Final Report

NASA Grant NAG-1-738

FOCAL REGION FIELDS OF DISTORTED REFLECTORS

N. E. Buris

J. F. Kauffman

Department of Electrical and Computer Engineering
North Carolina State University
Raleigh, North Carolina 27695-7911

July 1988
Focal Region Fields of Reflectors of Arbitrary Surface

Abstract

The problem of the focal region fields scattered by an arbitrary surface reflector under uniform plane wave illumination is solved. The Physical Optics (PO) approximation is used to calculate the current induced on the reflector. The surface of the reflector is described by a number of triangular domain-wise 5th degree bivariate polynomials. A 2-dimensional Gaussian quadrature is employed to numerically evaluate the integral expressions of the scattered fields. No Fresnel or Fraunhofer zone approximations are made. The relation of the focal fields problem to surface compensation techniques and other applications are mentioned. Several examples of distorted parabolic reflectors are presented. The computer code developed is included, together with instructions on its usage.
Introduction

Far field electromagnetic scattering from conducting surfaces of arbitrary shape has been studied by several researchers [1-6]. To our knowledge, however, little or no attention has been given to the near field scattering from arbitrary surfaces. The analysis of this problem may be useful to the efforts of compensation of reflector surface distortions. To illustrate this, let us consider the following situation. A plane wave, $\mathbf{E}^i$, is incident on an arbitrary reflector; the scattered fields, $\mathbf{E}^s$, in a region, $\Omega$, in the neighborhood of the reflector are calculated by the method outlined in the present paper. Using the reciprocity theorem yields, if in $\Omega$ the fields $\mathbf{E}^s$ are considered to be incident on the reflector, then the fields scattered by it will be $\mathbf{E}^i$. For antenna performance considerations $\mathbf{E}^i$ is required to meet certain specifications. These specifications can be easily employed in the context of the method presented here and yield the $\mathbf{E}^s$ fields in $\Omega$. If $\Omega$ is chosen to be the region occupied by the feed, then $\mathbf{E}^s$ is related to the excitation. Thus a scheme can be developed to provide the necessary excitation that renders desired far fields, $\mathbf{E}^i$, from a reflector antenna of arbitrary shape. In addition, focal field analysis may be applied to the problem of scattering by objects in the neighborhood of the feed and/or on the feed itself.

In the present paper, we develop a straightforward method to solve for the scattered fields (with emphasis in the near field zone) when a plane wave is incident on a reflector with arbitrary surface. The currents induced on the reflector are calculated by the Physical Optics (PO) approximation ($\mathbf{K} = 2\hat{n} \times \mathbf{H}^i$). Subsequently, these currents are used as sources of the scattered fields and explicit integral expressions of the latter in terms of $\mathbf{K}$ are shown. These integrals expressions involve no approximation due to the position of the observation points with respect to the sources and, therefore, are valid for all zones.
(near, Fresnel and Fraunhofer). Since we deal with rapidly oscillating integrands, accurate integration techniques are required and employed in the present work.

The novelty of our method consists of:

1. Triangular discretization of the reflector aperture by a max–min criterion based algorithm.

2. Reflector surface representation via interpolation of several fifth degree bivariate polynomials.

3. Development of a systematic, 2-dimensional, Gaussian quadrature calculation of the integrals involved in the scattered field expressions.

In this report, we examine the effects of various surface distortions on the focal region fields of parabolic reflectors. The performance of distorted reflectors with frequency is also discussed.

**Theory**

Let us consider a plane electromagnetic wave incident on an arbitrarily shaped and perfectly conducting surface, $\Sigma$, as in Figure 1. Let us also consider a Cartesian coordinate system $0xyz$ such that the magnetic field of the incident wave is along the $y$ axis and the wave vector is along the negative $z$ axis. The surface of the reflector in this coordinate system is described by the function

$$z = g(x, y). \tag{1}$$

The fields of the incident wave are given by

$$\bar{H} = \hat{y}H_0 e^{i\beta z} \tag{2}$$
and

$$\overrightarrow{E} = -\hat{z} \eta H_0 e^{j\beta z}$$  \hspace{1cm} (3)$$

where

$$\eta = \sqrt{\frac{\mu}{\varepsilon}}$$  \hspace{1cm} (4)$$

and \(\beta\) is the wavenumber. \(\hat{x}, \hat{y}\) and \(\hat{z}\) are the unit vectors along the axis of the \(Oxyz\) coordinate system and \(\eta\) is the intrinsic impedance of the medium. According to the physical optics (PO) approximation, a surface current density \(\overrightarrow{K}\) is induced on the reflector. This current density is the source of the scattered fields and is equal to

$$\overrightarrow{K}(\overrightarrow{r}) = 2\hat{n}(\overrightarrow{r}) \times \overrightarrow{H}(\overrightarrow{r}) ; \ \overrightarrow{r} \in \Sigma$$  \hspace{1cm} (5)$$

where \(\hat{n}(\overrightarrow{r})\) is the unit vector normal to the surface of the reflector. Using (1) we can obtain the expression

$$\hat{n}(\overrightarrow{r}) = \frac{\hat{z} - \nabla g(x, y)}{|\hat{z} - \nabla g(x, y)|}.$$  \hspace{1cm} (6)$$

Elaborating on Equation (6) and using Equation (2) and (5), the surface current density \(\overrightarrow{K}(\overrightarrow{r})\) is derived to be

$$\overrightarrow{K}(\overrightarrow{r}) = 2H_0 \frac{-\hat{x} - \hat{z} \frac{\partial g}{\partial z}}{\sqrt{1 + \left(\frac{\partial g}{\partial x}\right)^2 + \left(\frac{\partial g}{\partial y}\right)^2}} e^{j\beta z}.$$  \hspace{1cm} (7)$$

The vector potential due to this current is

$$\overrightarrow{A}(\overrightarrow{r}) = \frac{1}{4\pi} \int_{\Sigma} \overrightarrow{K}(\overrightarrow{r'}) \frac{e^{-j\beta |\overrightarrow{r} - \overrightarrow{r'}|}}{|\overrightarrow{r} - \overrightarrow{r'}|} d^2 r'$$  \hspace{1cm} (8)$$

The magnetic and electric scattered fields are related to \(\overrightarrow{A}(\overrightarrow{r})\) by

$$\overrightarrow{H}(\overrightarrow{r}) = \nabla \times \overrightarrow{A}(\overrightarrow{r})$$  \hspace{1cm} (9)$$
and
\[
\overline{E}(\bar{r}) = \frac{1}{j\omega \epsilon} \nabla \times \nabla \times \overline{A}(\bar{r}). \tag{10}
\]

Expanding Equation (10) we obtain the following integral expression of the electric field in terms of the current density.
\[
\overline{E}(\bar{r}) = \frac{1}{4\pi j\omega \epsilon} \int_{\Sigma} \frac{e^{-j\beta R}}{R} \left\{ \left[ \overline{K}(\bar{r}') \cdot (\bar{r} - \bar{r}') \right] \frac{3 + 3j\beta R - \beta^2 R^2}{R^4} (\bar{r} - \bar{r}') \right. \\
\left. - \frac{1 - \beta^2 R^2 + j\beta R}{R^2} \overline{K}(\bar{r}') \right\} d^2 r'. \tag{11}
\]

where \( \epsilon \) is the dielectric permittivity of the environment and \( \overline{R} = \bar{r} - \bar{r}' \). Equation (11) is in agreement with an alternative expression given by Silver [7]. The integration is carried over the illuminated portion of the reflector and \( d^2 r' \) is the area element on \( \Sigma \). If \( \sigma \) is the projection of \( \Sigma \) on the \( x-y \) plane, the integral of Equation (11) can be performed on \( \sigma \) due to the relation dictated by Equation (1).

Differential geometry considerations [2,8] yield the following relation between the area elements on \( \Sigma \) and \( \sigma \), \( d^2 r' \) and \( dx'dy' \) respectively.
\[
d^2 r' = \sqrt{1 + \left( \frac{\partial g(x',y')}{\partial x'} \right)^2 + \left( \frac{\partial g(x',y')}{\partial y'} \right)^2} \ dx'dy'. \tag{12}
\]

Combining Equation (7) and (12) we obtain
\[
\overline{E}(\bar{r})d^2 r' = 2H_0 e^{j\beta z'} \left[ -\hat{x} - \hat{z} \frac{\partial g(x',y')}{\partial x'} \right]. \tag{13}
\]

Substituting Equation (13) into Equation (11) we get
\[
\overline{E}(\bar{r}) = \frac{2H_0}{4\pi j\omega \epsilon} \int_{\sigma} \frac{e^{-j\beta(R-z')}}{R} \left\{ \left[ -(x - x') - (z - z') \frac{\partial g}{\partial x'} \right] \frac{3 + 3j\beta R - \beta^2 R^2}{R^4} (\bar{r} - \bar{r}') \right. \\
\left. + \frac{1 - \beta^2 R^2 + j\beta R}{R^2} \left( \hat{x} + \hat{z} \frac{\partial g}{\partial x'} \right) \right\} dx'dy'. \tag{14}
\]
Normalized to the magnitude of the electric field of the incident wave and expressed in component form, Equation (14) yields

\[
E_{sx}(\vec{r}) = \frac{1}{2\pi j\beta} \int_{\sigma} e^{-j\beta(R-z')} \left\{ \left[ x - x' + (z - z') \frac{\partial g}{\partial x'} \right] \frac{\beta^2 R^2 - 3 - 3j\beta R}{R^4} (x - x') + \frac{1 - \beta^2 R^2 + j\beta R}{R^2} \right\} dx'dy'
\]

(15)

\[
E_{sy}(\vec{r}) = \frac{1}{2\pi j\beta} \int_{\sigma} e^{-j\beta(R-z')} \left\{ \left[ (x - x') + (z - z') \frac{\partial g}{\partial x'} \right] \right\} \frac{\beta^2 R^2 - 3 - 3j\beta R}{R^4} (y - y') \} dx'dy'
\]

(16)

\[
E_{sz}(\vec{r}) = \frac{1}{2\pi j\beta} \int_{\sigma} e^{-j\beta(R-z')} \left\{ \left[ x - x' + (z - z') \frac{\partial g}{\partial x'} \right] \frac{\beta^2 R^2 - 3 - 3j\beta R}{R^4} \right\} \left( z - z' \right) \} dx'dy'
\]

(17)

**Calculation of Integrals**

The expressions of the scattered electric field in Equation (14) has been obtained by making only one approximation, namely the PO approximation (cf. Equation (5)). Since we are primarily interested in the focal region fields, the usual and convenient far field approximations cannot be used here. Neither can the Fresnel region approximations since we would like to treat geometries for which the focal point is in the near zone. Since we want to analyze arbitrary surfaces, a good way to represent the reflector is by a number of "target points." In an experimental version of this problem, target points would be small dots of an appropriate material placed on the surface of the reflector. We assume that the coordinates of these points can be measured by some technique (photogrammetric measurements). The projection of these points on the \(x - y\) plane provides a natural discretization of the domain of integration \(\sigma\) (see Figure 2). We employ a max min criterion algorithm \([9,10]\) to define triangles with vertices which are the projections of the target
points on \( \sigma \). Since the collection of all triangles constitutes a polygonal approximation to \( \sigma \), the integrals over \( \sigma \) can be found by simply calculating the contribution from each triangular domain and subsequently adding them up. For an analytical representation of the surface of the reflector, the same program mentioned above is used to interpolate a fifth degree bivariate polynomial in each triangular domain. This polynomial representation is used to estimate the values of \( z' \) and \( \frac{\partial g}{\partial z'} \) on the surface of the reflector. Needless to say, the larger the number of target points and, thus, triangles on \( \sigma \), the better the representation of \( \Sigma \). The integral of Equation (14) is not only vectorial but, it is complex as well. Therefore, numerically we need to perform six integrations in order to find the real and imaginary parts of all three components of the electric field at an arbitrary point \( \vec{r} = \hat{\imath}x + \hat{\jmath}y + \hat{k}z \).

The functions to be integrated can be very oscillatory in nature due to the phase term \( \exp(-j\beta(r - z')) \) and also due to the presence of the \( \frac{\partial g}{\partial x'} \) term. As is well known [11,12] there are no satisfactory quadrature formulae to integrate functions on triangular domains. However, a series of mappings can be employed to transform a triangle into a square (see Figure 3). There, a simple extension of the one-dimensional Gaussian Quadrature is available. A brief outline of this method follows.

By means of a linear transformation, every triangle of the \( x-y \) plane can be mapped onto a specific triangle in another plane, \( u-v \). Consider an arbitrary triangle in the \( x-y \) plane with vertices \((x_1, y_1), (x_2, y_2)\) and \((x_3, y_3)\). This arbitrary triangle can be mapped onto a basic triangle in another coordinate system, \( u-v \), by the transformation:

\[
x = x_1 + au + bv \quad (18a)
\]

\[
y = y_1 + cu + dv \quad (18b)
\]

where

\[
a = x_2 - x_1 \quad (19)
\]
\[ b = x_3 - x_1 \]  
\[ c = y_2 - y_1 \]  
and  
\[ d = y_3 - y_1 \]  
The Jacobian of mapping (18) is  
\[ J \left( \frac{x, y}{u, v} \right) = \begin{vmatrix} a & b \\ c & d \end{vmatrix} = (ad - bc) = (x_2 - x_1)(y_3 - y_1) - (x_3 - x_1)(y_2 - y_1). \]  
The inverse transformation is  
\[ u = \frac{[d(x - x_1) - b(y - y_1)]}{(ad - bc)} \]  
\[ v = \frac{[-c(x - x_1) + a(y - y_1)]}{(ad - bc)}. \]  
According to (18), the vertices \( A = (x_1, y_1) \), \( B = (x_2, y_2) \) and \( C = (x_3, y_3) \) are mapped onto the points \([0, 0], [1, 0] \) and \([0, 1] \), respectively, in the \( u - v \) coordinate system. These points define what we call the basic triangle in the \( u - v \) system. As mentioned above, there are no satisfactory quadrature formulae for integration on triangular domains for rapidly oscillatory integrands. The basic triangle in the \( u - v \) system can be mapped onto a square domain by means of the transformation  
\[ u = \frac{1 + r}{2} \]  
\[ v = \left( \frac{1 - r}{4} \right)(1 + s). \]  
The Jacobian of mapping (25) is  
\[ J \left( \frac{u, v}{r, s} \right) = \begin{vmatrix} 1/2 & 0 \\ -1 + s & 1 - r \end{vmatrix} = \frac{1 - r}{8}. \]
According to (25), the sides \( v = 0 \) and \( v = 1 - u \) of the basic triangle are mapped onto the lines \( s = -1 \) and \( s = +1 \) respectively, in the \( r - s \) coordinate system. Similarly, the side \( u = 0 \) is mapped onto the line \( r = -1 \) and, the point B is mapped onto the line \( r = 1 \). In Figure 2, we show the series of mappings described above.

An integral over \( D_{xy} \) can be numerically calculated using a Gaussian quadrature formula on the square domain \( D_{rs} \) provided we represent the integrand appropriately. Thus, if \( f(x, y) \) is a function defined over the triangular domain \( D_{xy} \) and \( I \) is the integral

\[
I = \int_{D_{xy}} \int f(x, y) dx \, dy,
\]

we know that

\[
I = \int_{0}^{1} \int_{u=0}^{1-u} f(x(u, v), y(u, v)) J \left( \frac{x, y}{u, v} \right) \, dudv
\]

and also

\[
I = \int_{-1}^{1} \int_{-1}^{1} f \left( \left( u(r, s), y(u(r, s), v(r, s)) \right) \right) J \left( \frac{x, y}{u, v} \right) J \left( \frac{u, v}{r, s} \right) \, dr \, ds.
\]

For our purposes, Equation (29) essentially looks like

\[
I = \int_{-1}^{1} \int_{-1}^{1} T(r, s) \, dr \, ds.
\]

where

\[
T(r, s) = f(x, y) J \left( \frac{x, y}{u, v} \right) J \left( \frac{u, v}{r, s} \right)
\]

An obvious 2–dimensional extension of the well–established 1–dimensional quadrature formulae is used here. That is

\[
I \approx \sum_{j=1}^{N} W_j T(r_j, s_j) dr,
\]
and, finally

$$I \simeq \sum_{k=1}^{N} \sum_{j=1}^{N} W_k W_j T(r_k, s_j). \quad (32)$$

In our analysis, \( r_k \) and \( s_j \) \((k, j = 1, 2, \ldots, N)\) are the roots of the \( N^{th} \) order Legendre polynomial, \( P_N \), and \( W_k(k = 1, 2, \ldots, N) \) are the appropriate weights for Gaussian quadrature, given by

$$W_k = \frac{2}{(1 - r_k^2) P'_N(r_k^2)}. \quad (33)$$

From (32) it is clear that the values of \( T \) are required at the points \((r_k, s_j)\) of the \( r - s \) coordinate system. Using the transformations (25) and (18), the corresponding points \((x_{kj}, y_{kj})\) of the \( x - y \) coordinate system can be found. It is clear that

$$f(x_{kj}, y_{kj}) = F(r_k, s_j). \quad (34)$$

There are \( N^2 \) points in each triangle. The value of \( N \) required to accurately calculate the integrals depends on the frequency and the surface of the reflector. It is found that if the reflector has extensive distortions, a large \( N \) is required.

One subtle point needs to be mentioned here. Since the square domain \( D_{rs} \) and the basic triangle domain \( D_{uv} \) are fixed and kept the same for all \( x - y \) triangles, the limits of integration in (28) and (29) are desired to stay fixed. As a consequence, \( A, B, \) and \( C \) must be in a counter-clockwise sense. It is important that the vertices of all the \( D_{xy} \) triangles be arranged in the same rotational sense. If instead of the counter-clockwise sense presented here, a clockwise sense were chosen, then the Jacobian \( J \left( \frac{\xi, \eta}{u, v} \right) \) would need a sign reversal to ensure the correct result.

To recapitulate, the projection, \( \sigma \), of the reflector on the \( x - y \) plane is divided in a number of triangles. The integrals of Equations (15)–(17) are decomposed into summations of integrals over the aforementioned triangles. For each triangle, the points \((x_{kj}, y_{kj})\) are
found by applying (25) and (18). Then, the quadrature formula (32) is used to evaluate the contribution of each triangle to the total integral. Finally, all these contributions are added together to yield approximate values for $E_{sx}, E_{sy}$ and $E_{sz}$.

**Oblique Incidence**

The generic geometry presented in the second section can be used to solve the problem of the scattered fields of a given reflector under oblique incidence. To demonstrate this assertion, consider a coordinate system $0\tilde{x}\tilde{y}\tilde{z}$, attached to the reflector. The relative orientation of the reflector system, $0\tilde{x}\tilde{y}\tilde{z}$, and the incident wave system, $0xyz$, is described by the Euler angles $\theta, \phi$ and $\psi$ as shown in Figure 4.

To solve for the scattered fields under these circumstances, we make use of the fact that $g(x, y)$ in Equation (1) is arbitrary. If the reflector in its own coordinate system is described by $\tilde{z} = g(\tilde{x}, \tilde{y})$, then we can express this reflector in $0xyz$ and then use the analysis of Section 2. To do this we perform three rotations. First a positive rotation about $0\tilde{z}$ by $\phi$. This rotation moves $0\tilde{y}$ to a new position $0\tilde{y}'$. The second rotation is about $0\tilde{y}'$ by $+\theta$. This rotation aligns the optical axis of the reflector with $0z$. Finally, a third rotation about $0z$ by an amount $\psi$ is required to align $0\tilde{y}'$ with $0y$ and thus take care of the polarization.

The sequence of rotations is shown in Figure 5. To demonstrate the use of these rotations, consider the following special cases.

1. Incident wave $\vec{E}^i$ on the $0\tilde{x}\tilde{y}$ plane and $\vec{H}^i$ field perpendicular to the $0\tilde{x}\tilde{z}$ plane (see Figure 6). Thus, $\phi = 0$ and $\psi = 0$. 

11
2. Propagation along the negative $0\tilde{z}$ axis (see Figure 7). Thus, $\theta = 0$, $\phi = 0$, and $\psi = \psi$. Please note that you need to rotate the $0\tilde{x}\tilde{y}\tilde{z}$ system to make it align with the $0xyz$ system. Furthermore, the values of $\theta, \phi,$ and $\psi$ have to be positive for counterclockwise rotations and negative for clockwise ones.

Numerical Examples

In an experimental application of the analysis presented above, the coordinates of the target points would be measured and provided as data to the developed computer code. Since this report does not involve experimental studies, an analytical method is needed to generate the coordinates of the target points. First we distribute the projections of the target points onto the $0\tilde{x}\tilde{y}$ plane. We do this by locating all of these points on circles of successively increasing radius up to a given value. The number of points on each circle also increases as we move outwards. This scheme of discretization is suitable for circular aperture reflectors and is shown in Figure 8. In this discretization example, we have used four circles with radius and points as follows.

<table>
<thead>
<tr>
<th>Circle #</th>
<th>Radius</th>
<th>Points on Circle</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>np = 3</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>2np = 6</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>2(2np) = 12</td>
</tr>
</tbody>
</table>

This scheme provides us with the $\tilde{z}$ and $\tilde{y}$ coordinates of the target points. To complete the description of the surface of the reflector, a function is needed to generate the $\tilde{z}$ coordinates of the targets. For all the examples presented in this section, the target points lay on a
surface generated by the function

\[
\tilde{z} = \frac{x^2 + y^2}{4F} + A_1 e^{A_2[(x-A_3)^2+(y-A_4)^2]} \quad \tilde{x}^2 + \tilde{y}^2 \leq \frac{D^2}{4}.
\]

Clearly, Equation (35) represents an ideal paraboloid of focal length, \(F\), that is perturbed by a Gaussian "bump" of height, \(A_1\), steepness, \(A_2\), located at the point \((\tilde{x}, \tilde{y}) = (A_3, A_4)\).

For \(A_1 = 0\) Equation (35) represents an ideal parabolic reflector of diameter, \(D\). We are primarily interested in the fields of the focal region. The observation points in this section will be assumed to be distributed on a line segment of electrical length \(\beta\ell\) that has its origin at the point \(O_b\) on the \(z\)-axis and is parallel the the \(0\tilde{z}\) plane (see Figure 9).

In Figure 10, the magnitude of the electric field components is shown for an ideal paraboloid with diameter, \(D = 2\text{m}\), focal length, \(F = 1.19\text{m}\), at frequency, \(f = 500\text{ MHz}\). The observation points are defined by \(\phi_{ob} = 0\) and \(\beta\ell = 20\), i.e., they lay on the \(0_b\tilde{z}\) axis. In this example \(\phi = \theta = \psi = 0^\circ\), i.e., the coordinate systems \(0xyz\) and \(0\tilde{x}\tilde{y}\tilde{z}\) are identical. That is the incident field is propagating along the negative \(0\tilde{z}\) axis and has its \(\overline{H}(\overline{E})\) field along the \(0\tilde{y}(0\tilde{x})\) axis. It is clear from Figure 10 that the cross polar field, \(E_y\), is very weak, negligible in comparison to \(E_x\) and \(E_z\). In Figure 11, the magnitude of the electric field components of a perturbed version of the paraboloid of the previous example is shown.

The characteristics of the perturbation are \(A_1 = 0.1\text{m}, A_2 = -3\), and \(A_3 = A_4 = 0.2\text{m}\). A comparison between Figure 10 and 11 reveals that the surface perturbation has quantitative effects on the focal region fields. First, the co-polar component has a maximum decreased by 10% from the corresponding value of the ideal surface case. Second, the cross-polar level is somewhat increased and third, the \(E_x\) component is no longer zero at the point \(O_b\).

In Figure 12, the magnitude of the normalized electric field components of the same distorted reflector used in Figure 11 is shown. The only difference between Figure 11 and 12 is the frequency. In Figure 11, the frequency is 5 GHz. The observation points are
again distributed on the $0_5 \tilde{x}$ axis and $\beta \ell$ is the same as earlier ($\beta \ell = 20 \rightarrow \ell \approx 20 \text{ cm}$).

As is expected, because of the higher frequency, the bump on the surface of the reflector has more severe effects now than in Figure 11. To illustrate this, we compare the focal fields of the distorted reflector with the fields of the ideal reflector at $f = 5 \text{ GHz}$. This comparison is depicted in Figures 13a-13f. From Figure 12, we see that the cross-polarized field component ($E_y$) is eight times smaller than the coplanar component or, down by 18 dB. We also see that, as in Figure 11, the axial field at the point $0_5$ is not zero when the surface has been distorted. Furthermore, we note that the phase of the cross-polar field component of the ideal surface parabolic reflector is not significant since the magnitude of the cross-polar field is zero (Figure 13b). Because of this fact, the phase of the cross-polar component of the ideal reflector is shown to exhibit a rather large numerical inaccuracy (Figure 13e). We may also point out that there is a significant increase (from 3.5 to 37) of the normalized coplanar component when we go from $f = 500 \text{ MHz}$ to $f = 5 \text{ GHz}$ (see Figures 10 and 13a). This is in accord to the well known limit of infinite focal fields at the Geometric Optics (GO) approximation.

Exactly the same number of target points has been used for all of the examples presented above. Furthermore, the coordinates of these points in the $0\tilde{x}\tilde{y}$ plane were also exactly the same. This enabled us to compare the results of the various test cases used. It is important to realize that, in general, the domain-wise 5th degree polynomial interpolation only approximates the surface of the reflector. As a consequence, if one uses two different sets of $x$-$y$ coordinates of target points to represent a given reflector surface (for example given by Equation (35)), different scattered fields will be obtained under the same conditions of illumination. This problem, however, can be solved very easily by increasing the number of target points to achieve good polynomial representation of the
reflector surface. Numerical verification of this solution has been done, and we have found that the higher the frequency, the larger the number of target points required to represent the reflector in order to achieve convergence of the scattered fields.

At $f = 35$ GHz, and when the propagation is along the optical axis, the focal region fields of ideal parabolic receiving antennas obtained by the method of the present report have been found in complete agreement with the numerical results obtained by a different method (expansion in spherical wave functions) as in [13]. For arbitrary illumination and/or reflectors, however, we have not found studies by other researchers to cross-examine our numerical results.

In these cases, the scattered fields at a few observation points are calculated using various densities of points in the Gaussian quadrature (this is equivalent to varying $N$ in Equation (32)) until convergence is achieved. For a given frequency, we have found that the smoother the surface of the reflector (the smaller $A_1$ in Equation (35)), the smaller $N$ is required for convergence of the fields. To demonstrate this process, the ideal parabolic reflector used earlier ($D = 2m$, $F = 1.19m$) is subject to obliquely incident illumination ($\theta = 5^\circ$, $\phi = 0$, $\psi = 0$). In contrast to the previous examples, now the antenna $(0\tilde{x}\tilde{y}\tilde{z})$ and the ray $(0\tilde{x}\tilde{y}\tilde{z})$ coordinate systems are not identical. The normalized incident field, $E_i/E_0$ in the two systems is

$$
\frac{E_i}{E_0} = -\hat{x}e^{i\beta z} = -\left(\hat{\tilde{x}} \cos 5^\circ - \hat{\tilde{z}} \sin 5^\circ\right) e^{i\beta(\tilde{z} \sin 5^\circ + \tilde{z} \cos 5^\circ)).
$$

The observation points are distributed on the negative $\tilde{z}$-axis, i.e., $\phi_{\text{on}} = 180^\circ$ and $\beta \ell = 20 \to \ell \approx 20$ cm. The geometry is shown in Figure 14. The magnitude of the normalized electric field components in the antenna coordinate system is plotted in Figures 15, 16, and 17 for various values of $N = ng$. From these figures, it is evident that convergence is
achieved very quickly. Note that the $\tilde{y}$-component is the smallest of all; about four orders of magnitude smaller than the $\tilde{x}$ and $\tilde{z}$ components. Convergence of the $\tilde{y}$-component is, however, still very rapid. In Figure 15, we observe that the scattered field has a maximum at a distance $\tilde{x}_0 = -11.2$ cm. If we interpret this in degrees from the optical axis, we find that the maximum of the scattered field occurs at an angle, $\theta_0$, where

$$\theta_0 = \tan \left| \frac{\tilde{x}_0}{F} \right| \approx 5.38^\circ.$$  \hspace{1cm} (37)

This was expected, since the angle of incidence, $\theta$, is $\theta = 5^\circ$.

As an additional example, consider the previous case when the reflector is distorted by a bump located at the origin. The surface of the reflector is described by Equation (34) with parameters $D = 2$ m, $F = 1.19$ m, $A_1 = 0.1$ m, $A_2 = -3$, $A_3 = 0$, and $A_4 = 0$. The frequency is again $f = 5$ GHz, and the incident illumination is described by Equation (36); i.e., $\theta = 5^\circ$, $\phi = 0$, $\psi = 0$. The magnitude of the normalized components of the scattered field on the negative $\tilde{x}$-axis ($\phi_{0b} = 180^\circ$, $\beta \ell = 20$) is shown in Figures 18, 19 and 20 for various $N = n_q$. The convergence is achieved rapidly, especially for the major components of the field ($E_{\tilde{x}}$ and $E_{\tilde{z}}$). Comparing the undistorted (Figure 15, 16, and 17) with the distorted (Figure 18, 19, and 20) we observe that the largest field component, $E_{\tilde{x}}$, suffers a severe reduction (by a factor of 5) from its level at the absence of distortion. The $E_{\tilde{z}}$ component suffers a comparable reduction. Moreover, the level of the smallest field component, $E_{\tilde{y}}$, is increased (by an order of magnitude) due to the distortion.

The parameter $A_1 = 0.1$ m used in the examples of distorted reflector examples is a rather extensive distortion, since it represents 10% of the radius of the reflector. In practical applications less severe distortions are expected. The higher the frequency, $f$, the larger the angle of incidence, $\theta$, the larger $N$ is required to achieve convergence.
Conclusions

A straightforward method to calculate the magnitude and phase of all three components of the focal region field scattered by an arbitrary reflector under plane wave illumination is presented. Discretization of the surface of the reflector and subsequent interpolation by fifth degree bivariate polynomials is made. The current induced on the reflector is evaluated by the PO approximation. The scattered field is given in an integral expression that involves no Fresnel or Fraunhofer zone approximations. A novel approach to the problem of numerical integration of the integrals is discussed. This method is based on a 2-dimensional Gaussian quadrature and provides a high degree of flexibility since the density of the sample points can be varied interactively as to adapt to the accuracy needs of the specific application. Several examples of distorted and undistorted parabolic reflectors are presented. The behavior of the scattered field with frequency and angle of incidence is discussed.
References


Figure 1. Arbitrary Reflector Geometry
Figure 2. Reflector Discretization
Figure 3. Successive mapping of an arbitrary triangle onto a basic one and, then, a square.
Figure 4. Relative orientation of antenna and ray coordinate systems.
Figure 5. Euler's angles $\theta$, $\phi$ and $\psi$ define the relative orientation between the antenna and the ray coordinate system.
Figure 6. Geometry for example I (section on oblique incidence)
Figure 7. Geometry for example II (section on oblique incidence)
Figure 8. Discretization Scheme
Figure 9. \(0_{\theta_b} \), \(\phi_{ob} \) and \(\ell \) are the parameters used to define the distribution of the observation points.
Figure 10. Normalized $E$. Observation points on the $0z$ axis. $f = 500$ MHz, $D = 2m$, $F = 1.19m$, $\beta\ell = 20$, $00_b = F$, $\theta = \phi = \psi = 0$. $A_1 = A_2 = A_3 = A_4 = 0$. 
Figure 12. Normalized $E$. The distorted reflector of Figure 11 at $f = 5$ GHz.
Figure 13a. Normalized $|E_z|$ of ideal and distorted reflectors.
Figure 13b. Normalized $|E_y|$ of ideal and distorted reflectors.
Figure 13c. Normalized $|E_z|$ of ideal and distorted reflectors.
Figure 13d. Phase of coplanar component of ideal and distorted reflectors.
Figure 13e. Phase of cross-polar component of ideal and distorted reflectors.
Figure 13f. Phase of axial component of ideal and distorted reflectors.
Figure 14. Oblique Incidence at $f = 5$ GHz. $\theta = 5^\circ$, $\phi = 0$, $\psi = 0$. $D = 2m$, $\beta \ell = 20$, $00_b = F = 1.19m$. 
Figure 15. Normalized $|E\tilde{z}|$. Oblique incidence ($\theta = 5^\circ, \phi = 0, \psi = 0$).
$D = 2\text{m}, F = 1.19\text{m}, A_1 = A_2 = A_3 = A_4 = 0.$
Figure 16. Normalized $|E\hat{y}|$. Oblique incidence ($\theta = 5^\circ, \phi = 0, \psi = 0$).

$D = 2m, F = 1.19m, A_1 = A_2 = A_3 = A_4 = 0$. 

IDEAL 

f=5 GHz, Theta=5° IDEAL 

--- ng=4 
\cdot ng=6 
\cdot\cdot ng=8 
- - ng=10
Figure 17. Normalized $|E_z|$. Oblique incidence ($\theta = 5^\circ$, $\phi = 0$, $\psi = 0$).

$D = 2m$, $F = 1.19m$, $A_1 = A_2 = A_3 = A_4 = 0$. 

f=5 GHz, Theta=5° IDEAL
Figure 18. Normalized $|E_x|$. Oblique incidence ($\theta = 5^\circ$, $\phi = \psi = 0$).

$D = 2m$, $F = 1.19m$, $A_1 = 0.1m$, $A_2 = -3$, $A_3 = A_4 = 0$.,
Figure 19. Normalized $|E_\gamma|$. Oblique incidence ($\theta = 5^\circ$, $\phi = \psi = 0$).
$D = 2m$, $F = 1.19m$, $d_1 = 0.1m$, $A_2 = -3$, $A_3 = A_4 = 0$. 
Figure 20. Normalized $|E_Z|$. Oblique incidence ($\theta = 5^\circ$, $\phi = \psi = 0$).

$D = 2m$, $F = 1.19m$, $A_1 = 0.1m$, $A_2 = -3$, $A_3 = A_4 = 0$. 
APPENDIX A

INSTRUCTIONS AND EXAMPLE OF CODE USAGE

Nick E. Buris
University of Massachusetts
Dept. of Electrical and Computer Eng.
Marcus Hall Rm 20
Amherst, MA 01003
tel # (413) 545-2170

ATTENTION: ALL LENGTHS IN METERS, FREQUENCY IN MHZ.
SUBROUTINE JOORAN USES SUBROUTINE IBIRAN, THEREFORE, WHEN COMPILING MAKE SURE THAT YOU LINK TO A COMPILED VERSION OF IBIRAN.

ARGUMENT DESCRIPTION

INPUTS

nd number of target points specifying the reflector
( nd must be greater than 3 )
xdtil real array of dimension nd; (x coordinates of target points)
ydtil real array of dimension nd; (y coordinates of target points)
zdtil real array of dimension nd; (z coordinates of target points)
nob number of observation points
xotil real array of dimension nob; (x coordinates of
observation points)

**yotil**
real array of dimension nob; (y coordinates of observation points)

**zotil**
real array of dimension nob; (z coordinates of observation points)

**ald**
Eulerian angle theta in degrees

**a2d**
Eulerian angle phi in degrees

**a3d**
Eulerian angle psi in degrees

**freq**
real variable; frequency in MHz

**ngauss**
integer variable that determines the density of points used in the Gaussian Quadrature method of numerical integration. The larger ngauss the better the approximation. Available ngauss=4,6,8,10,12,16,48. In the report, N and ng are used for ngauss.

**OUTPUT**

**Estil**
Complex array of dimension (3,nob). It contains the values of the scattered Electric field.

\[ Estil(j,k) = j\text{-th component of } E \text{ at } k\text{-th observ. point } j = 1 \Rightarrow Ex, \quad j = 2 \Rightarrow Ey, \quad j = 3 \Rightarrow Ez \]

**WORK ARRAYS (OUTPUTS)**

**wk**
real array of dimension 14*nd

**iwk**
integer array of dimension 31*nd+1

**xd**
real array of dimension nd

**yd**
real array of dimension nd

**zd**
real array of dimension nd

**xob**
real array of dimension nob

**yob**
real array of dimension nob

**zob**
real array of dimension nob

**x**
real array of dimension (ngauss, ngauss)

**y**
real array of dimension (ngauss, ngauss)

**z**
real array of dimension (ngauss, ngauss)

**zx**
real array of dimension (ngauss, ngauss)

**Ro**
real array of dimension (ngauss, ngauss)

**SCA1**
complex array of dimension (ngauss, ngauss)

**SCA2**
complex array of dimension (ngauss, ngauss)

**SCA3**
complex array of dimension (ngauss, ngauss)

**gw**
double precision real array of dimension ngauss

**gz**
double precision real array of dimension ngauss

**u**
real array of dimension ngauss

**v**
real array of dimension ngauss

**TO USE JOORAN SIMPLY CALL IT WITH A CALL STATEMENT LIKE**

```
call Jooran(nd,xdtil,ydtil,zdtil,nob,xotil,yotil,zotil,  
    1  ald,a2d,a3d,freq,ngauss,Estil,  
    2  wk,iwk,xd,yd,zd,xob,yob,zob,x,y,z,zx,Ro,SCA1,SCA2,SCA3,  
    3  gw,gz,u,v)```

The first two lines of arguments contain the input arguments and the output argument Estil. The last two lines of arguments contain work arrays. The general user need not worry about these arrays. However, if changes in the program are needed for extensions and/or modifications, then the user should consult the theoretical part of this report and keep in mind that:

1) \( x_d, y_d \) and \( z_d \) are the coordinates of the target points in the ray system \( \text{(} OXYZ \text{)} \).
2) \( x_{ob}, y_{ob} \) and \( z_{ob} \) are the coordinates of the observation points in the ray system \( \text{(} OXYZ \text{)} \).
3) \( x, y, z, z_x \) contain the values of \( x, y, z \) and \( z_x \) at the points defined by the grid \( \text{ngauss X ngauss} \) on each individual triangle. These values are used in the numerical integration scheme.
4) \( R_o \) contains the values of the distance of the current observation point from each of the point of the grid mentioned in 3.
5) SCA1, SCA2 and SCA3 contain some common factors that appear in the integral expressions for all the three components of the Electric field. Again, these values are calculated at the points of the grid mentioned in 3.
6) \( g_w, g_z \) are the weights and the zeros of Legendre polynomials used in the Gaussian Quadrature integration.
7) \( u, v \) are the coordinates of the points of the grid mentioned in 3 expressed in the \( u-v \) coordinate system. In this coordinate system all the individual triangles are mapped onto a basic triangle.
As an example of using Jooran consider the following data file

**NASA87.DAT**

```
5000.000 frequency in MHz
4 one of (4,6,8,10,12,16,48)
5.000000 0.0000000E+00 0.0000000E+00 Euler angles
22 # of target points

<table>
<thead>
<tr>
<th>Point</th>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.000000000000000E+00</td>
<td>0.000000000000000E+00</td>
<td>0.000000000000000E+00</td>
<td></td>
</tr>
<tr>
<td>0.333333333333333</td>
<td>0.000000000000000E+00</td>
<td>2.136752151277409E-02</td>
<td></td>
</tr>
<tr>
<td>-0.166666666666667</td>
<td>0.2886751345948129</td>
<td>2.136752151277409E-02</td>
<td></td>
</tr>
<tr>
<td>-0.166666666666667</td>
<td>2.136752151277409E-02</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0.666666666666667</td>
<td>0.000000000000000E+00</td>
<td>8.5470088605109637E-02</td>
<td></td>
</tr>
<tr>
<td>0.333333333333333</td>
<td>0.5773502691896258</td>
<td>8.5470088605109636E-02</td>
<td></td>
</tr>
<tr>
<td>-0.333333333333333</td>
<td>0.5773502691896258</td>
<td>8.5470088605109639E-02</td>
<td></td>
</tr>
<tr>
<td>-0.666666666666667</td>
<td>-2.9379182519358776E-17</td>
<td>8.5470088605109637E-02</td>
<td></td>
</tr>
<tr>
<td>-0.333333333333333</td>
<td>-0.5773502691896258</td>
<td>8.5470088605109637E-02</td>
<td></td>
</tr>
<tr>
<td>0.333333333333333</td>
<td>-0.5773502691896258</td>
<td>8.5470088605109639E-02</td>
<td></td>
</tr>
<tr>
<td>1.000000000000000</td>
<td>0.000000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>0.8660254037844387</td>
<td>0.500000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>0.500000000000000E+00</td>
<td>0.8660254037844387</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-2.2034386889519082E-17</td>
<td>1.000000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-0.500000000000000E+00</td>
<td>0.8660254037844387</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-0.8660254037844387</td>
<td>0.500000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-1.000000000000000E+00</td>
<td>-4.4068773779038163E-17</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-0.8660254037844387</td>
<td>-0.500000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-0.500000000000000E+00</td>
<td>-0.8660254037844387</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>-7.2674717409587322E-17</td>
<td>-1.000000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>0.500000000000000E+00</td>
<td>-0.8660254037844387</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>0.8660254037844386</td>
<td>-0.500000000000000E+00</td>
<td>0.1923076993614967</td>
<td></td>
</tr>
<tr>
<td>2 (nob # of observation points)</td>
<td>0.000000000000000E+00</td>
<td>1.299999952316284</td>
<td></td>
</tr>
<tr>
<td>4.7746483236551285E-02</td>
<td>0.000000000000000E+00</td>
<td>1.299999952316284</td>
<td></td>
</tr>
</tbody>
</table>
```

As an example of using Jooran consider the following data file.
This data file, nasa87.dat, is input to the following program:

```fortran
real xdtil(200),ydtil(200),zdtil(200)
real xotil(50),yotil(50),zotil(50)
real xd(200),yd(200),zd(200),wk(14*200)
real Xob(50),Yob(50),Zob(50)
complex Estil(3,50)
integer nd,iwk(31*200+1*1)

real x(48,48),y(48,48),z(48,48)
real xz(48,48)
real Ro(48,48)

complex SCA1(48,48),SCA2(48,48)
complex SCA3(48,48)
dimension gw(48),gz(48),u(48),v(48,48)
double precision gw,gz

open(1, file='nasa87.dat')
read(1,*)freq
read(1,*)ngauss
read(1,*)ald,a2d,a3d
read(1,*)nd
do 1 k=1,nd
read(1,*)xdtil(k),ydtil(k),zdtil(k)
1 continue
read(1,*)nob
do 2 k=1,nob
read(1,*)xotil(k),yotil(k),zotil(k)
2 continue

call Jooran(nd,xdtil,ydtil,zdtil,nob,xotil,yotil,zotil,1
   ald,a2d,a3d,freq,ngauss,Estil,
   wk,iwk,xd,yd,zd,xob,yob,zob,x,y,z,zx,Ro,SCA1,SCA2,SCA3,
   gw,gz,u,v)

open(2, file='nasout.dat')
write(2,*)' ',nob,' # of observation points'
do 3 k=1,nob
write(2,*)' ',xotil(k),',',yotil(k),$'','zotil(k),',Xotil,Yotil,Zotil'
write(2,*)' ',Estil(1,k),'Extil'
write(2,*)' ',Estil(2,k),'Eytil'
write(2,*)' ',Estil(3,k),'Eztil'
3 continue
stop
end
```
The output is sent to the file "nasout.dat"

NASOUT.DAT

2  # of observation points

0.0000000E+00, 0.0000000E+00, 1.300000
(-1.501289,-0.8454167) Extil
(-7.9167667E-06,1.5168914E-04) Eytil
(0.2458099,-0.5834020) Eztil
4.7746483E-02, 0.0000000E+00, 1.300000
(0.6274602,0.1810894) Extil
(-1.3166843E-03,-4.8433035E-04) Eytil
(-0.1880506,0.5139895) Eztil
Figure 11. Normalized $E$. Observation points on the $0x$ axis.

$f = 500$ MHz, $D = \varnothing m$, $F = 1.19m$, $\beta \ell = 20$,

$00_b = F$, $\theta = \phi = \psi = 0$. $A_1 = 0.1m$, $A_2 = -3$, $A_3 = 0.2m$, $A_4 = 0.2m$. 
APPENDIX B

SUBROUTINE JOORAN FORTRAN CODE
The following subroutine calculates the three components of the electric field scattered from an arbitrary perfect conductor. The incident wave is assumed to be a plane wave with orientation defined by three Eulerian angles with respect to the optical axis of the reflector. It is important to note that the fields are normalized to the magnitude of the electric field of the incident wave. The plane perpendicular to the direction of propagation and passes through the origin is the zero phase plane. The method used is based on a physical optics analysis and edge effects are included. For a detailed description of the method, please refer to the report accompanying this source code. This code was developed at North Carolina State University by Dr. Nick E. Buris and Dr. J. Frank Kauffman. Support for this study was kindly offered by a NASA-Langley Research Center grant.

This code is written so that it can be modified with the minimum possible effort.

**Argument Description**

**Inputs**

- **nd**: number of target points defining the reflector
- **xdtil**: real array of dimension nd; (x coordinates of target points)
- **ydti**: real array of dimension nd; (y coordinates of target points)
- **zdti**: real array of dimension nd; (z coordinates of target points)
- **nob**: number of observation points
- **xotil**: real array of dimension nob; (x coordinates of observation points)
- **yotil**: real array of dimension nob; (y coordinates of observation points)
- **zotil**: real array of dimension nob; (z coordinates of observation points)
- **a1d**: Eulerian angle theta in degrees
- **a2d**: Eulerian angle phi in degrees
- **a3d**: Eulerian angle alpha in degrees
- **freq**: real variable; frequency in MHz
- **ngauss**: integer variable that determines the density of points used in the Gaussian Quadrature method of numerical integration. Available ngauss = 4, 6, 8, 10, 12, 16, 48.

**Outputs**

- **Estil**: Complex array of dimension (3, nob). It contains the values of the scattered electric field. Estil(j,k) = j-th component of E at k-th observ. point; j = 1 => Ex, j = 2 => Ey, j = 3 => Ez
- **wk**: real array of dimension 14*nd
C iwk  integer array of dimension 31*nd+1
C xd  real array of dimension nd
C yd  real array of dimension nd
C zd  real array of dimension nd
C xob  real array of dimension nob
C yob  real array of dimension nob
C zob  real array of dimension nob
C x  real array of dimension (ngauss,ngauss)
C y  real array of dimension (ngauss,ngauss)
C z  real array of dimension (ngauss,ngauss)
C zx  real array of dimension (ngauss,ngauss)
C Ro  real array of dimension (ngauss,ngauss)
C SCA1  complex array of dimension (ngauss,ngauss)
C SCA2  complex array of dimension (ngauss,ngauss)
C SCA3  complex array of dimension (ngauss,ngauss)
C gw  double precision real array of dimension ngauss
C gz  double precision real array of dimension ngauss
C u  real array of dimension ngauss
C v  real array of dimension ngauss
C
C******************************************************************************
C subroutine Jooran(nd,xdtil,ydtil,zdtil,nob,xotil,yotil,zotil,
1        ald,a2d,a3d,freq,ngauss,Estil,
2        wk,iwk,xd,yd,zd,xob,yob,zob,x,y,z,zx,Ro,SCA1,SCA2,SCA3,
3        gw,gz,u,v)
real xdtil(nd),ydtil(nd),zdtil(nd)
real xotil(nob),yotil(nob),zotil(nob)
real xd(nd),yd(nd),zd(nd),xiryi,zi,wk(14*nd)
real Xob(nob),Yob(nob),Zob(nob)
complex Estil(3,nob)
integer nd,iop(2),iwk(31*nd+1*1),ierr,nxi,nyi
real x(ngauss,ngauss),y(ngauss,ngauss),z(ngauss,ngauss)
real zx(ngauss,ngauss)
real Ro(ngauss,ngauss)
complex SCA1(ngauss,ngauss),SCA2(ngauss,ngauss)
complex SCA3(ngauss,ngauss)
dimension gw(ngauss),gz(ngauss),u(ngauss),v(ngauss)
double precision gw,gz
common /observ/ Xo,Yo,Zo,beta
COMMON /IBCDPT/ X0,Y0,AP,BP,CP,DP,P00,P10,P20,P30,P40,P50,
1 P01,P11,P21,P31,P41,P51,P02,P12,P22,P32,P03,P13,
2 P23,P04,P14,P05,ITPV
common /vertex/ x1,x2,x3,y1,y2,y3
complex JKRZ
complex Ex,Ey,Ez,Extot,Eytot,Eztot,Exknt,Eyknt,Ezknt
external Ex,Ey,Ez

pi=acos(-1.0)
C Convert the angles from degrees into radians
alr=ald*pi/180.
a2r=a2d*pi/180.
\[ a_3r = a_3d \pi / 180. \]

C Now calculate the elements of the Rotation Transformation Matrix, \( R_{ij} \)

\[
\begin{align*}
cl &= \cos(a_1r) \\
c2 &= \cos(a_2r) \\
c3 &= \cos(a_3r) \\
s1 &= \sin(a_1r) \\
s2 &= \sin(a_2r) \\
s3 &= \sin(a_3r) \\
R11 &= c3 \cdot c2 \cdot cl - s3 \cdot s2 \\
R12 &= c3 \cdot s2 \cdot cl + s3 \cdot c2 \\
R13 &= -c3 \cdot s1 \\
R21 &= -s3 \cdot c2 \cdot cl - c3 \cdot s2 \\
R22 &= -s3 \cdot s2 \cdot cl + c3 \cdot c2 \\
R23 &= s3 \cdot s1 \\
R31 &= s1 \cdot c3 \\
R32 &= s1 \cdot s2 \\
R33 &= c1
\end{align*}
\]

C Now Enter the Wavenumber

\[
\beta = (2 \cdot \pi \cdot \text{freq}) / 300.
\]

C Now Enter the data points!

\[
\text{call rotate}(x_{dti1}, y_{dti1}, z_{dti1}, x_d, y_d, z_d, \text{nd}, a_1r, a_2r, a_3r)
\]

C Create the \((r, s)\) pairs. These are the points where the value of the integrand is needed in the \(r-s\) system. In this system the triangular domain is the square \((-1,1) \times (-1,1)\).

\[
\begin{align*}
\text{if}(\text{ngauss} \cdot \text{eq.} 4) & \text{ call gauss4}(\text{ngauss}, \text{gz}, \text{gw}) \\
\text{if}(\text{ngauss} \cdot \text{eq.} 6) & \text{ call gauss6}(\text{ngauss}, \text{gz}, \text{gw}) \\
\text{if}(\text{ngauss} \cdot \text{eq.} 8) & \text{ call gauss8}(\text{ngauss}, \text{gz}, \text{gw})
\end{align*}
\]
if(ngauss.eq.10) call gaus10(ngauss,gz,gw)
if(ngauss.eq.12) call gaus12(ngauss,gz,gw)
if(ngauss.eq.16) call gaus16(ngauss,gz,gw)
if(ngauss.eq.48) call gaus48(ngauss,gz,gw)

C.....................................................................

C Create the \([u,v]\) pairs. These are the points where the
value of the integrand is needed in the \(u-v\) system. In this
system the domain is a right triangular defined by the points:
\((x_1,y_1) \leftrightarrow [0,0] \quad (x_2,y_2) \leftrightarrow [1,0] \quad (x_3,y_3) \leftrightarrow [0,1]\)

call rstouv(ngauss,gz,u,v)

C.....................................................................

C Now run ibiran for the first time
C Create the point mess
iop(1)=0
iop(2)=0
maxnxi=1
nxi=1
nyi=1

xi=(xd(1)+xd(2)+xd(3)+xd(4))/4.
yi=(yd(1)+yd(2)+yd(3)+yd(4))/4.

call ibiran(nd,xd,yd,zd,iop,maxnxi,nxi,xi,nyi,yi,zi,iwk,wk,ierr)

nt=iwk(5)
nl=iwk(6)

write(2,*,'# of triangles =',nt,' # of triangles =',nt,

C.....................................................................

C This is the loop for each observation point

do 6 lobe=1,nob
Xo=Xob(lobe)
Yo=Yob(lobe)
Zo=Zob(lobe)

Extot=(0.0,0.0)
Eytot=(0.0,0.0)
Eztot=(0.0,0.0)

C.....................................................................

C This is the LOOP that deals with each triangle


The barycentric point is found of each triangle and the ibiran is called in order to calculate the coefficients of the polynomial in the u-v system at the triangle in question.

do 5 knt=1,nt
iop(1)=0
iop(2)=2
nxi=1
nyi=1
maxnxi=1
kk=15+3*knt

\[ x_1 = x_d(iwk(kk-2)) \]
\[ y_1 = y_d(iwk(kk-2)) \]
\[ x_2 = x_d(iwk(kk-1)) \]
\[ y_2 = y_d(iwk(kk-1)) \]
\[ x_3 = x_d(iwk(kk)) \]
\[ y_3 = y_d(iwk(kk)) \]

\[ a = x_2 - x_1 \]
\[ b = x_3 - x_1 \]
\[ c = y_2 - y_1 \]
\[ d = y_3 - y_1 \]

\[ abcd = a \cdot d - b \cdot c \]

\[ U_{parX} = d / abcd \]
\[ V_{parX} = -c / abcd \]

The coordinates of the Barycentric point of the knt-th triangle follow

\[ x_{baryc} = (x_1 + x_2 + x_3) / 3.0 \]
\[ y_{baryc} = (y_1 + y_2 + y_3) / 3.0 \]

Run IBIRAN in order to create the coefficients of the polynomial that describes the surface in the u-v system. To make sure that the correct triangle is considered IBIRAN is asked to run for the BARYCENTRIC POINT of the triangle. This point lies in the triangle no matter what !!!!!!!!!!!!!!!!!!!!!

call ibiran(nd,xd,yd,zd,iop,maxnxi,nxi,xbaryc,nyi,ybaryc, 1 zbaryc,iwk,wk,ierr)

Create the (x,y) pairs. These are the points where the value of the integrand is needed in the x-y system. In
this system the triangular domain is defined by the points
\((x_1, y_1), (x_2, y_2)\) and \((x_3, y_3)\)
call \text{uvtoxy}(ngauss, x, y, u, v, x_1, x_2, x_3, y_1, y_2, y_3)
Create the \(Z\) and \(Zx\) values. \(Z(k, j)\) is the value of
the reflector surface \(Z=Z(x,y)\) calculated at the point
\(x=x(k,j), y=y(k,j)\). \(Zx(k,j)\) is the value of the partial
derivative of \(Z\) with respect to \(x\) at the aforementioned point
\(- (x(k,j), y(k,j)) -

\[
\begin{align*}
do & 50 \ k=1, \text{ngauss} \\
u & = u(k) \\
do & 49 \ j=1, \text{ngauss} \\
v & = v(k,j) \\
\text{as0} & = P00 + v(k,j) \times (P01 + v(k,j) \times (P02 + v(k,j) \times (P03 + v(k,j) \times (P04 + v(k,j) \times P05)))) \\
\text{as1} & = P10 + v(k,j) \times (P11 + v(k,j) \times (P12 + v(k,j) \times (P13 + v(k,j) \times P14))) \\
\text{as2} & = P20 + v(k,j) \times (P21 + v(k,j) \times (P22 + v(k,j) \times P23)) \\
\text{as3} & = P30 + v(k,j) \times (P31 + v(k,j) \times P32) \\
\text{as4} & = P40 + v(k,j) \times P41 \\
\text{Z}(k,j) & = \text{as0} + u(k) \times (\text{as1} + u(k) \times (\text{as2} + u(k) \times (\text{as3} + u(k) \times (\text{as4} + u(k) \times P50)))) \\
\text{Zu} & = \text{as1} + u(k) \times (2.0 \times \text{as2} + u(k) \times (3.0 \times \text{as3} + u(k) \times (4.0 \times \text{as4} + u(k) \times 5.0 \times P50)))) \\
\text{as0} & = P01 + v(k,j) \times (2.0 \times P02 + v(k,j) \times (3.0 \times P03 + v(k,j) \times (4.0 \times P04 + v(k,j) \times (5.0 \times P05)))) \\
\text{as1} & = P11 + v(k,j) \times (2.0 \times P12 + v(k,j) \times (3.0 \times P13 + v(k,j) \times (4.0 \times P14)))) \\
\text{as2} & = P21 + v(k,j) \times (2.0 \times P22 + v(k,j) \times (3.0 \times P23)) \\
\text{as3} & = P31 + v(k,j) \times (2.0 \times P32) \\
\text{Zv} & = \text{as0} + u(k) \times (as1 + u(k) \times (as2 + u(k) \times (as3 + u(k) \times P41 ))) \\
\text{Zx}(k,j) & = \text{Zu} \times \text{UparX} + \text{Zv} \times \text{VparX} \\
49 & \text{continue} \\
50 & \text{continue}
\end{align*}
\]
Create the array \(\text{Ro}(k,j)\). This array contains the distance
of the point \((x(k,j), y(k,j), z(k,j))\) from the observation point
\((Xo,Yo,Zo)\).
 ALSO
Create the complex arrays \(\text{SCA}(k,j)\). These arrays contain
the common coefficients in front of all the field compo-
nents.

\[
\begin{align*}
do & 58 \ k=1, \text{ngauss} \\
do & 57 \ j=1, \text{ngauss} \\
\text{Ro}(k,j) & = \sqrt{((x(k,j)-Xo) \times (x(k,j)-Xo)) + \frac{((y(k,j)-Yo) \times (y(k,j)-Yo))}{2} + \frac{((z(k,j)-Zo) \times (z(k,j)-Zo))}{2}} \\
\text{RR} & = \text{Ro}(k,j) \\
\text{JKRZ} & = \text{cmplx}(0.0, -\beta \times (\text{RR} - z(k,j))) \\
\text{SCA1}(k,j) & = \text{cexp}(\text{JKRZ}) / \text{RR} \\
\text{SCA2}(k,j) & = \text{cmplx}(\beta \times \beta - 3.0 / (\text{RR} \times \text{RR}), -3.0 \times \beta / \text{RR}) \times \text{Zx}(k,j) / \text{RR} \\
\text{SCA3}(k,j) & = \text{cmplx}(1.0 / (\text{RR} \times \text{RR}) - \beta \times \beta , \beta / \text{RR}) \\
\end{align*}
\]
call calint(Ex, ngauss, x, y, z, zr, Ro, SCA1, SCA2, SCA3, Exknt, gw, gz, u, v)
call calint(Ey, ngauss, x, y, z, zr, Ro, SCA1, SCA2, SCA3, Eyknt, gw, gz, u, v)
call calint(Ez, ngauss, x, y, z, zr, Ro, SCA1, SCA2, SCA3, Ezknt, gw, gz, u, v)

C**********************************************************************
C Call the "CALINT" subroutine to calculate the
C Integrals on each triangular domain

Extot=Extot + Exknt
Eytot=Eytot + Eyknt
Eztot=Eztot + Ezknt

5 continue
Extot=Extot/cmplx(1.0 , 2.0*pi*beta)
Eytot=Eytot/cmplx(1.0 , 2.0*pi*beta)
Eztot=Eztot/cmplx(1.0 , 2.0*pi*beta)

C'''' Transform the E-fields back to the tilded coordinate system '''''}

Estil(1,lobe)=R11*Extot+R21*Eytot+R31*Eztot
Estil(2,lobe)=R12*Extot+R22*Eytot+R32*Eztot
Estil(3,lobe)=R13*Extot+R23*Eytot+R33*Eztot

C...............................................,.....,.,.....................72
C write(3,*)" "& R11*Xo+R21*Yo+R31*Zo,"correct"
C write(3,*)" "& R12*Xo+R22*Yo+R32*Zo,"correct"
C write(3,*)" "& R13*Xo+R23*Yo+R33*Zo,"correct"
C write(3,*)" Xotil(lobe),Yotil(lobe),Zotil(lobe),"coord. in til"
C write(3,*)" ,Estil(1,lobe)," Esxtil"
C write(3,*)" ,Estil(2,lobe)," Esytil"
C write(3,*)" ,Estil(3,lobe)," Esztil"
C write(3,*)"

6 continue
return
end

C...............................................72

C This subroutine rotates the antenna as well as the observation
C points into the incident wave coordinate system!
subroutine rotate(Axt,Ayt,Azt,Ax,Ay,Az,n,a1,a2,a3)
dimension Ax(n),Ay(n),Az(n),Axt(n),Ayt(n),Azt(n)

c1=cos(a1)
c2=cos(a2)
c3=cos(a3)
s1=sin(a1)
s2=sin(a2)
s3=sin(a3)
R11=c3*c2*c1-s3*s2
R12=c3*s2*c1+s3*c2
R13=-c3*s1
R21=-s3*c2*c1-c3*s2
R22=-s3*s2*c1+c3*c2
R23=s3*s1
R31=s1*c3
R32=s1*s2
R33=c1
do 1 i=1,n
Ax(i)= R11*Axt(i) + R12*Ayt(i) + R13*Azt(i)
Ay(i)= R21*Axt(i) + R22*Ayt(i) + R23*Azt(i)
Az(i)= R31*Axt(i) + R32*Ayt(i) + R33*Azt(i)
1 continue

return
dend

subroutine calint(f,n,x,y,z,zx,Ro,SCA1,SCA2,SCA3,result,gw,gz,u,v)
dimension u(n),v(n,n),x(n,n),y(n,n),z(n,n),zx(n,n)
real Ro(n,n)
complex SCA1(n,n),SCA2(n,n),SCA3(n,n)
dimension gw(n),gz(n)
double precision gw,gz
COMMON /IBCDPT/ X0,Y0,AP,BP,CP,DP,PO1,P10,P20,P30,P40,P50,
              P01,P11,P21,P31,P41,P02,P12,P22,P32,P03,P13,
              P23,P04,P14,P05,ITPV
common /vertex/ x1,x2,x3,y1,y2,y3
complex f,Tkj,result,res
external f

This program calculates the double integral
\[ I = \int f(x,y,z,zx) \, dx \, dy \quad \text{on a triangular domain} \]
where \( z = z(x,y) \) and \( zx \) is partialz/partialx (also function of \( x,y \)).
The domain is determined by the vertices of the triangle
\((x1,y1), (x2,y2)\) and \((x3,y3)\) GIVEN COUNTERCLOCKWISE !!!!
If the vertices are not arranged counterclockwise then
the value of I is the opposite of what it should be !!!

abcd as calculated below is the Jacobian of the mapping from the x-y to the u-v system. It is equal to the ratio of the areas of the triangular domain in the two systems. Since in the u-v system the triangle is always given by the points [0,0], [1,0] and [0,1], in this system its area is 1/2. Thus, it is not a surprise that abcd equals twice the area of the triangle in the x-y system (defined by the points (x1,y1), (x2,y2) and (x3,y3)) !!!!!!!!!!!!!!!!!!!!

\[ \text{abcd} = \text{a} \times \text{d} - \text{b} \times \text{c} \]
\[ \text{if(abcd.eq.0.0) goto 666} \]

Calculate the integral by using Gaussian Quadrature in 2-dimensions. For the algorithm see my notes.

This part of the subroutine calculates the values of the function \( T(r,s) \) at the points that are used by the Gauss-Legendre quadrature method. These values are used in the main in a summation scheme to provide the approximation to the integral. Since in this program we have used \( m=n \), \( r(k) \) and \( s(k) \) are exactly the same. For a more general approach read my notes.

A very important note should be made here. abcd is the value of the Jacobian of the mapping from the x-y to u-v. Since abcd is not the absolute value of the Jacobian one needs to make sure that the points \( (x1,y1), (x2,y2) \) & \( (x3,y3) \) are ordered counterclockwise. For more details look up my notes.

\[ \text{result} = (0.0,0.0) \]
\[ \text{res} = (0.0,0.0) \]
\[ \text{do 2 k=1,n} \]
\[ \text{do 1 j=1,n} \]
\[ \text{Tkj=0.125} \times (1.-g\text{z}(k)) \times \text{abcd} \times \text{f} ( x(k,j),y(k,j),z(k,j),z\text{x}(k,j),\text{Ro}(k,j), \]
\[ \text{2} \times \text{SCA1}(k,j),\text{SCA2}(k,j),\text{SCA3}(k,j) ) \]
\[ \text{res} = \text{res} + \text{gw}(j) \times \text{Tkj} \]
\[ \text{1 continue} \]
\[ \text{result} = \text{result} + \text{gw}(k) \times \text{res} \]
\[ \text{res} = (0.0,0.0) \]
\[ \text{2 continue} \]
\[ \text{go to 667} \]

666 write(0,*),"There is something wrong with the triangle. ",
   1"Colinear vertices."
667 continue
return
end

subroutine uvtoxy(n,x,y,u,v,x1,x2,x3,y1,y2,y3)
dimension u(n),v(n,n),x(n,n),y(n,n)
a=x2-x1
b=x3-x1
c=y2-y1
d=y3-y1
abcd= a*d - b*c
do 2 k=1,n
do 1 j=1,n
x(k,j)=x1 + a * u(k) + b * v(k,j)
y(k,j)=y1 + c * u(k) + d * v(k,j)
1 continue
2 continue
return
end

subroutine rstouv(n,gz,u,v)
dimension gz(n),u(n),v(n,n)
double precision gz

do 1 k=1,n
u(k)=0.5 + 0.5*gz(k)
1 continue

do 3 k=1,n
do 2 j=1,n
v(k,j)=(1.0-gz(k)) * (1.0+gz(j)) / 4.0
2 continue
3 continue
return
end

complex function Ex(x,y,z,zx,Ro,SCA1,SCA2,SCA3)
Here Ro and the SCA's are constants since Ex is called by
C
C
x(k,j),y(k,j),z(k,j),zx(k,j),Ro(k,j),SCA’s(k,j)
common /observ/ Xo,Yo,Zo,beta
complex SCA1,SCA2,SCA3
Ex=SCA1 * ( SCA2*(Xo-x)/Ro + SCA3 )
return
end

complex function Ey(x,y,z,zx,Ro,SCA1,SCA2,SCA3)
Here Ro and the SCA's are constants since Ey is called by x(k,j),y(k,j),z(k,j),zx(k,j),Ro(k,j),SCA's(k,j)
common /observ/ Xo,Yo,Zo,beta
complex SCA1,SCA2,SCA3
Ey=SCA1 * SCA2*(Yo-y)/Ro
return
end

complex function Ez(x,y,z,zx,Ro,SCA1,SCA2,SCA3)
Here Ro and the SCA's are constants since Ez is called by x(k,j),y(k,j),z(k,j),zx(k,j),Ro(k,j),SCA's(k,j)
common /observ/ Xo,Yo,Zo,beta
complex SCA1,SCA2,SCA3
Ez=SCA1 * ( SCA2*(Zo-z)/Ro + SCA3*zx )
return
end

c subroutine gauss4(n,z,w)
double precision z,w
dimension z(n),w(n)
n=4
nn=n/2
z(1)=.339981043584856d0
z(2)=.861136311594053d0
z(3)=.652145154862546d0
z(4)=.347854845137454d0
w(1)=.652145154862546d0
w(2)=.347854845137454d0
w(3)=.339981043584856d0
w(4)=.861136311594053d0

do 1 k=1,nn
z(nn+k)=-z(k)
w(nn+k)=w(k)
1 continue

do 2 k=1,nn
z(k)=-z(n+1-k)
w(k)=w(n+1-k)
2 continue
return
end
subroutine gauss6(n,z,w)
double precision z,w
dimension z(n),w(n)
n=6
nn=n/2

z(1)=.238619186083197d0
z(2)=.661209386466265d0
z(3)=.932469514203152d0

w(1)=.467913934572691d0
w(2)=.360761573048139d0
w(3)=.171324492379170d0

do 1 k=1,nn
   z(nn+k)=-z(k)
   w(nn+k)=w(k)
1 continue

do 2 k=1,nn
   z(k)=-z(n+1-k)
   w(k)=w(n+1-k)
2 continue
return
end

subroutine gauss8(n,z,w)
double precision z,w
dimension z(8),w(8)

The array z contains the zeros of the Legendre polynomial of the 8th degree. The array w contains the appropriate weights for the Gaussian quadrature. The values of w(k) are given by

\[ w(i) = \frac{2}{(1-z(i)^2) \left( P'(z(i))^2 \right)} \]

where \( P(z(i)) = 0 \) for \( i = 1, 2, \ldots, 8 \) and \( P'(z) = \frac{dP(z)}{dz} \)

n=8
nn=n/2

z(1)=.183434642495650d0
z(2)=.525532409916329d0
z(3)=.796666477413627d0
z(4)=.960289856497536d0
C subroutine gaus10(n,z,w)
    double precision z,w
    dimension z(n),w(n)

    n=10
    nn=n/2

    z(1)=.148874338981631d0
    z(2)=.433395394129247d0
    z(3)=.679409568299024d0
    z(4)=.865063366688985d0
    z(5)=.973906528517172d0

    w(1)=.295524224714753d0
    w(2)=.269266719309996d0
    w(3)=.219086362515982d0
    w(4)=.149451349150581d0
    w(5)=.066671344308688d0

    do 1 k=1,nn
    z(nn+k)=-z(k)
    w(nn+k)=w(k)
1 continue

    do 2 k=1,nn
    z(k)=-z(n+1-k)
    w(k)=w(n+1-k)
2 continue
    return
end
subroutine gaus12(n,z,w)
double precision z,w
dimension z(n),w(n)
n=12
nn=n/2

z(1)=.125233408511469d0
z(2)=.367831498998180d0
z(3)=.587317954286617d0
z(4)=.769902674194305d0
z(5)=.904117256370475d0
z(6)=.981560634246719d0
w(1)=.249147045813403d0
w(2)=.233492536538355d0
w(3)=.203167426723066d0
w(4)=.160078328543346d0
w(5)=.106939325995318d0
w(6)=.047175336386512d0

do 1 k=1,nn
  z(nn+k)=-z(k)
  w(nn+k)=w(k)
1 continue

do 2 k=1,nn
  z(k)=-z(n+1-k)
  w(k)=w(n+1-k)
2 continue
return
end

subroutine gaus16(n,z,w)
double precision z,w
dimension z(8),w(8)

The array z contains the zeros of the Legendre polynomial of the 16th degree. The array w contains the appropriate weights for the Gaussian quadrature. The values of w(k) are given by

\[ w(i) = \frac{2}{(1-z(i)^2) \cdot (P'(z(i))^2) } \]

where \[ P(z(i)) = 0 \quad i=1,2,\ldots,16 \] and \[ P'(z) = \frac{dP(z)}{dz} \]

n=16
nn=n/2

z(1) = 0.95012509836637440185d0
z(2) = 0.21603550779258913230d0
z(3) = 0.45016777657223786342d0
z(4) = 0.617876244402643748447d0
z(5) = 0.755404408355003033895d0
z(6) = 0.865631202387831743880d0
z(7) = 0.944575023073232576078d0
z(8) = 0.989400934991649932596d0

w(1) = 0.189450610455068496285d0
w(2) = 0.182603415044923558867d0
w(3) = 0.169156519395002538189d0
w(4) = 0.149595988816576732081d0
w(5) = 0.124628971255533872052d0
w(6) = 0.095158511682492784810d0
w(7) = 0.062253523938647892863d0
w(8) = 0.027152459411754094852d0

do 1 k=1,nn
z(nn+k)=-z(k)
w(nn+k)=w(k)
1 continue

do 2 k=1,nn
z(k)=-z(n+1-k)
w(k)=w(n+1-k)
2 continue

return
end

subroutine gaus48(n,z,w)

The array z contains the zeros of the Legendre polynomial of the 48th degree. The array w contains the appropriate weights for the Gaussian quadrature. The values of w(k) are given by

\[ w(i) = \frac{2}{(1-z(i)^2) \cdot (P'(z(i))^2)} \]

where \( P(z(i)) = 0 \quad i=1,2,...,48 \)
and \( P'(z) = \frac{dp(z)}{dz} \)

dimension z(n),w(n)
double precision z,w
\( n = 48 \)
\( n_n = n/2 \)

\[
\begin{align*}
z(1) & = .032380170962869362033d0 \\
z(2) & = .097004699209462698930d0 \\
z(3) & = .161223656068891178056d0 \\
z(4) & = .224763790394689061225d0 \\
z(5) & = .287362487355455576736d0 \\
z(6) & = .348755886292160738160d0 \\
z(7) & = .408686481990716729916d0 \\
z(8) & = .466902904750958404545d0 \\
z(9) & = .523160974722233033678d0 \\
z(10) & = .577224726083972703818d0 \\
z(11) & = .62868739677513623995d0 \\
z(12) & = .677872379632663905212d0 \\
z(13) & = .724034130923814654674d0 \\
z(14) & = .767159032515740339254d0 \\
z(15) & = .807066204029442627083d0 \\
z(16) & = .84358826162439350711d0 \\
z(17) & = .876572020274247885906d0 \\
z(18) & = .905879136715569672822d0 \\
z(19) & = .931386690706554333114d0 \\
z(20) & = .952987703160430860723d0 \\
z(21) & = .970591592546247250461d0 \\
z(22) & = .984124583722826857745d0 \\
z(23) & = .993530172266350757548d0 \\
z(24) & = .998771007252426118601d0 \\
\end{align*}
\]

\[
\begin{align*}
w(1) & = .064737696812683922503d0 \\
w(2) & = .064466164435950082207d0 \\
w(3) & = .063924238584648186624d0 \\
w(4) & = .063114192286254025657d0 \\
w(5) & = .062039423159829663904d0 \\
w(6) & = .060704439165893880053d0 \\
w(7) & = .059114839698395635746d0 \\
w(8) & = .057277292100403215705d0 \\
w(9) & = .055199503699984162868d0 \\
w(10) & = .052890189485193667096d0 \\
w(11) & = .050359035553854474958d0 \\
w(12) & = .047616658492490474826d0 \\
w(13) & = .044674560856694280419d0 \\
w(14) & = .041545082943464749214d0 \\
w(15) & = .038241351065830706317d0 \\
w(16) & = .034777222564770438893d0 \\
w(17) & = .031167227832798088902d0 \\
w(18) & = .027426509708356948200d0 \\
w(19) & = .023570760839324379141d0 \\
w(20) & = .019616160457355527814d0 \\
w(21) & = .015579315722943848728d0 \\
w(22) & = .011477234579234539490d0 \\
w(23) & = .007327553901276262102d0 \\
w(24) & = .003153346052305838633d0
\end{align*}
\]
do 1 k=1,nn
z(nn+k)=-z(k)
w(nn+k)=w(k)
1 continue

do 2 k=1,nn
z(k)=-z(n+1-k)
w(k)=w(n+1-k)
2 continue
return
end