DD FORM 1473 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

CONTENTS

February 28, 1975

To: Distribution list for NRL Report 7826, "Computer Analysis of Conformal Phased Arrays," by J.K. Hsiao and J.B.L. Rao, November 15, 1974.

The following changes should be made:

In all mathematical expressions appearing on pages 2, 3, 4, 6, 7, 8, and 9, substitute a single prime (′) or double prime (″) symbol in lieu of the single apostrophe (') or double apostrophe (").

COMPUTER ANALYSIS OF CONFORMAL PHASED ARRAYS

INTRODUCTION

In recent years, considerable interest has been shown in conformal arrays. They are presently under consideration for the range instrumentation aircraft at the Pacific Missile Range. Predicting the far-field pattern and polarization of conformal arrays is complicated, due to the arbitrariness of a conformal surface and the different orientations of the radiating elements. In this report a systematic approach of analyzing conformal arrays is described which can be conveniently implemented on a digital computer to render a numerical solution with the assumption that the location, orientation, and the radiation pattern of each element in the conformal array is known. A computer program is written to implement this method and is included in this report. The program is formulated in such a general way that the array size, location, element spacing and the curvature of the conformal surface can be varied with a minimum of change in the program. By including proper input data cards and the subroutines, the program can be used for the arrays on a general conformal surface or well-defined surface like that of a circular cylinder, circular torus, cone, or a plane. In the case of a general conformal surface, the position and orientation of each element are supplied as input data to the program. The position of an element is specified by three components in the rectangular coordinate system, referenced, to a point on the array surface. The element orientation is specified by three coordinate rotation angles. When specialized to a circular cylinder, circular torus, or plane, only the number of elements, the array aperture, and the radius of curvature in x and y directions need be specified. For a plane, the radius of curvature is specified as zero. For a cone, the number of rows, the base radius of curvature, the arc angle, the cone angle, the spacing between the rows and the inter-element spacing along the base need be specified. In all cases the element pattern could be specified as a horizontal or a vertical dipole. In addition an exact phase or an approximate phase (row-column phasing) can be selected for scanning the beam for a circular cylinder. Array patterns can be obtained for any polar-angle cut over an angular range of $\pm 90°$. The computer gives a printed output and a plot of the radiation patterns. Cross-polarization, if any, will be plotted on the same curve.

FAR-FIELD PATTERN

Given a current source, the magnetic-potential vector $\mathbf{A}$ satisfies the vector Helmholtz equation

$$\nabla^2 \mathbf{A} + k^2 \mathbf{A} = \mathbf{J}, \tag{1}$$

Note: Manuscript submitted September 16, 1974.

(Substitute a single prime (′) or double prime (″) symbol in lieu of the single (') or double (") apostrophe.)

where k is the free-space propagation constant and $\mathbf{J}$ is the current density. It is well known that the solution of this equation is

$$\mathbf{A} = \frac{\mu_0}{4\pi} \int_{v'} \mathbf{J}(\mathbf{r}') \frac{e^{-jkR}}{R} \, dv', \tag{2}$$

where R is the distance from the source point to a field point and $\mathbf{r}'$ is a position vector from a reference point to the source point.

The radiation H and E fields are respectively

$$\mathbf{H} = -\int_{v'} \mathbf{J}(\mathbf{r}') \times \nabla \frac{e^{-jkR}}{4\pi R} \, dv' \tag{3a}$$

and

$$\mathbf{E} = -\frac{1}{j\omega\epsilon_0} \int_{v'} \nabla \left(\mathbf{J} \times \nabla \frac{e^{-jkR}}{4\pi R} \right) dv'. \tag{3b}$$

The far-zone region corresponds to a region in which the radiation field predominates and hence is the region of most interest in connection with antenna radiation characteristics. The far-zone region is characterized by the condition that r (the distance from the reference point to a field point) is much greater than the maximum value of r' and also much greater than the free-space wavelength λ_0, that is, $kr \gg 1$; using the binomial expansion, we have

$$R = |\mathbf{r} - \mathbf{r}'| = r - \hat{\mathbf{r}} \cdot \mathbf{r}'$$

where $\hat{\mathbf{r}}$ is a unit vector along the $\mathbf{r}$ direction. We now approximate $(1/R)\, e^{-jkR}$ by $(1/r) \exp[-jk(r - \hat{\mathbf{r}} \cdot \mathbf{r}')]$. Retaining the terms with the factor $(1/r)$ only, we find

$$\mathbf{H} = \frac{jk}{4\pi r} e^{-jkr} \int_{r'} \mathbf{J} \times \hat{\mathbf{r}} \exp(jk\hat{\mathbf{r}} \cdot \mathbf{r}') \, dv' \tag{4a}$$

and

$$\mathbf{E} = -\xi_0 \, \hat{\mathbf{r}} \times \mathbf{H}, \tag{4b}$$

where ξ_0 is the free-space characteristic impedance. From these two equations, it is evident that both $\mathbf{H}$ and $\mathbf{E}$ are transverse-field limits in the plane which is perpendicular to the radius vector $\hat{\mathbf{r}}$.

Let us assume that

$$\mathbf{H} = \frac{jk}{4\pi r} \; (H_\theta \vec{\theta} + H_\varphi \; \vec{\theta});$$

then

$$\mathbf{E} = \frac{\xi_0 k}{4\pi r} \; (H_\varphi \vec{\varphi} - H_\theta \; \vec{\theta}).$$

The complex Poynting vector, which yields an average power density, can be written as

$$\begin{aligned} P &= \frac{1}{2} \; R_e \; \mathbf{E} \times \mathbf{H}^* \\ &= \frac{k^2 \xi_0}{(4\pi r)^2} \; \hat{\mathbf{r}} \; [\frac{1}{2} \, |H_\theta|^2 + \frac{1}{2} \, |H_\varphi|^2]. \end{aligned} \tag{5}$$

Therefore the antenna pattern, or the average power density, can be separated into two components; one is the θ-polarized component, and the other is the φ-polarized component. Thus the θ-polarization and φ-polarization components of an antenna radiation pattern are determined if either the H or E field is known.

The prior derivation is based on an electric current source. In the event of a magnetic current source, similar results would be arrived at. For detailed derivations, readers are referred to Refs. 1 and 2.

Let us assume an array of antenna elements. By the principle of superposition, the resultant radiation field of this array is the sum of each individual antenna element's radiation field. Hence

$$\begin{aligned} \mathbf{H} &= \sum_n \; \mathbf{H}_n \\ &= \frac{jk}{4\pi r} \; e^{-jkr} \; \sum_n \int_{v_n} \mathbf{J}_n \, (\mathbf{r'}_n) \times \hat{\mathbf{r}} \; \exp\,(jk\hat{\mathbf{r}} \cdot \mathbf{r}_n) \; dv'_n. \end{aligned} \tag{6}$$

One notices in the previous equation that r is the magnitude of vector $\mathbf{r}$, which is a position vector from a common reference point to a field point, while $\hat{\mathbf{r}}$ is a unit vector along this direction. This is depicted in Fig. 1.

Next one may introduce a reference point on each antenna element; the position vector from the common reference point to the reference point on an element is $\mathbf{r}_{no}$. Therefore

$$\mathbf{r'}_n = \mathbf{r'}_{no} + \mathbf{r''}_n,$$

(Substitute a single (′) or double (″) prime symbol in lieu of the single (') or double (") apostrophe as appears below.)

where r''_n is a vector from the element reference point to the current source point. Equation 6 then becomes

$$H = \frac{jk}{4\pi r} e^{-jkr} \sum_n h_n \exp jk\hat{r} \cdot r'_{no}, \tag{7}$$

where

$$h_n(\theta, \varphi) = \int_{v''} J_n(r''_n) \times \hat{r} \exp (jk\hat{r} \cdot r''_n) \, dv''.$$

The h_n is a function of θ and φ and can be viewed as an element pattern. Of course, in the previous formulation, a common coordinate to all elements is used, as shown in Fig. 1. The current distribution and antenna shape v" must be described in this common coordinate system. In a planar array, all antenna elements are oriented in the same direction; if the array elements are identical, then this element pattern can be factored out. Thus it is well known that in a planar array the antenna pattern is the product of the array pattern and element pattern. However, in a conformal array, the elements may be identical but they may have different orientation. Hence the element patterns are not identical in common coordinates, and therefore they cannot be factored out. Thus the problem of computation of the array pattern function is far more complicated. Furthermore, due to this difference of orientation, the polarization does not maintain the simple relation as in a planar array.

Figure 2 shows the coordinate system of a general conformal array under consideration. The position of the nth element in the array is given by a radius vector R_n from the reference point as

$$R_n = (x_n, y_n, z_n), \tag{8}$$

and the element pattern is assumed known in a different primed coordinate system and is given by $E_n(\theta'_n, \varphi'_n)$. The appropriate expression for the far-field pattern of a conformal array can be written as

$$F(\theta, \varphi) = \sum_{n=1}^{N} A_n E_n(\theta'_n, \varphi'_n) \exp [jkR_n \cdot (\hat{R} - \hat{R}_0)], \tag{9}$$

where

$\hat{R}$ = $(\sin\theta \cos\varphi, \sin\theta \sin\varphi, \cos\theta)$,

$\hat{R}_0$ = $(\sin\theta_0 \cos\varphi_0, \sin\theta_0 \sin\varphi_0, \cos\theta_0)$,

A_n = the excitation coefficient of the nth element,

k = the free-space wavenumber,

(θ, φ) = the spherical polar coordinates of the conformal array,

(θ'_n, φ'_n) = the primed spherical coordinates in which the far-field, expression for the nth element is known,

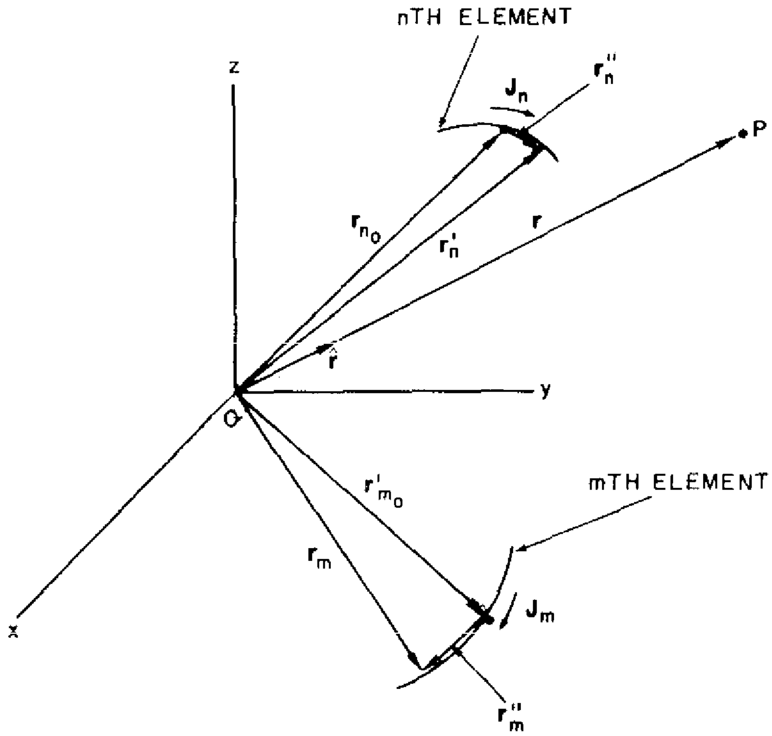

Fig. 1 — Coordinate system for antenna elements n and m

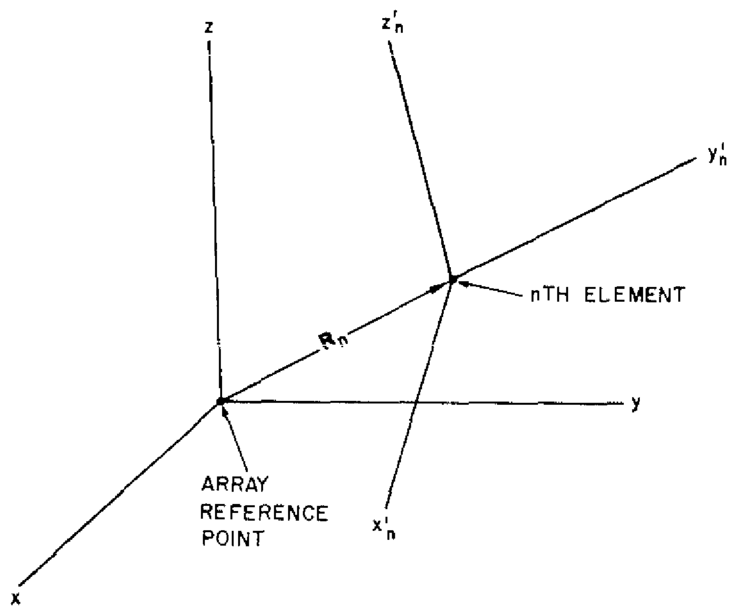

Fig. 2 — Coordinate system of the conformal array

(Substitute a single (′) or double (″) prime symbol in lieu of the single (') or double (") apostrophe when it appears on this and following page.)

and

(θ_0, φ_0) = the direction of the pattern maximum in the unprimed coordinate system.

Equation (9) is expressed in mixed coordinates to conveniently represent the far-field pattern. However, to complete the radiation pattern $F(\theta, \varphi)$ of a conformal array, it is first necessary to transform the element patterns to a common coordinate system (unprimed coordinate system) and then to express the element pattern in terms of the unprimed coordinates θ and φ. This can be done most conveniently using a coordinate transformation, as will be discussed in the next section.

COORDINATE TRANSFORMATION

The element pattern is assumed known in a primed coordinate system and has the general form

$$E(\theta', \varphi') = E_{\theta'}(\theta', \varphi')\,\hat{\theta}' + E_{\varphi'}(\theta', \varphi')\hat{\varphi}', \qquad (10)$$

where $E_{\theta'}$ and $E_{\varphi'}$ are θ' and φ' components. The radial component is not included in Eq. (10) because the interest here is in the far-field radiation pattern. The subscript n is ommited in Eq. (10) for brevity.

It is also assumed, as noted earlier, that the element position in the array is specified in rectangular coordinates with respect to a reference point on the conformal surface and that each element orientation is specified by Euler angles ξ_x, ξ_y, ξ_z with respect to the unprimed rectangular coordinates x, y, and z. Therefore, to transform the element pattern given in Eq. (10) from primed to unprimed coordinates, it is first necessary to transform the pattern into primed rectangular coordinates and finally to unprimed polar coordinates. These transformations can be represented by the following matrix formulation:

$$\begin{vmatrix} E_R(\theta, \varphi) \\ E_\theta(\theta, \varphi) \\ E_\varphi(\theta, \varphi) \end{vmatrix} = |D_{RP}|\ |R_M|\ |D'_{PR}| \begin{vmatrix} 0 \\ E_{\theta'}(\theta', \varphi') \\ E_{\varphi'}(\theta', \varphi') \end{vmatrix}, \qquad (11)$$

where $|D'_{PR}|$ is the matrix which transforms the primed polar coordinates to primed rectangular coordinates; the subscript PR means polar to rectangular coordinates. The matrix $|R_M|$ transforms primed to unprimed rectangular coordinates. The matrix $|D_{RP}|$ transforms unprimed rectangular to polar coordinates; the subscript RP represents rectangular to polar coordinates. The forms of these matrices is obtained next. It is well known that the transformation from polar to rectangular coordinates can be represented in matrix form as

$$\begin{vmatrix} E_{x'} \\ E_{y'} \\ E_{z'} \end{vmatrix} = |D'_{PR}| \begin{vmatrix} E_{R'} \\ E_{\theta'} \\ E_{\varphi'} \end{vmatrix} , \tag{12}$$

where the transformation matrix $|D'_{PR}|$ is given as

$$|D'_{PR}| = \begin{vmatrix} \sin\theta'\cos\varphi' & \cos\theta'\cos\varphi' & -\sin\varphi' \\ \sin\theta'\sin\varphi' & \cos\theta'\sin\varphi' & \cos\varphi' \\ \cos\theta' & -\sin\theta' & 0 \end{vmatrix} . \tag{13}$$

This matrix is known to be real orthogonal. If $|D'_{RP}|$ is the inverse of $|D'_{PR}|$, for an orthogonal matrix the following relation is known to be true:

$$|D'_{RP}| = |D'_{PR}|^{-1} = |D'_{PR}|^T, \tag{14}$$

where $|D'_{PR}|^T$ is the transpose of $|D'_{PR}|$. Therefore the transformation matrix $|D_{RP}|$ in Eq. (11) is given by the transpose of the matrix given in Eq. (13) with θ and φ replacing θ' and φ'.

As mentioned before, the matrix $|R_M|$ is used to transform a function from primed to unprimed rectangular coordinates. Since the far-field element pattern is a function of angular variables only, the coordinate transformation involves only the change in element orientation. This can be obtained by three successive rotations about the three coordinate axes. The first rotation is for an angle ξ_x about the x axis. The orthogonal matrix between the primed and unprimed rectangular coordinate systems for this rotation is

$$|C| = \begin{vmatrix} 1 & 0 & 0 \\ 0 & \cos\xi_x & -\sin\xi_x \\ 0 & \sin\xi_x & \cos\xi_x \end{vmatrix} . \tag{15}$$

The second rotation is for an angle ξ_y about the y axis. The orthogonal matrix for this rotation is

$$|B| = \begin{vmatrix} \cos\xi_y & 0 & \sin\xi_y \\ 0 & 1 & 0 \\ -\sin\xi_y & 0 & \cos\xi_y \end{vmatrix} . \tag{16}$$

(Substitute a single (′) or double (″) prime symbol in lieu of the single (') or double (") apostrophe where it appears o this ~~Iwo~~ and following page.)

The third and final rotation is for an angle ξ_z about the z axis. The orthogonal matrix for this rotation is

$$|A| = \begin{vmatrix} \cos\xi_z & -\sin\xi_z & 0 \\ \sin\xi_z & \cos\xi_z & 0 \\ 0 & 0 & 1 \end{vmatrix} \tag{17}$$

In all three rotations the angle of rotation is positive when the rotation is clockwise with respect to the axis of rotation. The overall transformation matrix may be written as

$$|R_M| = |A|\,|B|\,|C|. \tag{18}$$

One should note here that the order of matrix multiplication is not commutative; thus the sequence of these transformations is not interchangeable. Equation (9) can now be rewritten in the matrix form with the aid of Eq. (11) as

$$\begin{vmatrix} F_R(\theta,\varphi) \\ F_\theta(\theta,\varphi) \\ F_\varphi(\theta,\varphi) \end{vmatrix} = \sum_{n=1}^{N} A_n \exp\,[jkR_n \cdot (\hat{R} - \hat{R}_0)]\;|D_{RP}|\,|R_M|\,|D'_{PR}| \begin{vmatrix} 0 \\ E_{\theta'}(\theta',\varphi') \\ E_{\varphi'}(\theta',\varphi') \end{vmatrix}. \tag{19}$$

However, the right-hand side is still expressed in primed coordinate variables θ' and φ'. These variables can be eliminated by considering the relations between the primed and unprimed coordinate variables. The relations between rectangular coordinate variables can be written as

$$\begin{vmatrix} x \\ y \\ z \end{vmatrix} = |R_M| \begin{vmatrix} x' \\ y' \\ z' \end{vmatrix}, \tag{20}$$

and the relations between the rectangular and polar coordinate variables are given by

$$\begin{vmatrix} x \\ y \\ z \end{vmatrix} = \begin{vmatrix} \sin\theta\cos\varphi \\ \sin\theta\sin\varphi \\ \cos\theta \end{vmatrix} \tag{21}$$

and

$$\begin{vmatrix} x' \\ y' \\ z' \end{vmatrix} = \begin{vmatrix} \sin\theta'\cos\varphi' \\ \sin\theta'\sin\varphi' \\ \cos\theta' \end{vmatrix} \tag{22}$$

Substituting Eqs. (18), (21), and (22) in Eq. (20), it can be shown that

$$\theta' = \cos^{-1} Z'(\theta, \varphi) \tag{23}$$

$$\varphi' = \tan^{-1} \frac{y'(\theta, \varphi)}{x'(\theta, \varphi)} \quad . \tag{24}$$

The ambiguity in the value of the arctangent function in Eq. (24) is resolved by applying the same set of rules that one uses to determine the value of $\tan^{-1}(y/x)$, where x and y are the rectangular coordinate variables.

DESCRIPTION OF THE COMPUTER PROGRAM

A computer program has been written in FORTRAN language which computes and plots the far-field pattern of a conformal array for a given set of array parameters and is included as the appendix. The program is divided into a main program and several subroutines to allow flexibility. The name of the main program is Coflaray (short name for conformal array). The array element orientations (angles) and positions can be supplied as input data or can be calculated using subroutine ELAGPO (short name for element angle and position). Three separate ELAGPO subroutines were developed. The first one is for a doubly curved surface with uniformly spaced elements; the second one is for a doubly curved surface with projected uniformly spaced elements; the third one is for an array on a conic surface. Therefore, depending on the conformal array under consideration, the corresponding ELAGPO subroutine should be used. It is also possible to write additional ELAGPO subroutines for any well-defined conformal array, and they can be substituted for the ones written for this report. Except for this ELAGPO subroutine, the other subroutines which should be included with the Coflaray program, given in the order in which they are called in the program, are:

CODTRF	—	This subroutine computes the elements of the R_M matrix, as given by Eq. (18), for each array element.
DPRMAT	—	This subroutine computes the elements of the matrices $\lvert D'_{PR} \rvert$ and $\lvert D_{PR} \rvert$ for each array element, as given by Eqs. (13) and (14).
ANGTRF	—	This subroutine computes the relations between primed and unprimed coordinate variables, as given by Eqs. (23) and (24).
ELPAT	—	This subroutine is used to select the array element pattern as either that of a vertical dipole or a horizontal dipole.
MATMUT	—	This subroutine performs the matrix multiplication shown in Eq. (11).
FRAME	—	This subroutine is used to establish the coordinate-system frame to plot the radiation patterns.
PENCHG	—	This subroutine is called in the FRAME subroutine to change to a different pen in plotting.
REZERO	—	This subroutine is used to reset the origin if more than one plot is desired in any given computer run.

The program requires four data cards. The first data card should contain six variables in an integer format of 6I5. These variables are:

NE — Number of elements in the array. If the subroutine ELAGPO is to be used, NE should be less than or equal to zero.

NC — Number of antenna pattern cuts required in φ plane.

NP — Number of points at which the antenna pattern is calculated and plotted.

LLL — Controls the amount of printout needed:
If LLL = 0, printout for diagnostic purpose;
LLL = 1, print element positions and rotation angles;
LLL = 2, print pattern function only;
LLL > 2, no printout.

LBP — If it is zero, the scanning is obtained by using row-column planar-array phasing. If it is one, exact conformal-array phasing is used.

NBP — Number of phase-shift bits used in digital phase control. If NBP is greater than 10, analog phases are assumed.

The second data card (or set of cards) depends on whether or not the ELAGPO subroutine is needed. If it is needed, then which particular ELAGPO subroutine is used. First, we will discuss the set of data cards needed when the ELAGPO subroutine is not used. In that case the data cards should contain all the element positions, the element orientations (rotation angles), and the specification of the ground plane. The data cards should conform to the following read and format statements:

```
    READ 101, ((W(I,J), I = 1, NE), J = 1,3),
    READ 101, ((G(I,J), I = 1, NE), J = 1,3),
    READ 100, (LG(I), I = 1,3),
101 FORMAT (8F10.6),
100 FORMAT (6I5),
```

where

W(I,J) are the element positions,

G(I,J) are the element rotation angles,

and the

LG array specifies the ground plane of radiators as follows:

If LG(1) = 1, LG(2) = LG(3) = 0, the ground plane is the zy plane;
LG(1) = LG(3) = 0, LG(2) = 1, the ground plane is the xz plane;
LG(1) = LG(2) = 0, LG(3) = 1, the ground plane is the xy plane.

When the subroutine ELAGPO is used to compute the array element positions and orientations, the second data card contains the description of the conformal surface and the array dimensions. For a doubly curved surface (with uniformly or projected uniformly spaced elements), the second data card should contain seven variables conforming with the format of 2I5,4F10.6,I2. These variables are:

NCX	—	Number of columns in the x direction
NRY	—	Number of rows in the y direction
AX	—	Aperture in the x direction, in wavelengths
AY	—	Aperture in the y direction, in wavelengths
RX	—	Radius of curvature in the x direction, in wavelengths
RY	—	Radius of curvature in the y direction, in wavelengths
LP	—	If LP = 0, the array element is a horizontal dipole; if LP = 1, it is a vertical dipole.

When the subroutine ELAGPO for a conic array is used, the second data card should contain eight variables conforming with the format I5,5F10.4,2I5. These variables are:

MM	—	Number of rows;
RB	—	Base radius in wavelengths;
ARC	—	Cone arc (in degrees) occupied by the array;
TC	—	Cone angle in degrees;
DX	—	Spacing in the x direction, in wavelengths;
DY	—	Spacing in the y direction, in wavelengths;
LP	—	If LP = 0, the array element is a horizontal dipole; if LP = 1, it is a vertical dipole;
LRT	—	If LRT = 0, the array element distribution is on a rectangular grid; if LRT = 1, it is on a triangular grid.

The third and the fourth (or last two) data cards should contain the angular range and the plane in which the radiation pattern is desired. The third data card should contain one or more values of φ [FI(I) in degrees] defining the plane or planes in which the radiation pattern is desired. This data should conform to the format 8F10.6. The last data card contains four variables conforming with the format 8F10.6; these variables are:

FIO	—	Scan angle φ_0 in degrees;
TAO	—	Scan angle θ_0 in degrees;
TAPI	—	Initial value of θ;
TAPF	—	Final value of θ over which the radiation pattern is desired.

EXAMPLES

A few examples of computing the radiation patterns of a conformal array are included here which illustrate several features of the program.

The first example considered is uniformly spaced circular-arc array with 32 elements, as shown in Fig. 3. The individual elements are assumed to be vertical dipoles (dipoles normal to the array plane). The array aperture in the x direction (projection of the array arc onto the x axis) is assumed to 15.5λ, so that the average interelement spacing in the projected plane (x axis) is 0.5λ. The interest here is to find the radiation pattern when scanned to 15° in the array plane ($\varphi = 0^\circ$ plane). Since the interest is in an uniformly spaced circular-arc array, the subroutine ELAGPO for a doubly curved surface with uniformly spaced elements is used. The four data cards for this example have the following values:

Data Card 1: NE=0, NC=1, NP=361, LLL=2, LBP=1, NBP=11

Data Card 2: NCX=32, NRY=1, AX=15.5, AY=0., RX=12.66333, RY=0., LP=1

Data Card 3: FI(I)=0.

Data Card 4: FIO=0., TAO=15°, TAPI=−90°, TAPF=90°.

Using these data cards, the computer prints (printout not included here) the values of the normalizing factor, the normalized values (expressed in dB) of the radiation field at 361 values of θ, with increments of 0.5° over the interval $-90 < \theta < 90^\circ$, and the steering phases used to scan the beam. In this example it is specified (LBP = 1) to use correct steering phases. The computer output includes a plot of the radiation pattern which is shown in Fig. 4.

The second example is the same as the first example except that the array elements are assumed to be projected uniformly spaced (when projected onto the x axis they have equal interelement spacing) on the circular arc, as shown in Fig. 5. The advantage of this type of distribution is that the pattern can be scanned using simpler row steering of a uniformly spaced linear array [3], as will be illustrated in Example 3. The data cards for Example 2 are the same as those of Example 1. However, the subroutine ELAGPO for a doubly curved surface with projected uniformly spaced elements is used instead of the one used in Example 1. The computed radiation pattern is shown in Fig. 6.

The third example is the same as the second example, except for the steering phases used to scan the array pattern. In this example, approximate steering (linear array steering) phases are used instead of the correct steering phases. So the data cards are the same as that of Example 2, except that the value of LBP in the first data card is changed from 1 to 0. The ELAGPO subroutine used is the same as that used in Example 2. The computed radiation pattern for this example is shown in Fig. 7.

The fourth example considered is a 7-by-13-element, uniformly spaced array on a circular cylindrical surface, as shown in Fig. 8. The array apertures are assumed to be 3λ in the x direction and 6λ in the y direction. The radius of curvature of the circular

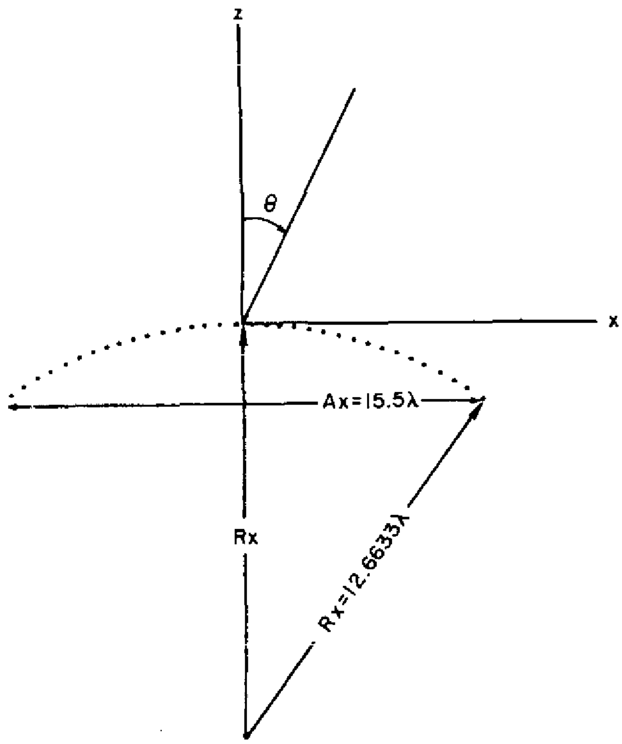

Fig. 3 — Uniformly spaced circular-arc array

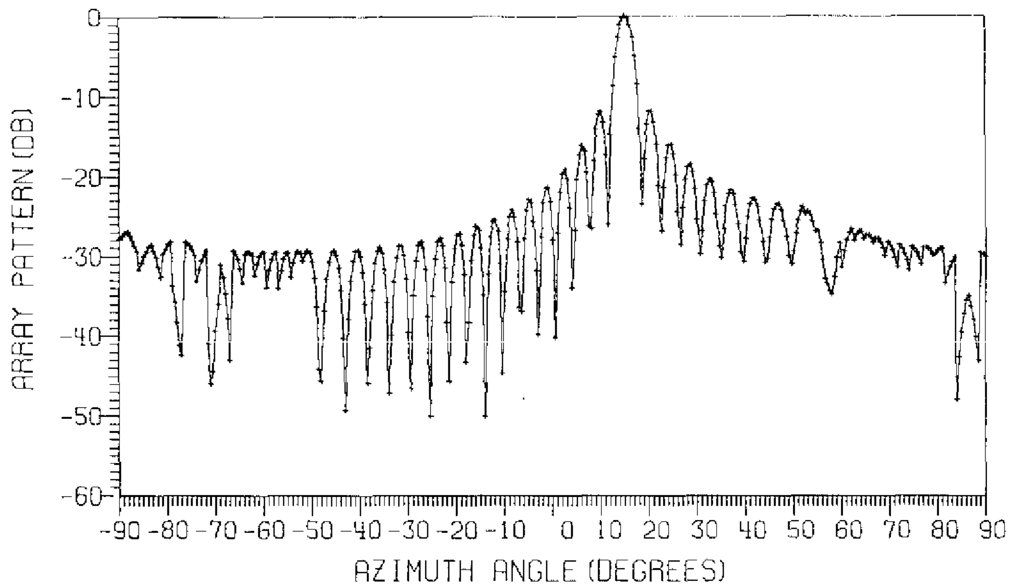

Fig. 4 — Radiation pattern of a uniformly spaced circular-arc array

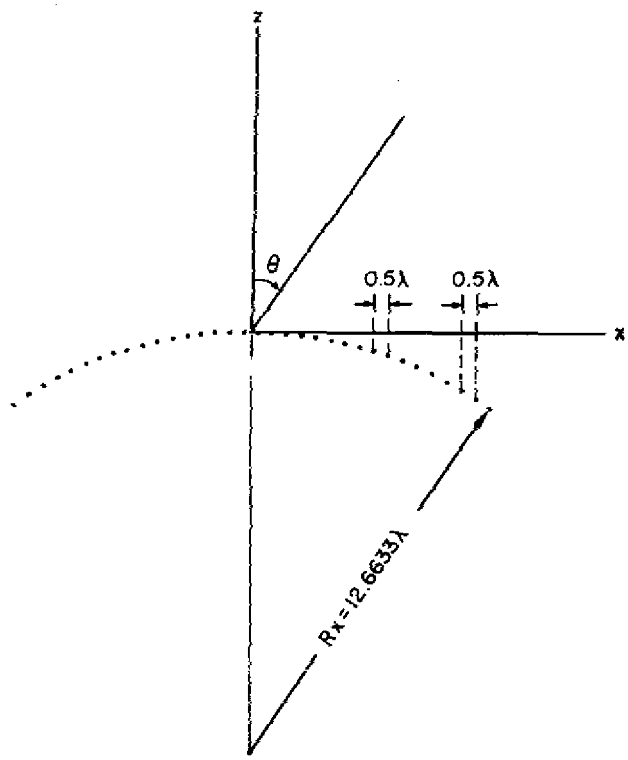

Fig. 5 — Projected uniformly spaced circular-arc array

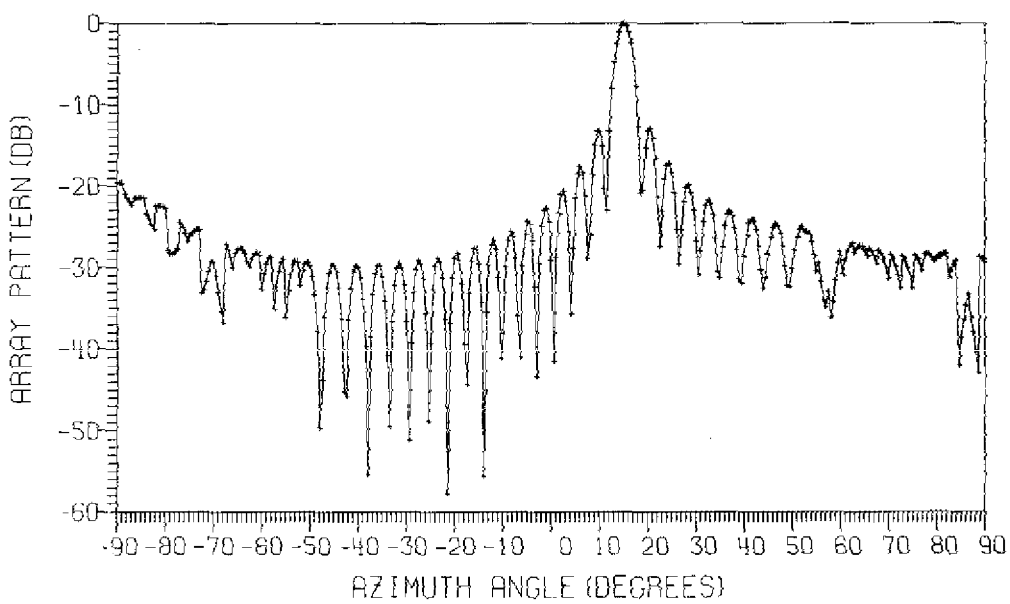

Fig. 6 — Radiation pattern of a projected uniformly spaced circular-arc array, scanned to 15° by applying exact phase steering

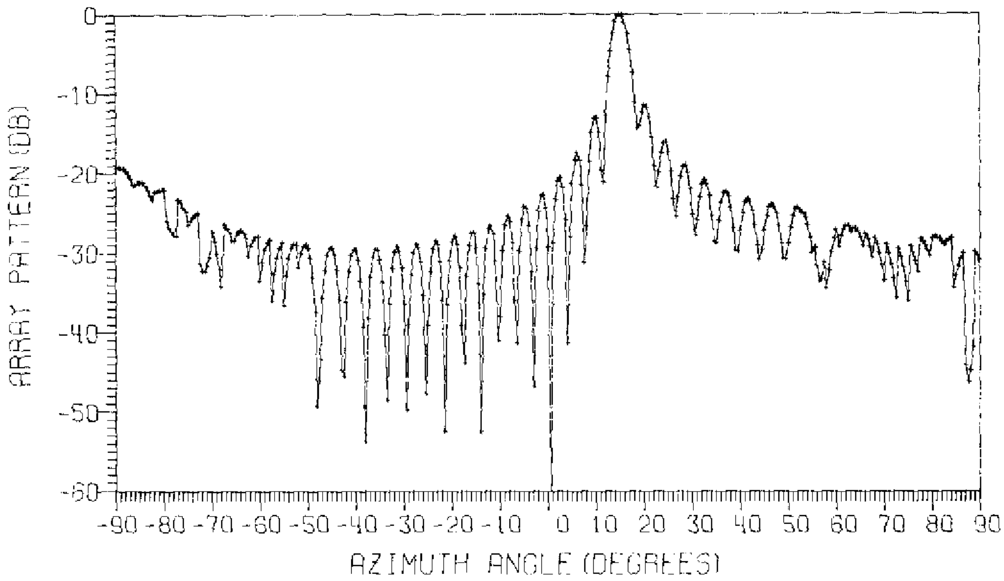

Fig. 7 — Radiation pattern of a projected uniformly spaced circular-arc array, scanned to 15° by applying linear-array phase steering

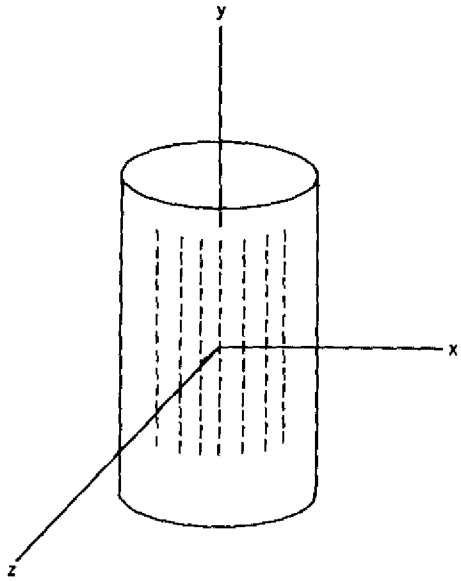

Fig. 8 — Uniformly spaced circular-cylinder array

cylinder is assumed to be 5.1816λ. To obtain a broadside pattern in the $\varphi = 0$ plane (xz plane), the data cards should contain the following values:

Data Card 1: NE=0, NC=1, NP=361, LLL=2, LBP=1, NBP=11

Data Card 2: NCX=7, NRY=13, AX=3.0, AY=6.0, RX=5.1816, RY=0., LP=1

Data Card 3: FI(I)=0.

Data Card 4: FIO=0., TAO=0., TAPI=−90°, TAPF=90°.

Using the subroutine ELAGPO for a doubly curved surface with uniformly spaced elements, the radiation pattern plotted by the computer is shown in Fig. 9. The step changes noted in Fig. 9 and some of the later figures are the result of the provision provided in the program which makes it possible to drop the element contribution whenever the element becomes invisible (due to the curved surface) from the point at which the radiation field is being computed.

The fifth example is a 7-by-13-element, uniformly spaced planar array. The array apertures in the x and y directions are assumed to be the same as that of Example 4. The broadside pattern in the $\varphi = 0$ plane for this planar array can be obtained using the same ELAGPO subroutine and data cards as those of Example 4 by simply changing the RX value to zero in Data Card 2. The radiation pattern for this planar array is shown in Fig. 10.

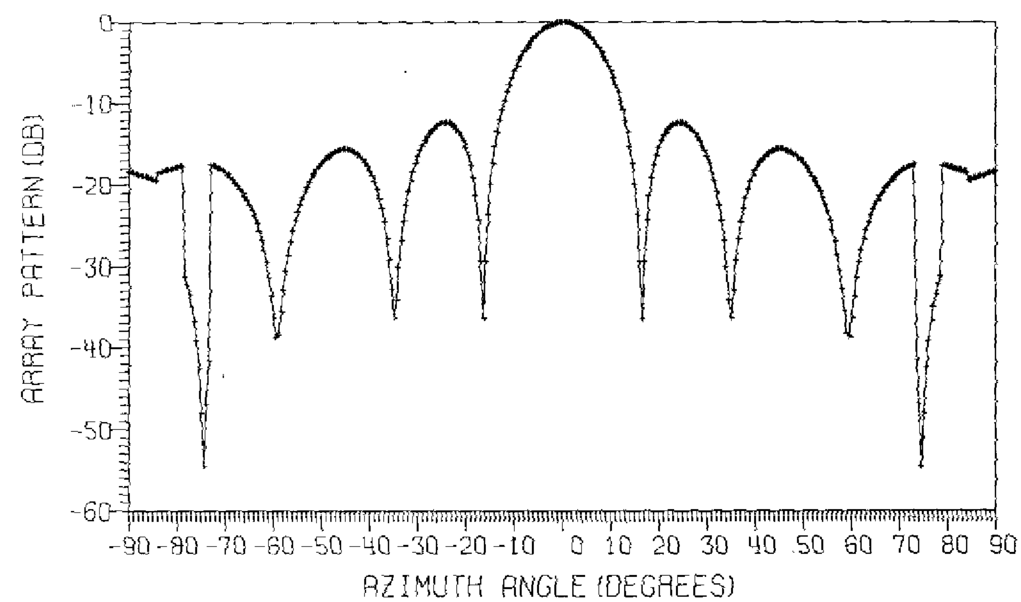

Fig. 9 — Radiation pattern of a uniformly spaced circular-cylinder array

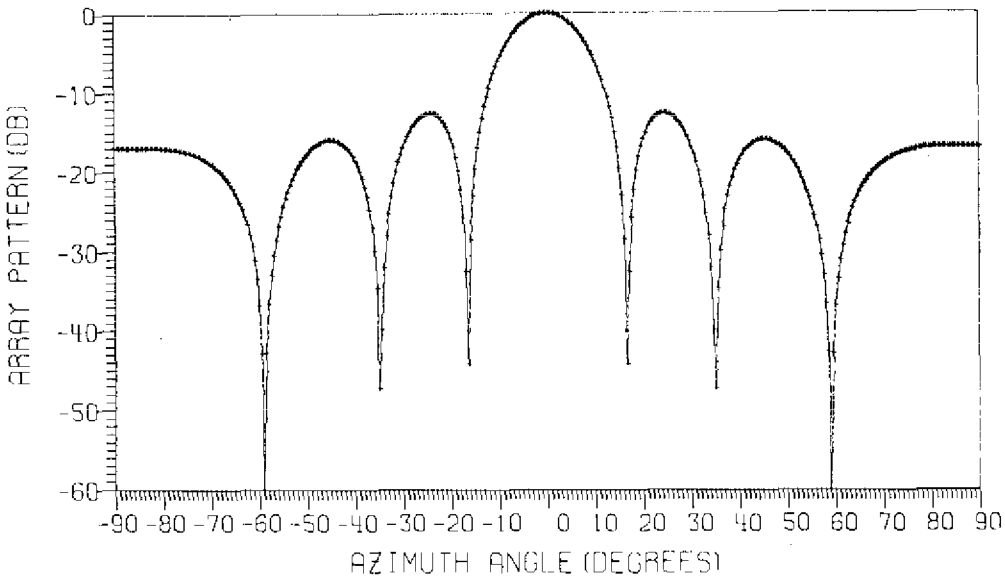

Fig. 10 — Radiation pattern of a uniformly spaced planar array

The sixth example is a 7-by-13-element array on a conic surface with a base radius of 137.46λ and a cone angle of 7°. There are seven rows, and each row contains 13 elements. The spacing between the rows is assumed to 0.5λ; the interelement spacing along the base arc is also assumed to be 0.5λ. It is assumed that the array elements are vertical dipoles and that a rectangular grid arrangement is used. The coordinate system applied to this array is shown in Fig. 11. The following values for the data cards are used to obtain a pattern in the $\varphi = 0$ plane with zero scan angles:

Data Card 1: NE=0, NC=1, NP=361, LLL=2, LBP=1, NBP=11

Data Card 2: MM=7, RB=137.46, ARC=2.55, TC=7°, DX=0.5, DY=0.5, LP=1, LRT=0

Data Card 3: FI(I)=0

Data Card 4: FIO=0., TAO=0., TAPI=−90°, TAPF=90°.

The radiation pattern, obtained by using these data cards and the subroutine ELAGPO for a conic surface, is shown in Fig. 12. Because of the conic surface, there is a crosspolarization component which is plotted in the same figure as a separate curve. In this example, the crosspolarization component is quite low because the radius of curvature of the conic surface is large and the array surface approximates a planar surface.

The final example illustrates two additional and useful features of the subroutine ELAGPO for a conic surface. The first feature is that this subroutine can be used for the arrays on a circular cylinder by specifying the cone angle TC = 0. The second additional feature is that a triangular grid instead of a rectangular grid arrangement can be specified. In this example an array on a circular cylinder with a triangular grid is considered, as shown in Fig. 13. To obtain a broadside pattern in the $\varphi = 0$ plane, the following data cards are used:

Data Card 1: NE=0, NC=1, NP=361, LLL=2, LBP=1, NBP=11

Data Card 2: MM=10, RB=5.1816, ARC=33.167, TC=0., DX=0.75, DY=.666, LP=1, LRT=1

Data Card 3: FI(I) = 0.

Data Card 4: FIO=0., TAO=0., TAPI=−90°, TAPF=90°.

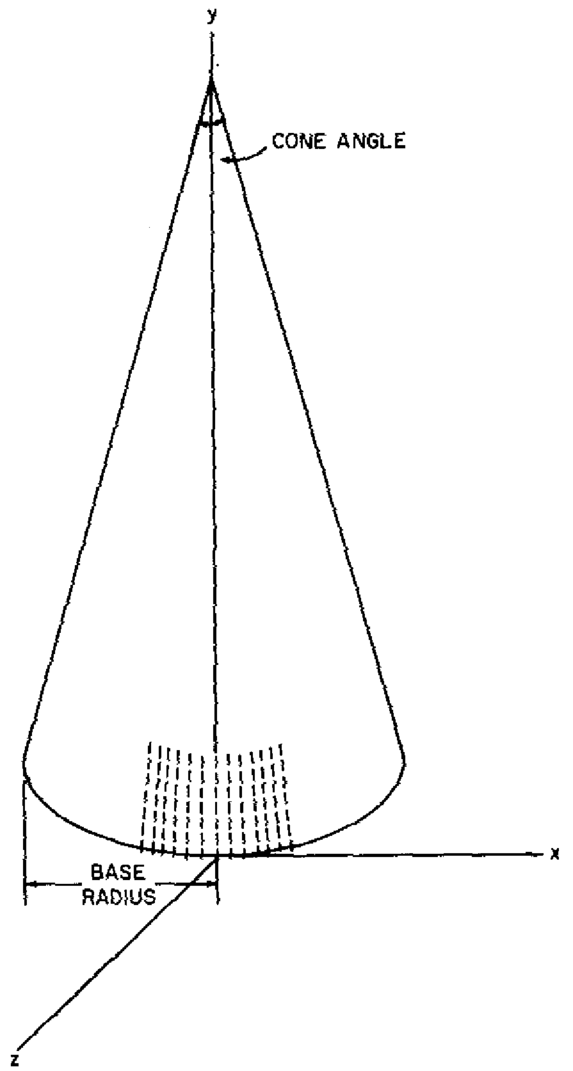

Fig. 11 — Conic-surface array

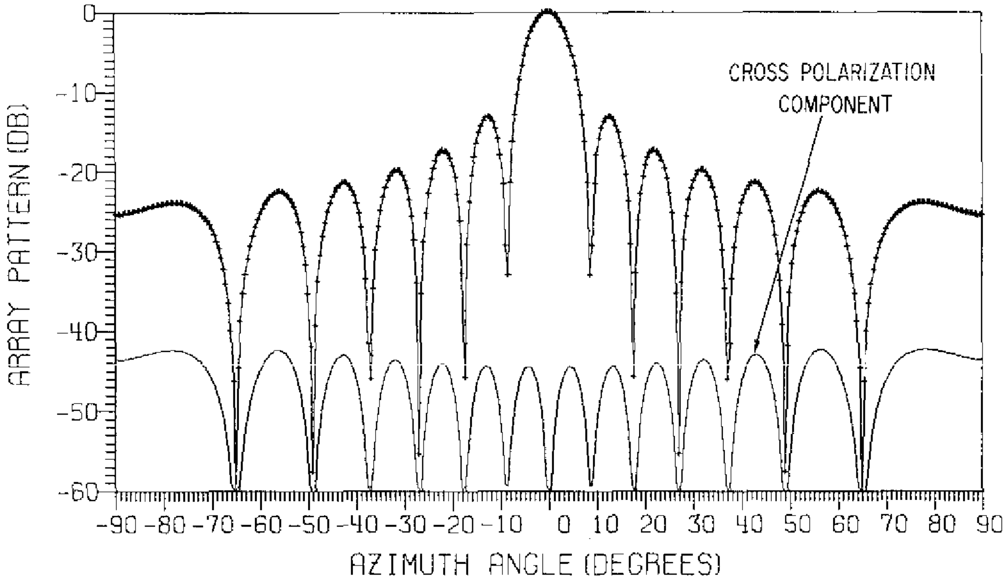

Fig. 12 — Radiation pattern of a conic-surface array

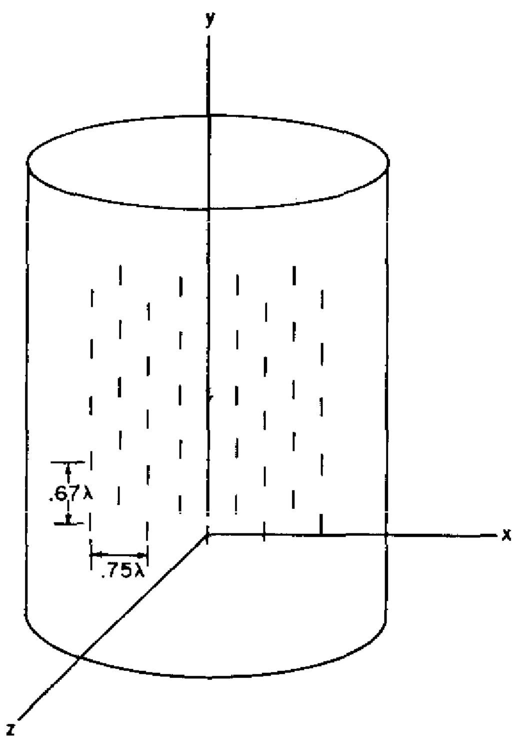

Fig. 13 — Triangular-grid array on a circular cylinder

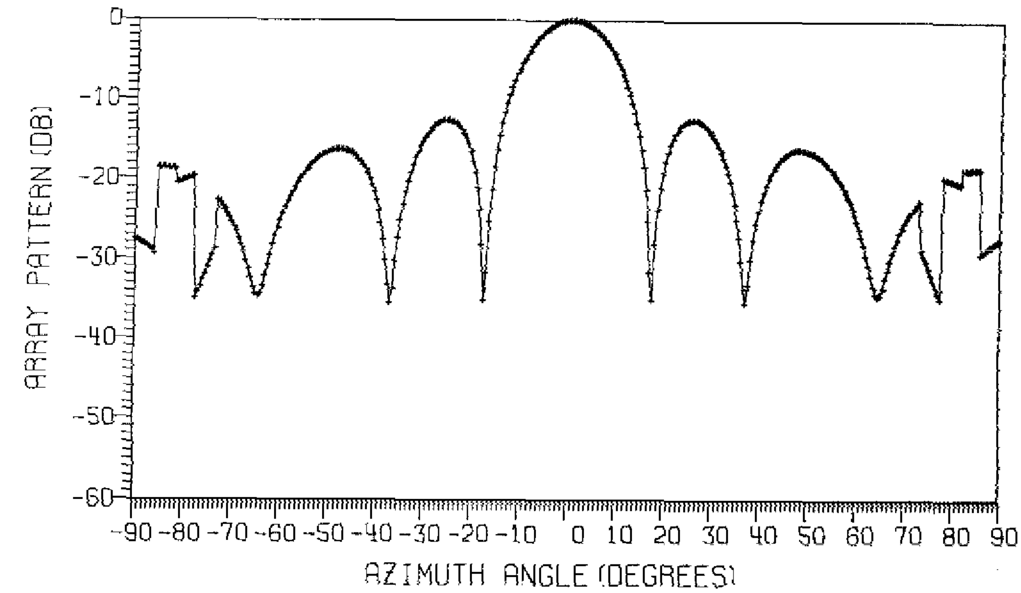

Fig. 14 — Radiation pattern of a triangular-grid array on a circular cylinder

The radiation obtained by using these data cards with subroutine ELAGPo for a conic surface is shown in Fig. 14. Because of the large value of DX(.75), the radiation pattern obtained by using a rectangular grid (not included here) will have high sidelobes. From Fig. 14 it is clear that the triangular grid has the effect of reducing these sidelobes.

REFERENCES

1. R.E. Collins and F.J. Zucker, *Antenna Theory*, Part I, McGraw-Hill Book Company, New York, 1969.
2. R.C. Hansen, *Microwave Scanning Antennas*, Vol. 1 Academic Press, New York, 1964.
3. J.B.L. Rao and J.K. Hsiao, "Phased Array Antenna Analysis for Pacific Missile Range Instrumentation Aircraft," NRL Report 7806, Sept. 1974.

```
      PROGRAM COFLAR_Y
      COMMON/1/PLTARRAY(254),FMAT(1000,3)
      COMMON/2/G(1000,3),W(1000,3)
      COMMON/3/RMAT(1000,3,3),DMAT(3,3),EMAT(3),TEMP(3)
      COMMON/4/FI(10)
      COMMON/5/DMATUP(3,3)
      COMMON/6/LG(3),A(3),B(3)
      DIMENSION CR(3),CI(3),BP(1000)
C
C ENTER PLOT FRAME PARAMETERS
      CALL PLOTS (PLTARRAY,254,18)
      XM=9.0
      YM=5.
      SY=2.
      SYM=SY+YM
      YSL=60.
      NY=60
C
C     ENTER RANDOM NUMBER SEED
      RS=TIMEF(X)
      CALL RANFSET(RS)
C
C INPUT DATA
C     G-ARRAY, THREE COORDINATE ROTATION ANGLES FOR EACH ELEMENT
C     W-ARRAY ELEMENT LOCATIONS
C     ROTATIONAL CONVENTION-CLOCKWISE FROM PRIMED TO UNPRIMED COORDINATE
C     IS CONSIDERED POSITIVE
C     ROTATIONAL SEQUENCES-X,Y,Z
C     LG ARRAY SECIFIES THE GROUND PLANE OF RADIATORS
C     LG(1)=1,LG(2)=LG(3)=0 ZY PLANE
C     LG(1)=LG(3)=0,LG(2)=1,XZ PLANE
C     LG(1=LG(2)=0,LG(3)=1,XY PLANE
C     NE-NUMBER OF ARRAY ELEMENT
C     NC-NUMBER OF ANTENNA PATTERN CUTS IN ANGLE FI PLANE
C     NP-NUMBER OF POINT OF ANTENNA PATTERN TO BE PLOTTED
C     LLL=0,SET PRINT-OUT FOR DIAGNOSTIC PURPOSE
C     IF LLL=1, PRINT ELEMENTS POSITIONS AND ROTATION ANGLES
C     IF LLL=2, PRINT PATTERN FUNCTION ONLY
C     IF LLL GREATER THAN 2 NO PRINT-OUT
C     LBP=0,APPROXIMATE BY A PLANAR ARRAY PHASE
C     LBP=1,WITH CORRECT CONFORMAL ARRAY PHASE
C     NBP, NO. OF PHSE SHIFT BITS,IF NBP.GT.10 CORRCET PHASES ARE USED
C     TAPI,PATTERN PLOT STARTING ANGLE
C     TAPF,PATTERN PLOT FINAL ANGLE
C
C ENTER INPUT DATA
      KK=0
 93   READ 100,NE,NC,NP,LLL,LBP,NBP
C
      IF(EOF,60)99,91
 91   IF(KK.EQ.0)GO TO 92
      CALL REZERO(XM)
 92   KK=1
      IF(NE.LE.0)GO TO 1
      READ 101,((W(I,J),I=1,NE),J=1,3)
      READ 101,((G(I,J),I=1,NE),J=1,3)
      READ 100,(LG(I),I=1,3)
      GO TO 2
 1    CALL ELAGPO(NE,LP)
```

```
 2     IF(LLL.GT.1)GO TO 201
       PRINT 104
       DO 202,I=1,3
 202   PRINT 102,(W(N,I),N=1,NE)
       PRINT 105
       DO 212 I=1,3
 212   PRINT 102,(G(N,I),N=1,NE)
 201   PI=3.14159265358979323846
       PI2=2.*PI
       ATR=PI/180.
C
C COMPUTE THE R-MATRIX
       DO 10 I=1,NE
 10    CALL CODTRF(I)
       IF(LLL.GT.0)GO TO 203
       DO 204 I=1,3
       DO 204 J=1,3
 204   PRINT 102,  (RMAT(L,I,J),L=1,NE,10)
C
C DETERMINE FIELD POINT ANGLES
 203   READ 101,(FI(I),I=1,NC)
 101   FORMAT( 8F10.6)
C
C ENTER SCAN ANGLES
       READ 101,FIO,TAO,TAPI,TAPF
       FIO=FIO*ATR
       TAO=TAO*ATR
       TAPIR=TAPI*ATR
       TAPFR=TAPF*ATR
       TAINC=(TAPFR-TAPIR)/(NP-1)
       TAAINC=TAINC/ATR
       JN=(NP-1)/3
       COTAO=COSF(TAO)
       SITAO=SINF(TAO)
       COFIO=COSF(FIO)
       SIFIO=SINF(FIO)
       XP=SITAO*COFIO
       YP=SITAO*SIFIO
       ZP=COTAO*LBP+1.0-LBP
       DO 70 K=1,NE
       BPK=W(K,1)*XP+W(K,2)*YP+ W(K,3)*ZP
       BPK=BPK-INTF(BPK)
       IF(NBP.GT.10) GOTO 73
       BP(K)=0.
       DO 71 N=1,NBP
       B1=1./2.**N
       IF(BPK.LE.B1)GO TO 72
       BPK=BPK-B1
       BP(K)=B1+BP(K)
 72    IF(N.NE.NBP)GO TO 71
       R=RANF(-1)
       IF(R.LT..5)GO TO 71
       BP(K)=B1+BP(K)
 71    CONTINUE
       GOTO 70
   73 BP(K)=BPK
 70    CONTINUE
C
C COMPUTE PATTERN
```

```
      DO 20 I=1,NC
      FI(I)=FI(I)*ATR
      TA=TAPIR
      COFIUP=COSF(FI(I))
      SIFIUP=SINF(FI(I))
      FNOR=0.
      DO 30 J=1,NP
      COTAUP=COSF(TA)
      SITAUP=SINF(TA)
      CALL DPRMAT(COFIUP,SIFIUP,COTAUP,SITAUP,DMATUP)
      IF(LLL.GT.0)GO TO 211
      JMOD=MOD(J,JN)
      IF(JMOD.NE.1)GO TO 211
      TAANG=TA/ATR
      PRINT 109,TAANG
      PRINT 102,((DMATUP(L,M),M=1,3),L=1,3)
 211  DO 31 L=1,3
 31   CR(L)=CI(L)=0.
      XF=SITAUP*COFIUP
      YF=SITAUP*SIFIUP
      ZF=COTAUP
C
C     SUM THE ELEMENT CONTRIBUTION
      DO 40 K=1,NE
      COFI=COFIUP
      SIFI=SIFIUP
      COTA=COTAUP
      SITA=SITAUP
C
C TRANSFER ANGEL FROM UNPRIMED COORDINATE TO PRIMED COORDINATE
      CALL ANGTRF(COFI,SIFI,COTA,SITA,K,LJG)
      IF(LLL.GT.0)GO TO 209
      KMOD=MOD(K,10)
      IF(KMOD.NE.1 .OR.JMOD.NE.1)GO TO 209
      PRINT 108,K
      PRINT 102,COFI,SIFI,COTA,SITA
      PRINT 102,(A(L),L=1,3),(B(L),L=1,3)
C
C  FORM D MATRIX
 209  IF(LJG.LE.0)GO TO 40
      CALL DPRMAT(COFI,SIFI,COTA,SITA,DMAT)
      IF(LLL.GT.0)GO TO 205
      IF(KMOD.NE.1 .OR.JMOD.NE.1)GO TO 205
      PRINT 102,((DMAT(L,M),M=1,3),L=1,3)
C
C ENTER ELEMENT PATTERN FUNCTION
 205  CALL ELPAT(COFI,SIFI,COTA,SITA,LP)
C
C PERFORM MATRIX MULTIPLICATIONS
      CALL MATMUT(K)
      IF(LLL.GT.0)GO TO 207
      IF(KMOD.NE.1 .OR.JMOD.NE.1)GO TO 207
      PRINT 102,(TEMP(L),L=1,3)
C
C FIND THE PHASE
 207  PHASE=      W(K,1)*XF+W(K,2)*YF+W(K,3)*ZF
      PHASE=PHASE-INTF(PHASE)
      PHASE=PI2*(PHASE-BP(K))
      PRE=COSF(PHASE)
```

```
      PIM=SINF(PHASE)
C
C FORM PATTERN FUNCTION
      DO 41 L=1,3
      CR(L)=CR(L)+TEMP(L)*PRE
 41   CI(L)=CI(L)+TEMP(L)*PIM
 40   CONTINUE
      DO 32 L=1,3
      FMAT(J,L)=CR(L)**2+CI(L)**2
 32   IF(FMAT(J,L).GT.FNOR)FNOR=FMAT(J,L)
 30   TA=TA+TAINC
      XSL=TAPF-TAPI
      NX=XSL
      IF(I.EQ.1)GO TO 21
      CALL REZERO
 21   CALL FRAME(XM,YM,XSL,YSL,NX,NY,TAPI)
      IF(LLL.GT.2)GO TO 213
      PRINT 107,FNOR
C
C PLOT ANTENNA PATTERN
 213  DO 60 K=1,3
      TA=TAPI
      LM=0
      DO 50 L=1,NP
      FMAT(L,K)=FMAT(L,K)/FNOR
      IF(K.LE.1)GO TO 50
      IF(FMAT(L,K).GT.0.0000001)GO TO 52
      FMAT(L,K)=-YSL
      GO TO 53
 52   FMAT(L,K)=10.*ALOG10(FMAT(L,K))
 53   Y=YM*(1.+FMAT(L,K)/YSL)+SY
      X=(TA+90.0)*XM/XSL
      IF(Y.GT.SY)GO TO 56
      Y=SY
      GO TO 54
 56   IF(Y.GT.SYM)Y=SYM
      IF(K.GT.2)GO TO 54
      IF(LM.GT.0)GO TO 55
      CALL SYMBOL(X,Y,.06,3,0.,-1)
      GO TO 57
 55   CALL SYMBOL(X,Y,.06,3,0.,-2)
      GO TO 57
 54   IF(LM.GT.0)GO TO 51
      CALL PLOT(X,Y,3)
      GO TO 57
 51   CALL PLOT(X,Y,2)
 57   LM=1
 50   TA=TA+TAAINC
      IF(LLL.GT.2)GO TO 60
      PRINT 103,K
      PRINT 106,(FMAT(J,K),J=1,NP)
 60   CONTINUE
 20   CONTINUE
      PRINT 110,(BP(K),K=1,NE)
  110 FORMAT(//,10X,*STEERING PHASE*,//,(10X,10F12.6))
      GO TO 93
 99   CALL STOP PLOT
 100  FORMAT(6I5)
 102  FORMAT( /,(10X,10E12.3))
 103  FORMAT(//,20X,*ARRAY PATTERN (DB)*,5X,*K=*,I5)
 104  FORMAT(//,20X,*ELEMENT LOCATIONS*)
 105  FORMAT(//,20X,*ELEMENT ROTATION ANGLES*)
 106  FORMAT(//,(10X,10F12.6))
 107  FORMAT(//,10X,*ARRAY NORMALIZING FACTOR*,5X,F15.6)
 108  FORMAT(//,10X,*ELEMENT*,I5)
 109  FORMAT(//,20X,*AZIMUTH ANGLE=*,F6.2)
      END
```

```
      SUBROUTINE ELAGPO(NE,LP)
C     THIS SUBROUTINE IS FOR A DOUBLY CURVED SURFACE
      COMMON/2/G(1000,3),W(1000,3)
      COMMON/6/LG(3),A(3),B(3)
C     GROUND PLANE IS ON XZ PLANE
      LG(1)=LG(3)=0
      LG(2)=1
C     NRY,NO OF ROWS IN Y DIRECTION
C     NCX, NO OF COLUMN IN X DIRECTION
C     AX, X-DIRECTION APERTURE, IN WAVELENGTHS
C     AY, Y-DIRECTION APERTURE, IN WAVELENGTHS
C     RX, RADIUS OF CURVATURE IN X DIRECTION
C     RY, RADIUS OF CURVATURE IN Y DIRECTION
      PI=3.14159265358979323846
      PIH=PI/2.
C     LP=0, HORIZONTAL DIPOLES, LP=1, VERTICAL DIPOLES
C     THE FIRST ELEMENT STARTED AT -X AND -Y
      READ 100,NCX,NRY,AX,AY,RX,RY,LP
  100 FORMAT(2I5,4F10.6,I2)
      PRINT 101,NCX,NRY,AX,AY,RX,RY,LP
  101 FORMAT(10X,2I5,4F10.4,I5)
      IF(RY.EQ.0.)GO TO 2
      ARCYH=ASINF(.5*AY/RY)
      ARCY=2.*ARCYH
      ARCYINC=ARCY/(NRY-1)
    2 IF(RX.EQ.0.)GO TO 3
      ARCXH=ASINF(.5*AX/RX)
      ARCX=2.*ARCXH
      ARCXINC=ARCX/(NCX-1)
    3 IF(NCX-1.EQ.0)GO TO 8
      DX=AX/(NCX-1)
    8 IF(NRY-1.EQ.0)GO TO 9
      DY=AY/(NRY-1)
    9 AUGY=AUGX=0.
      LL=0
      DO 10 I=1,NRY
      IF(RY.GT.0.)GO TO 4
      YS=(I-1)*DY-AY/2.
      ZS=0.
      AUGYY=0.
      GO TO 5
    4 AUGY=(I-1)*ARCYINC-ARCYH
      YS=RY*SINF(AUGY)
      ZS=RY*COSF(AUGY)-RY
      AUGYY=AUGY
    5 DO 10 J=1,NCX
      LL=LL+1
      IF(RX.GT.0.)GO TO 6
      XS=(J-1)*DX-AX/2.
      ZXS=0.
      AUGXX=0.
      GO TO 7
    6 AUGX=(J-1)*ARCXINC-ARCXH
      XS=RX*SINF(AUGX)
      ZXS=RX*COSF(AUGX)-RX
      AUGXX=AUGX
    7 W(LL,1)=XS
      W(LL,2)=YS
      W(LL,3)=ZS+ZXS
```

* When the array elements are uniformly spaced.

```
      IF(LP.GT.0)GO TO 1
      G(LL,1)=PIH+AUGX
      G(LL,2)=AUGYY
      G(LL,3)=PIH
      GO TO 10
1     G(LL,1)=PIH-AUGY
      G(LL,2)=AUGXX
      G(LL,3)=0.
10    CONTINUE
      NE=LL
      RETURN
      END
```

```
      SUBROUTINE ELAGPO(NE,LP)
C     THIS SUBROUTINE IS FOR A DOUBLY CURVED SURFACE
      COMMON/2/G(1000,3),W(1000,3)
      COMMON/6/LG(3),A(3),B(3)
C     GROUND PLANE IS ON XZ PLANE
      LG(1)=LG(3)=0
      LG(2)=1
C     NRY,NO OF ROWS IN Y DIRECTION
C     NCX, NO OF COLUMN IN X DIRECTION
C     AX, X-DIRECTION APERTURE, IN WAVELENGTHS
C     AY, Y-DIRECTION APERTURE, IN WAVELENGTHS
C     RX, RADIUS OF CURVATURE IN X DIRECTION
C     RY, RADIUS OF CURVATURE IN Y DIRECTION
      PI=3.14159265358979323846
      PIH=PI/2.
C     LP=0, HORIZONTAL DIPOLES, LP=1, VERTICAL DIPOLES
C     THE FIRST ELEMENT STARTED AT -X AND -Y
      READ 100,NCX,NRY,AX,AY,RX,RY,LP
 100  FORMAT(2I5,4F10.6,I2)
      PRINT 101,NCX,NRY,AX,AY,RX,RY,LP
 101  FORMAT(10X,2I5,4F10.4,I5)
      IF(NRY-1.EQ.0) GOTO8
      DY=AY/(NRY-1)
      GOTO 20
    8 DY=0.0
   20 IF(NCX-1.EQ.0) GOTO 9
      DX=AX/(NCX-1)
      GOTO 22
    9 DX=0.0
   22 AUGY=AUGX=0.0
      LL=0
      DO 10 I=1,NRY
      YS=(I-1)*DY-AY/2.0
      IF(RY.GT.0.) GOTO 4
      ZS=0.0
      AUGYY=0.0
      GOTO 5
    4 AUGY=ASINF(YS/RY)
      ZS=RY*COSF(AUGY)-RY
      AUGYY=AUGY
    5 DO 10 J=1,NCX
      LL=LL+1
      XS=(J-1)*DX-AX/2.0
      IF(RX.GT.0.) GOTO 6
      ZXS=0.0
      AUGXX=0.0
      GOTO 7
    6 AUGX= ASINF(XS/RX)
      ZXS=RX*COSF(AUGX)-RX
      AUGXX=AUGX
 7    W(LL,1)=XS
      W(LL,2)=YS
      W(LL,3)=ZS+ZXS
      IF(LP.GT.0)GO TO 1
      G(LL,1)=PIH+AUGX
      G(LL,2)=AUGYY
      G(LL,3)=PIH
      GO TO 10
 1    G(LL,1)=PIH-AUGY
```

*When the array elements are projected uniformly spaced.

```
      G(LL,2)=AUGXX
      G(LL,3)=0.
10    CONTINUE
      NE=LL
      RETURN
      END
```

```
      SUBROUTINE ELAGPO(NE,LP)
C     THIS SUBROUTINE IS FOR CONIC ARRAY
      COMMON/2/G(1000,3),W(1000,3)
      COMMON/6/LG(3),A(3),B(3)
C     MM, NUMBER OF ROWS
C     RB, BASE RADIUS
C     ARC, CONE ARCH
C     TC CONE ANGLE
C     DX, SPACING IN X-DIRECTION
C     DY, SPACING IN Y DIRECTION
C     LP=0, HORIZONTAL DIPOLES, LP=1, VERTICAL DIPOLES
C     LRT=1, TRIANGULAR GRID
C     LRT=0, RECTANGULAR GRID
      READ 100,MM,RB,ARC,TC,DX,DY,LP,LRT
 100  FORMAT(I5,5F10.6,2I2)
      PRINT 101,MM,RB,ARC,TC,DX,DY,LP,LRT
 101  FORMAT(10X,I5,5F10.4,2I5)
C     GROUND PLANE IS ON XZ PLANE
      LG(1)=LG(3)=0
      LG(2)=1
 2    PI=3.14159265358979323846
      PIH=PI/2.
      ATR=PI/180.
      TC=TC*ATR
      ARC=ARC*ATR
      ARCH=ARC/2.
      TTC=TANF(TC)
      I=0
      DO 10 M=1,MM
      IMOD=MOD(M-1,2)*LRT
      R=RB-(M-1)*DY*TTC
      RL=R*ARC
      D=R*SINF(ARCH)
      NN=RL/DX+1
      ARCINC=ARC/(NN-1)
      NN=NN-IMOD
      DO 10 J=1,NN
      I=I+1
      AUG=(1.-J-.5*IMOD)*ARCINC+ARCH
      W(I,1)=COSF(PIH-AUG)*R
      W(I,2)=DY*(M-1)
      W(I,3)=SINF(PIH-AUG)*R-D
      IF(LP.GT.0)GO TO 3
      G(I,1)=AUG+PIH
      G(I,2)=TC
      G(I,3)=PIH
      GO TO 10
 3    G(I,1)=PIH-TC
      G(I,2)=AUG
      G(I,3)=0.
 10   CONTINUE
      NE=1
      RETURN
      END
```

```
      SUBROUTINE CODTRF(L)
      COMMON/1/PLTARRAY(254),FMAT(1000,3)
      COMMON/2/G(1000,3),W(1000,3)
      COMMON/3/RMAT(1000,3,3),DMAT(3,3),EMAT(3),TEMP(3)
      DIMENSION A(3,3),B(3,3),C(3,3)
      COX=COSF(G(L,1))
      COY=COSF(G(L,2))
      COZ=COSF(G(L,3))
      SIX=SINF(G(L,1))
      SIY=SINF(G(L,2))
      SIZ=SINF(G(L,3))
C FIND THE X RATATION MATRIX
      C(1,1)=1.
      C(1,2)=C(1,3)=C(2,1)=C(3,1)=0.
      C(2,2)=C(3,3)=COX
      C(3,2)=SIX
      C(2,3)=-SIX
C FIND THE Y ROTATION MATRIX
      B(1,1)=B(3,3)=COY
      B(1,3)=SIY
      B(3,1)=-SIY
      B(1,2)=B(2,1)=B(2,3)=B(3,2)=0.
      B(2,2)=1.
C FORM MATRIX PRODUCT
      DO 10 I=1,3
      DO 10 J=1,3
      A(I,J)=0.
      DO 10 K=1,3
 10   A(I,J)=A(I,J)+B(I,K)*C(K,J)
C FORM Z-AXIS ROTAT+ON MATRIX
      C(1,1)=C(2,2)=COZ
      C(2,1)=SIZ
      C(1,2)=-SIZ
      C(1,3)=C(3,1)=C(3,2)=C(2,3)=0.
      C(3,3)=1.
      DO 20 I=1,3
      DO 20 J=1,3
      RMAT(L,I,J)=0.
      DO 20 K=1,3
 20   RMAT(L,I,J)=RMAT(L,I,J)+C(I,K)*A(K,J)
      RETURN
      END
```

```
SUBROUTINE DPRMAT(COFI,SIFI,COTA,SITA,DMAT)
DIMENSION DMAT(3,3)
DMAT(1,1)=SITA*COFI
DMAT(1,2)=COTA*COFI
DMAT(1,3)=-SIFI
DMAT(2,1)=SITA*SIFI
DMAT(2,2)=COTA*SIFI
DMAT(2,3)=COFI
DMAT(3,1)=COTA
DMAT(3,2)=-SITA
DMAT(3,3)=0.
RETURN
END
```

```
      SUBROUTINE ANGTRF(COFI,SIFI,COTA,SITA,L,LJG)
      COMMON/3/RMAT(1000,3,3),DMAT(3,3),EMAT(3),TEMP(3)
      COMMON/6/LG(3),A(3),B(3)
      LJG=1
      B(1)=SITA*COFI
      B(2)=SITA*SIFI
      B(3)=COTA
      DO 20 I=1,3
      A(I)=0.
      DO 10 K=1,3
10    A(I)=A(I)+RMAT(L,K,I)*B(K)
      IF(A(I).GE.0. .OR.LG(I).LT.1)GO TO 20
      LJG=0
      GO TO 9
20    CONTINUE
      COTA=A(3)
      IF(ABS(COTA).GT.1.)COTA=SIGN(1.,A(3))
      SITA=SQRT(1.-COTA**2)
      IF(ABS(A(1)).GT.1.E-10)GO TO 1
      COFI=0.
      SIFI=1.
      GO TO 2
1     TGFI=A(2)/A(1)
      COFI=SQ-T(1./(1.+TGFI**2))
      SIFI=SQRT(1.-COFI**2)
2     COFI=SIGNF(COFI,A(1))
      SIFI=SIGNF(SIFI,A(2))
99    RETURN
      END
```

```
SUBROUTINE ELPAT(COFI,SIFI,COTA,SITA,LP)
COMMON/3/RMAT(1000,3,3),DMAT(3,3),EMAT(3),TEMP(3)
EMAT(1)=0.
EMAT(2)=-SITA*(LP-1)
EMAT(3)=SITA*LP
RETURN
END
```

```
      SUBROUTINE MATMUT(L)
      COMMON/1/PLTARRAY(254),FMAT(1000,3)
      COMMON/3/RMAT(1000,3,3),DMAT(3,3),EMAT(3),TEMP(3)
      COMMON/5/DMATUP(3,3)
      DO 10 I=1,3
      TEMP(I)=0.
      DO 10 J=1,3
10    TEMP(I)=TEMP(I)+DMAT(I,J)*EMAT(J)
      DO 20 I=1,3
      EMAT(I)=0.
      DO 20 J=1,3
20    EMAT(I)=EMAT(I)+RMAT(L,I,J)*TEMP(J)
      DO 30 I=1,3
      TEMP(I)=0.
      DO 30 J=1,3
30    TEMP(I)=TEMP(I)+DMATUP(J,I)*EMAT(J)
      RETURN
      END
```

```
      SUBROUTINE FRAME(XM,YM,XSL,YSL,NX,NY,TAPI)
C NX,NUMBER OF DIVISIONS ON X-AXIS
C NY,NUMBER OF DIVISIONS ON Y-AXIS
C YM,MAX LENGTH OF Y-AXIS IN INCH
C XM,MAX LENGTH OF X-AXIS IN INCH
C SY=SHIFT OF ORIGIN ON Y-AXIS
C HN= HEIGHT OF LABELING CHARACTER IN MULTIPLES OF 0.035
CXSL,YSL-X-AXIS,Y-AXIS SCALE
C LL=1,FOR DB SCALE,LL=0,FOR ABSOLUTE VALUE
      COMMON/1/PLTARRAY(254),FMAT(1000,3)
      SY=2.
      HN=5.
      YMSY=YM+SY
      HLAB=HN*.035
      HLAS=HLAB+.035
      WLAB=4.*HLAB/7.
      XSCL=XSL/NX
      YSCL=YSL/NY
      DY=YM/NY
      Y=SY
      NNY=NY+1
      CALL PENCHG(12)
      CALL PLOT(0.,SY,3)
      CALL PLOT(XM,SY,2)
      CALL PLOT(XM,YMSY,2)
      CALL PLOT(0.,YMSY,2)
      CALL PLOT(0.,SY,2)
      CALL PENCHG(11)
      DO 10 J=1,NNY
      CALL PLOT(0.,Y,3)
      MODY=MOD(J-1,1
      IF(MODY.NE.0)GO TO 11
      CALL PLOT(-.2,Y,2)
      A=YSCL*(J-1)-YSL
      CALL NUMBER(-7.5*WLAB,Y-HLAB/2.,HLAB,A,0.,4HF4.0)
      GO TO 10
 11   CALL PLOT(-.1,Y,2)
 10   Y=Y+DY
      X=0.
      DX=XM/NX
      NNX=NX+1
      DO 20 K=1,NNX
      CALL PLOT(X,SY,3)
      MODX=MOD(K-1,10)
      IF(MODX.NE.0)GO TO 21
      CALL PLOT(X,SY-.2,2)
      A=(K-1)*XSCL+TAPI
      CALL NUMBER(X-3.5*WLAB,SY-HLAB*2.5,HLAB,A,0.,4HF4.0)
      GO TO 20
 21   CALL PLOT(X,SY-.1,2)
 20   X=X+DX
      CALL SYMBOL(.5*XM-20.5*WLAB,-5.*HLAB+SY,HLAS,22HAZIMUTH ANGLE(DEGR
     CEES),0.,22)
 33   CALL SYMBOL (-9.0*WLAB,YM/2.+SY-13.7*WLAB,HLAS,17HARRAY PATTERN(DB
     C),90.,17)
 32   CALL PLOT(0.,0.,3)
      END
```

```
SUBROUTINE PENCHG(IP)
RETURN
END
```