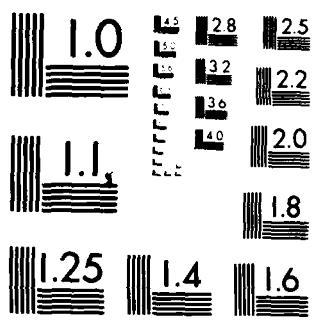
AD-A131 060 A FORTRAN COMPUTER PROGRAM TO COMPUTE THE RADIATION PATTERN OF AN ARRAY-FED PARABOLOID REFLECTOR(U) NAVAL RESEARCH LAB WASHINGTON DC J K HSIAO 21 JUL 83 1/1

UNCLASSIFIED NRL-8740 F/G 9/2 NL

END
DATE
FILMED
8 83
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

②

NRL Report 8740

A FORTRAN Computer Program to Compute the Radiation Pattern of an Array-Fed Paraboloid Reflector

J. K. HSIAO

Electromagnetics Branch
Radar Division

July 21, 1983

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 8740

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)
A FORTRAN COMPUTER PROGRAM TO COMPUTE THE RADIATION PATTERN OF AN ARRAY-FED PARABOLOID REFLECTOR

5. TYPE OF REPORT & PERIOD COVERED
Interim report on a continuing NRL problem

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)
James K. Hsiao

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS
Naval Research Laboratory
Washington, DC 20375

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
62712N; XF12-141-100
53-1854-0

11. CONTROLLING OFFICE NAME AND ADDRESS
Naval Electronic Systems Command
Washington, DC 20360

12. REPORT DATE
July 21, 1983

13. NUMBER OF PAGES
15

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

15. SECURITY CLASS. (of this report)

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)
Antenna
Reflector antenna
Antenna pattern

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

This report describes a computer program to compute the radiation pattern of a paraboloid reflector fed by an array. The program is formulated without simplification and approximation assumptions. Its results should represent an exact solution. The program is designed so that it is efficient for multiple sources. Judging from the examples computed, the increase in computer time is modest when the number of source elements increases.

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

A FORTRAN COMPUTER PROGRAM TO COMPUTE THE RADIATION PATTERN OF AN ARRAY-FED PARABOLOID REFLECTOR

INTRODUCTION

Computing the radiation pattern of a paraboloid reflector has been treated widely in literature [1-4]; however, most of this treatment concerns only the problem of single-feed source whether the source is at the focal point or offset. In all these treatments, when the source is offset, it is always assumed that the offset is very small and that the distance from the source to a point on the reflector surface can be approximated by adding a few linear terms to the case of the feed at the focal point. This approximation may be acceptable in a single-source case when the feed point usually is located close to the focal point to avoid high sidelobes. However, in the case of an array-fed system, this assumption may not be valid.

To facilitate fast computation, the longitudinal current component is ignored in this computation [1] The error introduced by this omission, in general, is small in the region close to the main beam. However, when fields far from the main beam are of interest, the amount of phase error can not be ignored. Galindo-Israel and Mittra [3] introduced a method to correct this error; however, this correction depends on the location of the offset feed. It probabily would not be efficient for a multiple-feed case.

In this report a computer program to compute the radiation pattern of an array-fed paraboloid reflector is presented. This program is written so that it is efficient for multiple-feed sources and the solution will be exact. No simplification or other approximating assumption is made.

The purpose of developing this computer program is threefold.

- It will be used as a reference to check the validity and efficiency of other programs which may be developed using a more simplified approach.
- It is an interim step to develop a computer program to compute the radiation pattern of a dual-reflector system which is currently under a feasibility study at NRL.
- This program may be used to investigate the pattern-synthesis method for an array-fed reflector antenna system.

MATHEMATICAL FORMULATION

Figure 1 shows an array-fed paraboloidal reflector antenna system. The radiation field at a point Φ and θ has the form

$$\vec{E}(\Phi, \theta) = C \int_a \sum_n \vec{J}_n(\theta, \phi) \frac{1}{\rho_n'} \exp[-jk(\rho_n' - \hat{\rho} \cdot \hat{R})] \, da, \tag{1}$$

Manuscript approved May 9, 1983.

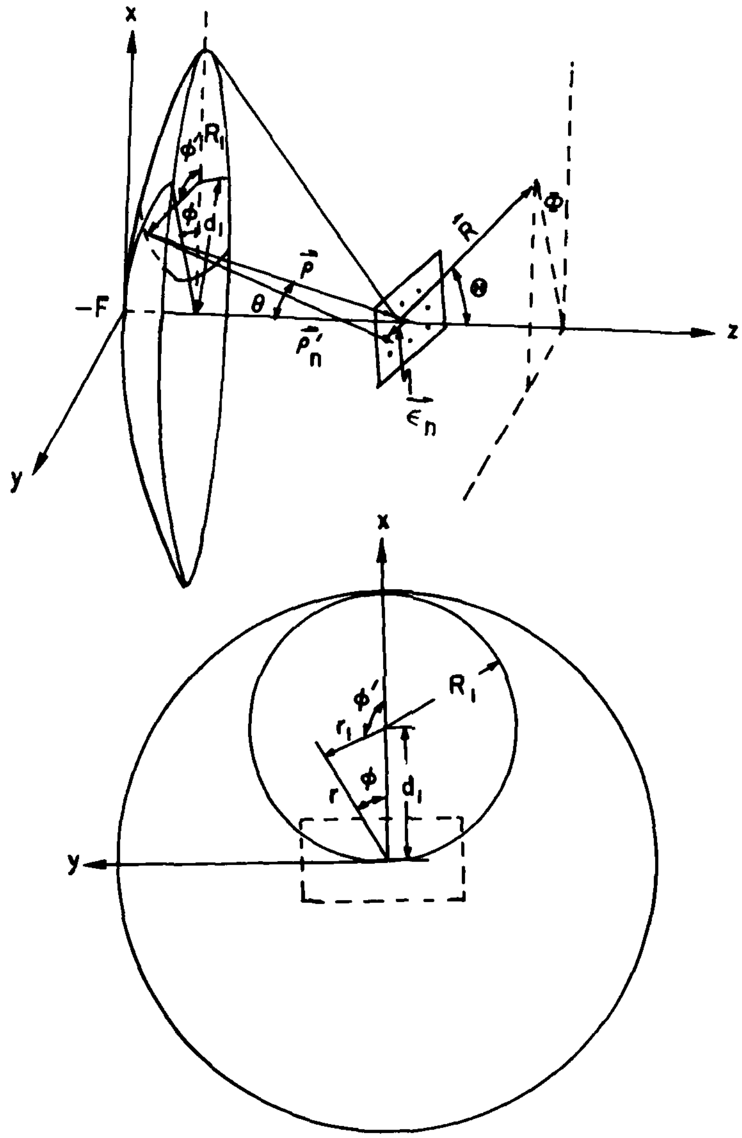


Fig. 1 — Configuration of an array-fed parabaloid reflector

where $J_j(\theta, \phi)$ is the current density induced on the surface of the reflector and ρ'_n is the distance for the nth array feed element to a point on the surface, and

$$\vec{\rho}'_n = \vec{\rho} - \vec{\epsilon}_n, \tag{2}$$

where ϵ_n is a position vector from the reference point to the location of the nth array element. For convenience, the reference point is assumed to be at the focal point of the parabola. Unit vector R points the direction from the current source to a field point which is a function of Φ and θ.

Assume that only part of the paraboloid is used. This used part is known to be a circle at the upper half of the reflector (see Fig. 1). The center of this circle is at a distance d_1 above the z-axis. Any point on this circle can be represented by

$$x = \rho \sin\theta \cos\phi = r \cos\phi \tag{3a}$$

$$y = \rho \sin\theta \sin\phi = r \sin\phi. \tag{3b}$$

In terms of the coordinates of the ϕ' and r_1 of this circle,

$$x = d_1 + r_1 \cos\phi'$$

and (4)

$$y = r_1 \sin\phi'.$$

The position vector from a feed element in the array to a point (x, y, z) on the reflector surface is

$$\vec{\rho}'_n = \frac{(x - \epsilon_x)}{|\rho'_n|}\hat{x} + \frac{(y - \epsilon_y)}{|\rho'_n|}\hat{y} + \frac{(z - \epsilon_z)}{|\rho'_n|}\hat{z}$$

$$= \rho'_{nx}\hat{x} + \rho'_{ny}\hat{y} + \rho'_{nz}\hat{z}. \tag{5}$$

The paraboloid surface can be described by the following equation:

$$x^2 + y^2 = 4f(f + z) \tag{6}$$

where f is the focal length of the parabola.

The surface currents $\vec{J}_n$ are related to the feed element exciting field by the following relation:

$$\vec{J}_n = \hat{n} \times \vec{\rho}'_n \times \hat{e}_n, \tag{7}$$

where $\hat{e}_n$ is a unit vector in the direction of the tangential field on the surface of the paraboloid which is excited by the nth array element and

$$\hat{e}_n = \hat{\rho}'_n \times \hat{u}_n \times \hat{\rho}'_n / |\hat{\rho}'_n \times \hat{u}_n \times \hat{\rho}'_n|, \tag{8}$$

where $\hat{u}_n$ is a unit vector in the direction of the field in the nth array element. Assume that the array elements and paraboloid have the same coordinates, and that

$$\hat{u}_n = u_{nx}\hat{x} + u_{ny}\hat{y} + u_{nz}\hat{z} \tag{9}$$

where $|\hat{u}_n| = 1$.

The normal vector $\hat{n}$ in Eq. (7) for a paraboloid surface has the following form:

$$\hat{n} = \frac{x}{|n|}\hat{x} - \frac{y}{|n|}\hat{y} + \frac{2f}{|n|}\hat{z}$$

$$= + n_x\hat{x} + n_y\hat{y} + n_z\hat{z}. \tag{10}$$

Appendix A shows that

$$\vec{J}_n = \frac{1}{\sqrt{1-\alpha^2}} \{[-u_{nx}(n_y\rho'_{ny} + n_z\rho'_{nz}) + u_{ny}\rho'_{nx}n_y + u_{nz}\rho'_{nx}n_z]\hat{x}$$

$$+ [u_{nx}\rho'_{ny}n_x - u_{ny}(n_x\rho'_{nx} + n_z\rho'_{nz}) + u_{nz}\rho'_{nz}n_y n_z]\hat{y}$$

$$+ [u_{nx}\rho'_{nx}n_x + u_{ny}n_y\rho'_{nz} - u_{nz}(n_x\rho'_{nx} + n_y\rho'_{ny})]\hat{z}\}, \quad (11)$$

where $\alpha = \hat{\rho}'_n \cdot \hat{u}_n = u_{nx}\rho'_{nx} + u_{ny}\rho'_{ny} + u_{nz}\rho'_{nz}$.

The u_{nx}, u_{ny} and u_{nz} are the normalized excitation field at the nth element in the feed array. Usually, the elements are linear polarized either in the x or in the y direction. In this case only one component will be equal to unity.

The $\hat{\rho}'_n$ components are

$$\rho'_{nx} = (d_1 + r_1 \cos\phi' - \epsilon_{nx})/|\rho'_n|, \quad (12a)$$

$$\rho'_{ny} = (r_1 \sin\phi' - \epsilon_{ny})/|\rho'_n|, \quad (12b)$$

$$\rho'_{nz} = \left\{-\frac{1}{4f}[4f^2 - (d_1^2 - 2d_1r_1\cos\phi' + r_1^2)] - \epsilon_{nz}\right\}/|\rho'_n|, \quad (12c)$$

and components of $\hat{n}$ vector are

$$n_x = -(d_1 + r_1\cos\phi')/|n|, \quad (13a)$$

$$n_y = -r_1\sin\phi'/|n|, \quad (13b)$$

$$n_z = 2f/|n|, \quad (13c)$$

where $|n| = (d_1^2 + 2d_1r_1\cos\phi' + r_1^2 + 4f^2)^{1/2}$.

When inserting these relations into Eq. (11), on finds that the current $\vec{J}_n$ is a function of r_1 and ϕ' which will be used as the integration variables. The vector $\hat{\rho}$ has the following form:

$$\hat{\rho} = \left\{(d_1 + r_1\cos\phi')\hat{x} + r_1\sin\phi\hat{y} - \frac{1}{4f}[4f^2 - (d_1^2 - 2d_1r_1\cos\phi' + r_1^2)]\hat{z}\right\}/|\rho| \quad (14)$$

and

$$\hat{R} = \sin\Theta\cos\Phi\hat{x} + \sin\Theta\sin\Phi\hat{y} + \cos\Theta\hat{z}. \quad (15)$$

Therefore,

$$\hat{\rho}\cdot\hat{R} = \{(d_1 + r_1\cos\phi')\sin\Theta\cos\Phi + r_1\sin\phi'\sin\Theta\sin\Phi$$

$$- \frac{1}{4f}[4f^2 - (d_1^2 - 2d_1r_1\cos\phi' + r_1^2)\cos\Theta\}/|\rho|. \quad (16)$$

Insert this relation into Eq. (1) and perform the integration. The electric field has the following form:

$$\vec{E}(\Theta, \Phi) = C\int\int\sum_n \vec{J}_n(r_1, \phi')\frac{1}{\rho'_n}\exp[-jk(\rho'_n - \hat{\rho}\cdot\hat{R})]r_1dr_1d\phi'. \quad (17)$$

The Θ and Φ components are

$$E_\Theta = \cos\Theta(\cos\Phi E_x + \sin\Phi E_y) - \sin\Phi E_z \quad (18a)$$

and

$$E_\Phi = -\sin\Phi E_x + \cos\Phi E_y, \tag{18b}$$

where E_x, E_y, and E_z are the x, y, and z components of $\vec{E}(\Theta, \Phi)$ in Eq. (17).

COMPUTER PROGRAM

The preceding formulation is coded into a FORTRAN computer program which has been tried on the Texas Instruments ASC computer. Appendix B lists that program. The FORTRAN program can compute the ratiation pattern of a reflector which is only a portion of paraboloid. This portion is a circle, has a radius r_1 and center at d, as shown in Fig. 1. The integration is then computed by integrating the ϕ angle from 0 to 2π and the radius r_1 from 0 to a desired value. The first data card has a I5 fixed point format which specifies the following parameters:

- IEND is the number of points to be integrated in the radius r_1 direction
- JEND is the number of points to be integrated in the ϕ' direction (from 0 to 2π radians).
- NEL is the number of feed array elements.
- NPT is the number of feed point to be computed at a given Φ angle.
- NPOL specifies the feed element polarization, if NPOL = 1, the excitation field is polarized at y-direction. If NPOL = 0, the polarization is at x-direction.

The second data card has an 8F10.6 floating point format.

- FO is the focal length.
- RSO is the radius of the complete paraboloid.
- D1 is the center of the circle of the portion of the paraboloid to be used.
- A1 is the radius of the circle of the portion of the paraboloid to be used.
- FIC is the Φ angle in degrees.
- ARAG is the range of Θ angel to be plotted in degrees.

All of the preceding data, except angles, are in terms of wavelength.

The next few data cards specify the location of the array elements in x, y, and z directions. These cards are all in F10.6 form; each contains 8 data. Therefore, the number of cards required depends on the number of array elements. For example, if the array has 24 elements, three cards are required to specify the locations of each coordinate. Nine cards are then needed. The sequence is that first read in the x-component locations, then the y, finally the z-locations. The output of this program is a plotted antenna pattern for E_Θ and E_Φ components at a cut of Φ angle specified by FIC and plotted a range from 0 degrees to ARAG degrees. However, if ARAG is negative, then it will plot from $-$ARAG to +ARAG.

A function subroutine PATF is included in this program. This function specifies the array element pattern in its own coordinates sin θ', and cos ϕ'. At the presented time, the subroutine function is set at an isotropic element pattern. The user may change it to fit his requirement.

EXAMPLES

Figures 2a and 2b show the radiation pattern of a paraboloid dish with a 13.88-wavelength radius and a 25.9-wavelength focal length. A single source located at the focal point is used in both plots. Figure 2a is plotted by use of the method developed by Galindo-Israel [2]. This method uses a Jacobian series to expand the integral of Eq. (1) into a series of functions in terms of Φ and Θ. This approach should be efficient in computation time. However, when the series is expanded for better convergence, the transformation must be performed at the vicinity of the center of the main beam. Since the main beam position is a function of the sources' location, this program is not necessarily efficient when multiple sources are used. Figure 2b is plotted by use of the method described in the Mathematical Formulation section of this report. The patterns on both figures are exactly the same. The central processing time for Fig. 2a is 41.6 s and that for Fig. 2b is 52.46 s. Galindo-Israel's method is about 30% more efficient.

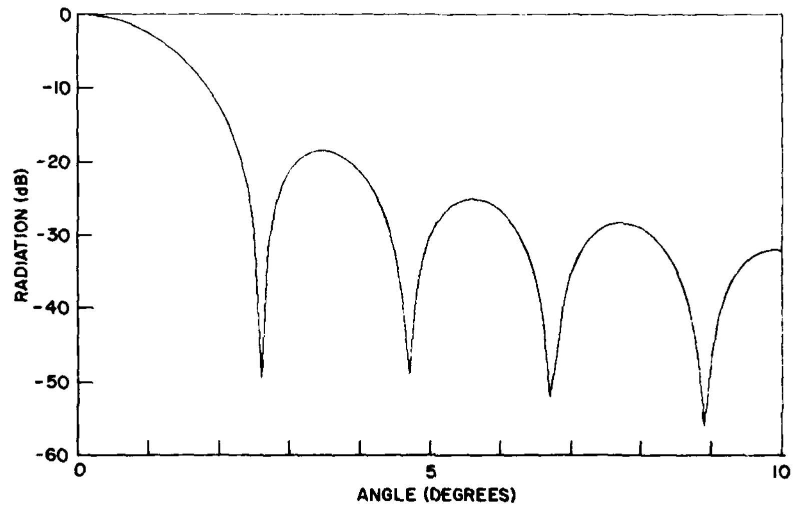


Fig. 2a — Radiation pattern of a paraboloid reflector computed by the series expansion method

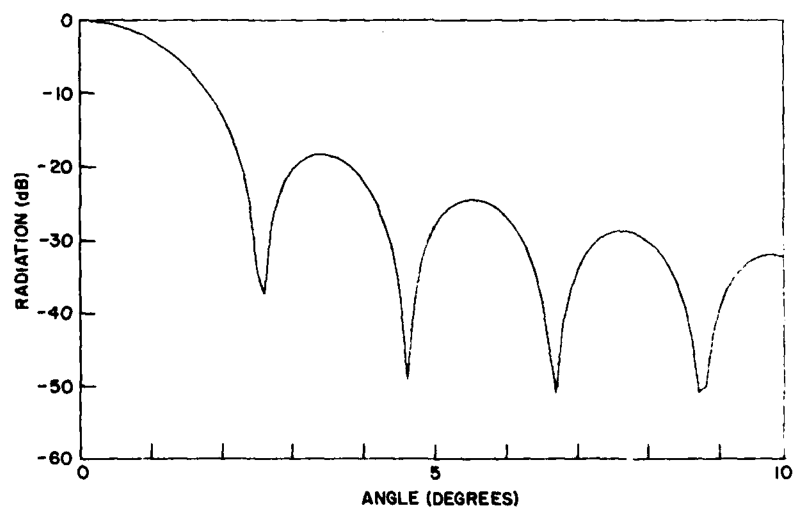


Fig. 2b — Radiation pattern of a paraboloid reflector computed by the exact solution

Next, we tried an offset case. The results are shown in Fig. 3. The required central process times for our program and the Galindo-Israel program are about the same as that in Figs. 2a and 2b. Two cases of a two-source and a five-source are tried. Their patterns are shown in Fig. 4. The central process time is 54.73 s for the two-source case and 57.68 s for the five-source case. There is no way to compare with Galindo-Israel's method because that program is not designed for multiple-feed sources.

From these examples, the increase of computer time for multiple sources is reasonable. This program probably is efficient in this sense.

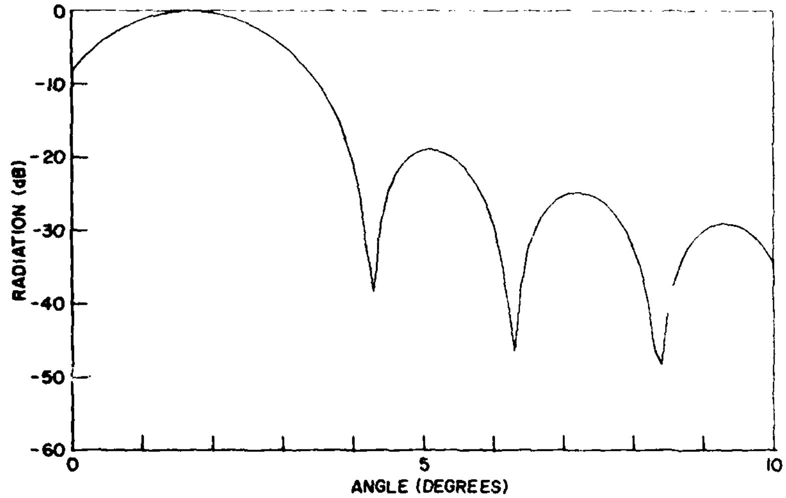


Fig. 3 — Radiation pattern of a paraboloid reflector—off-set feed

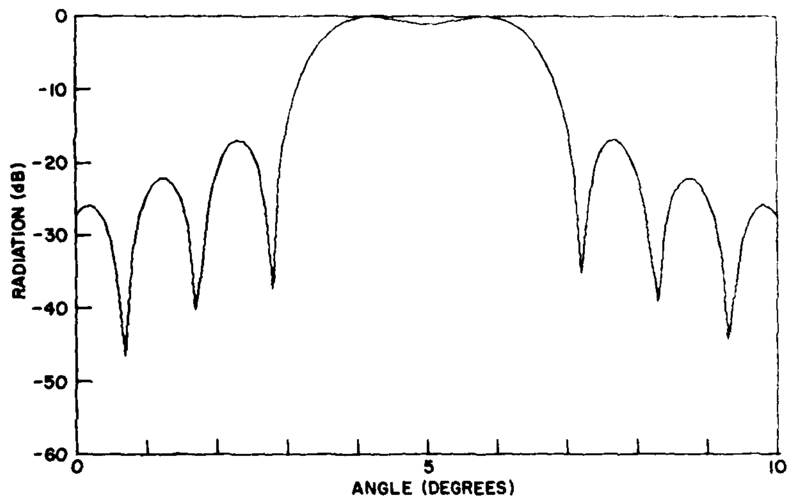


Fig. 4a — Radiation pattern of a paraboloid reflector—two multiple-source off-set feed

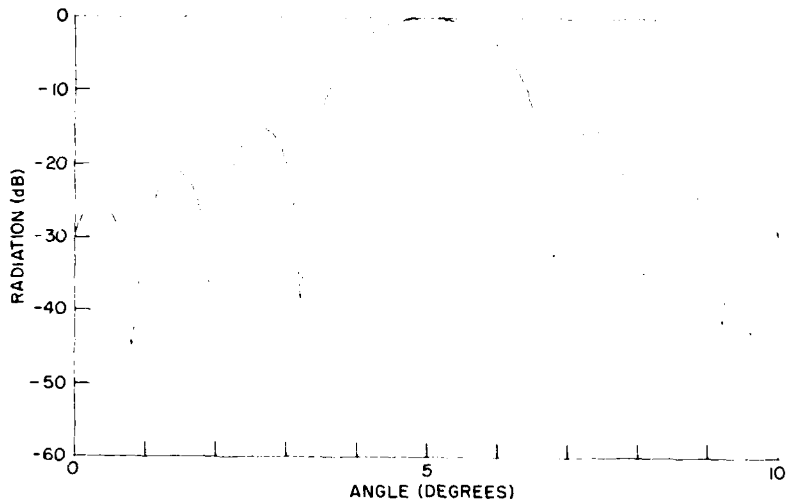


Fig. 4b — Radiation pattern of a paraboloid reflector—five multiple-source offset feed

CONCLUSIONS

A computer program which computes the radiation pattern of a paraboloid reflector with multiple feed sources is presented. From some computer exampes, the program is efficient for a large number of feed sources.

REFERENCES

1. S. Silver, ed., "Microwave Antenna Theory and Design," in *Radiation Laboratory Series* (McGraw-Hill, 1949), Vol. 12.

2. J. Ruze, "Lateral-Feed Displacement in a Paraboloid," *IEEE Trans. Antennas Propag.*, Sept. 1965.

3. V. Galindo-Israel and R. Mittra, "A New Series Representation for the Radiation Integral with Application to Reflector Antennas," *IEEE Trans. Antennas Propag.*, Sept. 1977.

4. R. Mittra, Y. Rahmat-Samii, V. Galindo-Israel, and R. Norman, "An Efficient Technique for the Computation of Vector Secondary Pattern of Offset Paraboloid Reflectors," *IEEE Trans. Antennas Propag.*, May 1979.

Appendix A

DERIVATION OF EQUATION (11)

$$\hat{\rho}_n' = \rho_{nz}'\hat{x} + \rho_{ny}'\hat{y} + \rho_{nz}'\hat{z},$$
$$\hat{u}_n = n_{nx}\hat{x} + u_{ny}\hat{y} + u_{nz}\hat{z},$$
$$\hat{n} = n_x\hat{x} + n_y\hat{y} + n_z\hat{z},$$

where

$$\rho_{nx}'^2 + \rho_{ny}'^2 + \rho_{nz}'^2 = 1,$$
$$u_{nx}^2 + u_{ny}^2 + u_{nz}^2 = 1,$$
$$n_x^2 + n_y^2 + n_z^2 = 1,$$

and vectors $\hat{\rho}_n'$, $\hat{u}_n$, and $\hat{n}$ are defined in Eq. (5), Eq. (9), and Eq. (10) respectively.

$$\vec{e}_n = \hat{\rho}_n' \times \hat{u}_n \times \hat{\rho}_n',$$
$$= \hat{u}_n - \hat{\rho}_n'(\hat{\rho}_n' \cdot \hat{u}_n).$$

Let

$$\hat{\rho}_n' \cdot \hat{u} = (u_{nx}\rho_{nx}' + u_{ny}\rho_{ny}' + u_{nz}\rho_{nz}') = \alpha,$$
$$\vec{e}_n = (u_{nx} - \alpha\rho_{nx}')\hat{x} + (u_{ny} - \alpha\rho_{ny}')\hat{y} + (u_{nz} - \alpha\rho_{nz}')\hat{z},$$
$$|e_n| = \sqrt{1 - \alpha^2},$$
$$\hat{e}_n = \frac{\vec{e}_n}{\sqrt{1 - \alpha^2}},$$
$$\vec{J}_n = \hat{n} \times \hat{\rho}_n' \times \hat{e}_n$$
$$= \hat{n} \times \hat{\rho}_n' \times \hat{u}_n/\sqrt{1 - \alpha^2}$$
$$= \frac{\hat{\rho}_n'}{\sqrt{1 - \alpha^2}}(\hat{n} \cdot \hat{u}_n) - \frac{\hat{u}_n}{\sqrt{1 - \alpha^2}}(\hat{n} \cdot \hat{\rho}_n')$$
$$= \frac{1}{\sqrt{1 - \alpha^2}}\{[-u_{nx}(n_y\rho_{ny}' + n_z\rho_{nz}') + u_{ny}\rho_{nx}'n_y + u_{nz}\rho_{nx}'n_z]\hat{x}$$
$$+ [u_{nx}\rho_{ny}'n_x - u_{ny}(n_x\rho_{nx}' + n_z\rho_{nz}) + u_{nz}\rho_{ny}'n_z]\hat{y}$$
$$+ [u_{nx}\rho_{nz}'n_x + u_{ny}\rho_{ny}' - u_z(n_x\rho_{nx}' + n_y\rho_{ny}')]\}\hat{z}.$$

Appendix B

COMPUTER PROGRAM LIST

```
          SOURCE LISTING            ASC FAST FORTRAN COMPILER        RELEASE FTFX05PAT

ISN          STATEMENT              CP OPTIONS = (B,D,E,K,M,V)      DATE = 06/24/82 ISN

                PROGRAM REFPAT
                COMMON/D/ELDL(3,100)
                COMPLEX PFI,PP,PSUM,PC
                DIMENSION PF1(3,500),PSUM(3,500),PAT(3,500),PP(3),JSP(500)
                DIMENSION ERAS(500),XX(500),YY(500)
                DIMENSION SITAC(200),COTAC(200),SIFI(200),COFI(200)
                DIMENSION EPLEN(100),FJX(100),FJY(100),FJZ(100)
        C       THIS PROGRAM COMPUTES THE RADIATION PATTERN OF OFF-SET REFLECTOR
        C          WITH EXACT SOLUTION, REFLECTOR CAN BE PORTION OF THE REFLECTOR
        C          OF A PRABOLOID AND IT CAN HAVE MORE THEN ONE RADIATION SOURCE
                CALL R$STOP
                READ 100,IEND,JEND,NEL, NPT,NPOL
          100   FORMAT(16I5)
        C       IEND, NUMBER OF POINTS INTEGRATED IN R DIRECTION
        C       JEND, NUMBER OF POINTS INTEGRATED IN FI DIRECTION
        C       IEND AND JEND MUST BE ODD
        C       NPT, NUMBER OF FIELD POINTS IN THETA DIRECTION
        C       NEL, NUMBER OF ARRAY ELEMENTS
        C       NPOL=1, Y -POLARIZATION,X=0, X-POLARIZATION
                READ 101,FO,RSO,D1,A1,FIC,ARAG
          101   FORMAT(8F10.6)
        C       FO, REFLECTOR FOCAL LENGTH(INWAVELENGTH)
        C       RSO, REFLECTOR RADIUS IN WAVELENGTH
        C       D1, CENTER OF THE CIRCLE OF PORTION OF THE REFLECTOR USED
        C       A1, RADIUS OF THE USED REFLECTOR CIRCLE
        C       FIC, FIELD POINT AT FIC PLANE
        C       ARAG ANGLE RANGE
        C       IF ARAG.LT.0. PLOT -ARAG TO ARAG
                PRINT 102,IEND,JEND,NPT,NEL,NPOL
          102   FORMAT(10X,16I5)
                PRINT 103,FO,RSO,D1,A1,FIC, ARAG
          103   FORMAT(10X,9F10.4)
                PI=3.1415926536
                ATR=PI/180.
                PI2=PI*2.
                FIC=FIC*RTA
                COFIC=COS(FIC)
                SIFIC=SIN(FIC)
                XSL=8.
                YSL=5.
                IF(ARAG.GT.0.)GO TO 3
                ARAG2=-ARAG*2.
                TA=ARAG*ATR
                AST=ARAG
                ASF=-ARAG
                GO TO 4
            3   ARAG2=ARAG
                TA=0.
                AST=0.
                ASF=ARAG
```

```
AT          SOURCE LISTING          ASC FAST FORTRAN COMPILER       RELEASE FTFX05

ISN         STATEMENT               OP OPTIONS = (B,D,E,K,M,V)      DATE = 06/24/82

      4     TACINC=ARAG2/(NPT-1)
            TAINC=TACINC*ATR
            TAC=0.
            XSCL=XSL/ARAG2
            DO 1 N=1,NPT
            PSUM(1,N)=CMPLX(0.,0.)
            PSUM(2,N)=CMPLX(0.,0.)
            PSUM(3,N)=CMPLX(0.,0.)
            XX(N)=TAC*XSCL
            COTAC(N)=COS(TA)
            SITAC(N)=SIN(TA)
            TA=TA+TAINC
      1     TAC=TAC+TACINC
            TA=0.
            TAINC=PI2/JEND
            DO 2 J=1,JEND
            COFI(J)=COS(TA)
            SIFI(J)=SIN(TA)
            JMOD=MOD(J,2)
            JSP(J)=JMOD*2+(1-JMOD)*4
            IF(J.EQ.1.OR.J.EQ.JEND)JSP(J)=1
      2     TA=TA+TAINC
            FO2=FO*FO
            SINC=A1/IEND
            R1=SINC
     C      ENTER OFF-SET ARRAY ELEMENT LOCATIONS
            READ 101,(ELOC(1,I),I=1,NEL)
            READ 101,(ELOC(2,I),I=1,NEL)
            READ 101,(ELOC(3,I),I=1,NEL)
            PRINT 103,((ELOC(K,I),K=1,3),I=1,NEL)
            DO 10 I=1,IEND
            RDSQ=R1**2+D1**2
            RD1=R1*D1
            DO 11 N=1,NPT
            PFI(1,N)=CMPLX(0.,0.)
            PFI(2,N)=CMPLX(0.,0.)
      11    PFI(3,N)=CMPLX(0.,0.)
            DO 20 J=1,JEND
            R=SQRT(RDSQ+2.*RD1*COFI(J))
            RCOFI=(D1+R1*COFI(J))
            RSIFI=R1*SIFI(J)
            R2=R*R
            FOR2=FO-R2/(4.*FO)
      12    UN=SQRT(R2+4.*FO2)
            PP(1)=CMPLX(0.,0.)
            PP(2)=CMPLX(0.,0.)
            PP(3)=CMPLX(0.,0.)
            DO 22 L=1,NEL
            PX=RCOFI-ELOC(1,L)
            PY=RSIFI-ELOC(2,L)
            PZ=-FOR2-ELOC(3,L)
            PX2=PX**2
            PY2=PY**2
            PZ2=PZ**2
            EPLEN2=PX2+PY2+PZ2
            EPLEN(L)=SQRT(EPLEN2)
            IF(NPOL.GT.0)GO TO 21
            UM=SQRT(PY2+PZ2)
            FJNOR=UN*UM*EPLEN(L)
            FJX(L)=(RSIFI*PY-2.*FO*PZ)/FJNOR
            FJY(L)=-RCOFI*PY/FJNOR
            FJZ(L)=-RCOFI*PZ/FJNOR
            GO TO 23
      21    UM=SQRT(PX2+PZ2)
            FJNOR=UN*UM*EPLEN(L)
```

```
PAT          SOURCE LISTING          ASC FAST FORTRAN COMPILER        RELEASE FT=X05
ISN           STATEMENT              CP OPTIONS = (B,D,E,K,M,V)       DATE = 06/24/82

0099              FJX(L)=-RSIFI*PX/FJNOR
0100              FJY(L)=(RCOFI*PX-2.*F0*PZ)/FJNOR
0101              FJZ(L)=-RSIFI*PZ/FJNOR
0102       25     AUG=EPLEN(L)*PI2
0103              PC=CMPLX(COS(AUG),SIN(AUG))
0104              PP(1)=PP(1)+FJX(L)*PC
0105              PP(2)=PP(2)+FJY(L)*PC
0106              PP(3)=PP(3)+FJZ(L)*PC
0107       22     CONTINUE
          C       COMPUTE PHASE DUE TO DIFFERENT FIELD POINT
0108              RCC=RCOFI*COFIC+RSIFI*SIFIC
0109              DO 30 N=1,NPT
0110              PHASE=FOR2*COTAC(N)-RCC*SITAC(N)
0111              AUG=PHASE*PI2
0112              PC=CMPLX(COS(AUG),SIN(AUG))
0113              PFI(1,N)=PFI(1,N)+PP(1)*JSP(J)*PC
0114              PFI(2,N)=PFI(2,N)+PP(2)*JSP(J)*PC
0115              PFI(3,N)=PFI(3,N)+PP(3)*JSP(J)*PC
0116       30     CONTINUE
0117       20     CONTINUE
0118              IF(I.EQ.IEND)RSIMP=R
0119              RSIMP=JSP(J)*R
0120              DO 15 L=1,NPT
0121              PSUM(1,L)=PSUM(1,L)+PFI(1,L)*RSIMP
0122              PSUM(2,L)=PSUM(2,L)+PFI(2,L)*RSIMP
0123       15     PSUM(3,L)=PSUM(3,L)+PFI(3,L)*RSIMP
0124       10     R1=R1+SINC
          C       FIND PATTERN FUNCTION
0125              PNOR=0.
0126              DO 40 I=1,NPT
0127              PC= COTAC(I)*(PSUM(1,I)*COFIC+PSUM(2,I)*SIFIC)-PSUM(3,I)*SITAC(I)
0128              PAT(1,I)=PC*CONJG(PC)
0129              IF(PAT(1,I).GT.PNOR)PNOR=PAT(1,I)
0130              PC=-PSUM(1,I)*SIFIC+PSUM(2,I)*COFIC
0131              PAT(2,I)=PC*CONJG(PC)
```

```
PAT        SOURCE LISTING          ASC FAST FORTRAN COMPILER        RELEASE FTFX05

ISN           STATEMENT            CP OPTIONS = (B,D,E,K,M,V)       DATE = 06/24/82

              IF(PAT(2,I).GT.PNOR)PNOR=PAT(2,I)
              PRINT 105,(PSUM(K,I),K=1,3)
        105   FORMAT(10X,6E12.4)
        40    CONTINUE
              PRINT 104,(PAT (1,I),I=1,NPT)
              PRINT 104,(PAT (2,I),I=1,NPT)
        104   FORMAT(//,(10X,8E12.4))
              CALL PLOTS(ERAS,500,2.)
              CALL ORIGIN(4.,0.)
              CALL SQUARE(0.,XSL,0.,YSL)
              CALL NXAXIS(0.,0.,AST   , 5.,ASF ,XSL,-.1,.1,-1)
              CALL NYAXIS(0.,0.,-60.,10.,0.,YSL,-.1,.1,-1)
              CALL CENTER(XSL/2.,-.8,.3,14HANGLE(DEGREES),0.,14)
              CALL CENTER(-.5,YSL/2.,.3,13HRADIATION(DB),90.,13)
              DO 50 K=1,2
              DO 60 I=1,NPT
              IF(PAT(K,I).LE.0.)GO TO 61
              DB=10.*ALOG10(PAT(K,I)/PNOR)
              YY(I)=(1.+DB/60.)*YSL
              GO TO 62
        61    YY(I)=-60.
        62    IF(YY(I).GT.YSL)YY(I)=YSL
        60    IF(YY(I).LT.0.)YY(I)=0.
              JJ=0
              IF(K.GT.1)JJ=10
              CALL LINE(XX,YY,NPT,1,JJ,4)
        50    CONTINUE
              CALL ENDPLT
              END
```

D

FI

8 -