"MODESRCH", AN IMPROVED COMPUTER PROGRAM FOR OBTAINING ELF/VLF/LF MODE CONSTANTS IN AN EARTH-IONOSPHERE WAVEGUIDE

Interim Report No. 77T

D. G. Morfitt and C. H. Shellman

PROPAGATION TECHNOLOGY DIVISION
NAVAL ELECTRONICS LABORATORY CENTER
SAN DIEGO, CALIFORNIA 92152

1 OCTOBER 1976

Prepared for Defense Nuclear Agency
on DNA Subtask HB-042
NELC PROBLEM M402
**'MODESRCH', An Improved Computer Program for Obtaining ELF/VLF/LF Mode Constants in an Earth-Ionosphere Waveguide**

**Authors:**
David G. Morfit and Charles H. Shellman

**Performing Organization Name and Address:**
Naval Electronics Laboratory Center
271 Catalina Blvd.
San Diego, CA 92152

**Monitoring Agency Name and Address:**
Defense Nuclear Agency
Washington, DC 20305

**Report Date:**
1 October 1976

**Distribution Statement:**
Approved for public release; distribution is unlimited.

**Key Words:**
- Very low frequency Waveguide modes
- Low frequency Roots of a complex function
- Waveguide propagation
'MODESRCH', An Improved Computer Program for Obtaining ELF/VLF/LF Mode Constants in an Earth-Ionosphere Waveguide.

by

D. G. MORFITT

and

C. H. SHELLMAN
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abstract</td>
<td>v</td>
</tr>
<tr>
<td>I. Introduction</td>
<td>1</td>
</tr>
<tr>
<td>II. Discussion of Modesrch Procedure</td>
<td>4</td>
</tr>
<tr>
<td>III. The General Root-Finder</td>
<td>37</td>
</tr>
<tr>
<td>IV. Structure and Description of the Modesrch Computer Program</td>
<td>53</td>
</tr>
<tr>
<td>V. Running the Program</td>
<td>70</td>
</tr>
<tr>
<td>VI. References</td>
<td>106</td>
</tr>
<tr>
<td>VII. Appendices</td>
<td>108</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>Page #</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>iv</td>
</tr>
<tr>
<td>I  INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>II DISCUSSION OF MODESrch PROCEDURE</td>
<td>4</td>
</tr>
<tr>
<td>A. General Background.</td>
<td>4</td>
</tr>
<tr>
<td>B. Introduction to MODESrch Theory.</td>
<td>5</td>
</tr>
<tr>
<td>C. General Steps in the Solution of the Modal Equation.</td>
<td>7</td>
</tr>
<tr>
<td>D. Optimization of Input Ionospheric Density Profiles.</td>
<td>10</td>
</tr>
<tr>
<td>E. Integration of the Differential Equations for $\bar{X} = \frac{R + 1}{C}$ Through the Ionosphere.</td>
<td>12</td>
</tr>
<tr>
<td>F. Integration of $\bar{X} = \frac{(R + 1)}{C}$ Through Free Space over a Curved Earth.</td>
<td>16</td>
</tr>
<tr>
<td>G. The Ground Reflection Coefficient Matrices.</td>
<td>24</td>
</tr>
<tr>
<td>H. Excitation Factors, Height Gains and Auxiliary Functions</td>
<td>29</td>
</tr>
<tr>
<td>I. Use of Lagrange Interpolation</td>
<td>34</td>
</tr>
<tr>
<td>J. Comments on the Differentiation of $F(c)$ wrt $\theta$</td>
<td>35</td>
</tr>
<tr>
<td>III THE GENERAL ROOT-FINDER</td>
<td>37</td>
</tr>
<tr>
<td>IV</td>
<td>STRUCTURE AND DESCRIPTION OF THE MODFSRCH COMPUTER PROGRAM</td>
</tr>
<tr>
<td>----------</td>
<td>------------------------------------------------------------------------</td>
</tr>
<tr>
<td>A.</td>
<td>Program Structure</td>
</tr>
<tr>
<td>B.</td>
<td>Computer Program Description</td>
</tr>
<tr>
<td>V</td>
<td>RUNNING THE PROGRAM</td>
</tr>
<tr>
<td>A.</td>
<td>Control Cards</td>
</tr>
<tr>
<td>B.</td>
<td>Input Variables</td>
</tr>
<tr>
<td>C.</td>
<td>Program Outputs</td>
</tr>
<tr>
<td>D.</td>
<td>Examples of Program Inputs and Outputs</td>
</tr>
<tr>
<td>E.</td>
<td>Error Messages</td>
</tr>
<tr>
<td>VI</td>
<td>REFERENCES</td>
</tr>
<tr>
<td>VII</td>
<td>APPENDICES</td>
</tr>
<tr>
<td>A.</td>
<td>Modification of the Modal Equation.</td>
</tr>
<tr>
<td>B.</td>
<td>Derivation of the Equations for Free Space Integration</td>
</tr>
<tr>
<td>C.</td>
<td>Modified Ground Reflection Coefficients</td>
</tr>
<tr>
<td>D.</td>
<td>Mesh Square Analysis</td>
</tr>
<tr>
<td>E.</td>
<td>Solution of the Quadratic Equation for the General Root-Finder</td>
</tr>
<tr>
<td>F.</td>
<td>Fortran Listing of the MODFSRCH Computer Program</td>
</tr>
</tbody>
</table>
ABSTRACT:

MODESRCH is a FORTRAN computer program designed to obtain rapidly, economically, and accurately, the mode constants needed to describe propagation in the earth-ionosphere waveguide at ELF, VLF and LF (10 Hz to 60 kHz). It is suggested that MODESRCH, along with required mode summing procedures, be used to replace "waveguide" and "wave-hop" codes currently used in network analysis programs which cover the ELF to LF frequency bands. This report contains a discussion of the analytical approach taken in MODESRCH, a FORTRAN listing of the program, instructions for using the program and some sample calculations.
INTRODUCTION

Radio field strengths at ELF, VLF and LF (10 Hz to about 60 kHz) are usually calculated using procedures based on either normal mode theory (reference 1) or on a semi-geometrical optics approach (reference 2).

Normal mode theory treats the field as being composed of one or more discrete families (modes) of plane waves confined to the natural waveguide bounded by the earth and the ionosphere. The number of waveguide modes needed to adequately represent the field is proportional to the earth-ionosphere separation measured in radio wavelengths. Thus, only a single mode is required below 3 kHz, about ten to fifteen modes at 30 kHz, and as many as twenty-five modes may be important at 60 kHz. The propagation parameters for each mode depend on the characteristics of the earth and the ionosphere, and are independent of the relative positions of the transmitter and receiver. For a homogeneous waveguide, once all of the important modes have been found, the radio fields can be determined as a function of distance by a simple mode summing calculation.

Computer programs based on waveguide mode theory have been brought to a high degree of sophistication and versatility in recent years, making it possible to calculate fields for any antenna configuration at any altitude, for both vertically and horizontally polarized waves, and for inhomogeneous waveguides. Unfortunately, solutions to the modal equation cannot be determined explicitly. Instead, the complex angles of incidence at the waveguide boundaries, which uniquely identify each mode, must be determined by iterative techniques. At frequencies above about 30 kHz, where there is a large number of modes to be found, the mode finding procedure of reference 1 may miss some solutions and it has taken human judgment to determine if the set of modes
found was complete. If the set of modes was judged incomplete, new initial conditions had to be resubmitted to the computer in an attempt to compute the missing modes, resulting in additional computer and manpower costs. Systems analysis codes do not allow human intervention, and so they have not been able to rely on mode theory when a sizeable number of modes is needed to represent the radio fields.

Previous practice has been to compute radio fields at frequencies below 30 kHz using waveguide mode theory and to use a wave-hop theory, such as described in reference 2, for computing fields at and above 30 kHz. The wave-hop procedure considers the total field at the receiver to be composed of a groundwave component and a number of skywaves following unique geometric ray paths connecting the transmitter and receiver with intermediate reflections off the earth and the ionosphere. Calculations of radio field strength, using the wave-hop procedure of reference 2, agree well with waveguide mode calculations for horizontally homogeneous normal ionospheres. They do not agree for severely depressed ionospheres over low conductivity ground.

The wave-hop approach has features which make it somewhat inconvenient for system analyses. For each separation distance between transmitter and receiver there is a unique ray geometry. Thus, for calculations of fields as a function of distance; e.g., for aircraft operations, certain inherent calculations must be repeated many times. Also, for inhomogeneous waveguides such as across day-night ionospheric gradients or across nuclear-induced ionospheric depressions, the geometry is ill-defined and the calculations tend toward approximations.

Aside from the limitations of either the waveguide approach or the wave-hop approach to long-wave propagation prediction, it would be advantageous to
use a single method across the entire ELF, VLF, LF frequency bands. Systems
codes used for network analyses would be considerably reduced in size, saving
computer costs. Comparisons of systems performance as a function of frequency
would be more meaningful if a single propagation theory were invoked.

The MODESRCH computer program described in this report has the required
ability to compute waveguide mode constants rapidly, economically and accur-
ately over the frequency band of interest. MODESRCH contains all of the basic
features of the previously reported waveguide mode computational procedures of
reference 1 and several important additions. The new additions virtually
guarantee that all waveguide modes of importance will be found at any frequency
up to at least 60 kHz. MODESRCH, used with presently available mode summing
techniques, will provide the radio fields required for systems analysis.

This report describes the MODESRCH theory. It also contains a listing
of the FORTRAN program, instructions for the user, and examples of MODESRCH
calculations.
II. DISCUSSION OF MODESRCH PROCEDURE

A. General Background

In the propagation of ELF, VLF and LF terrestrial radio waves to great distances, the waves are confined within the space between the earth and the ionosphere. This space acts as a waveguide, and the "waveguide concept" is applicable for characterizing the propagated fields as a function of distance.

The energy within the waveguide is considered to be partitioned among a series of modes. Each mode represents a resonant condition; i.e., for a discrete set of angles of incidence of the waves on the ionosphere, resonance occurs and energy will propagate away from the source. The complex angles \( \theta \) for which this occurs are called the eigenangles. They may be obtained using the 'full-wave' procedures described in reference 1 by solving the determinantal equation

\[
F_1(\theta) = | \mathcal{R}_d(\theta) \mathcal{R}_d(\theta) - 1 | = 0
\]  

(1)

where

\[
\mathcal{R}_d(\theta) = \begin{bmatrix} \mathcal{R}_{\|d}(\theta) & \mathcal{R}_{\perp d}(\theta) \\ \mathcal{R}_{\perp d}(\theta) & \mathcal{R}_{\|d}(\theta) \end{bmatrix}
\]  

(2)

is the complex ionospheric reflection coefficient matrix looking up into the ionosphere from height "d" and

\[
\bar{\mathcal{R}}_d(\theta) = \begin{bmatrix} \bar{R}_{\|d}(\theta) & 0 \\ 0 & \bar{R}_{\perp d}(\theta) \end{bmatrix}
\]  

(3)

is the complex reflection matrix looking down from height "d" towards the ground.

Note: Any variable "A" denoted by \( \mathcal{A} \) in this report implies a matrix.
The complex angle, $\theta$, is the angle of incidence at height $z = H$ where the modified index of refraction given by

$$n^2 = 1 - \alpha(H - Z) = 1 - \frac{2}{r_e} (H - Z)$$

is unity. Here $r_e$ is the radius of the earth.

The notation $\parallel$ for the $R$'s and $R'$s denotes vertical polarization while the notation, $\perp$, denotes horizontal polarization. The first subscript on the $R$'s refers to the polarization of the incident wave while the second applies to the polarization of the reflected wave.

To find solutions of (1) for the eigenangles, $\theta$, the well known Newton-Raphson method may be used. The procedure is to guess a solution $\theta_0$ to the equation $F_1(\theta) = 0$. The function $F_1(\theta)$ is then reevaluated for $\theta_0 + \delta \theta$ and the correction to $\theta_0$ found from the equation

$$\Delta \theta = -\frac{F_1(\theta_0)\delta \theta}{F_1(\theta_0 + \delta \theta) - F_1(\theta_0)}.$$  

The correction determined by (5) is then evaluated and the process repeated until the quantities $|\Delta \theta_1|$ and $|\Delta \theta_2|$ are reduced to within a preassigned tolerance.

The subscripts $r$ and $i$ denote the real and imaginary parts, respectively. The final solution for mode "n" is then given by

$$[\theta_n] = [\theta_n] + \Delta \theta_n \quad j = 1, 2, 3, \ldots$$

B. Introduction to MODESRCH Theory

The "MODESRCH" procedure also obtains the full-wave solution of equation 1 in terms of the eigenangles. In this case; however, the $R$ and $R'$ terms are modified in order to make simpler equations.
The basic premise of the MODESRCH theory is to modify the function $F_1(\theta)$ so that it contains no poles and then in this form, apply a newly-developed root-finding algorithm to obtain the complete set of eigenangles.

In particular, the $F_1(\theta)$ equation of reference 1 becomes upon modification (see Appendix A):

$$F_3(\theta) = (\|n_\| - \|X\|d_\|) (\|n_\| - \|X\|d_\|) - \|X\|Xd_\|d_\|$$  \hspace{1cm} (7)

where $\bar{X} = (R + 1)/c$  \hspace{1cm} (8)

and $c = \cos \theta$ (\(\theta = \) complex angle of incidence)

or $\|X\| = (\|R\| + 1)/c$  \hspace{1cm} (9)

The modified variables:

are related to making simpler equations in the 'MODESRCH' procedure and in removing the zeros in $F_3(\theta)$ at $\theta = 90^\circ$. The functions $F_1(\theta)$ and $F_3(\theta)$ are not the same, but except for the point where $\theta=90^\circ$, they have the same zeros.

The solutions to the modal equation; \(\text{i.e., equation} 7\) for $F_3(\theta)=0$ are the complex eigenangles $\theta$ needed to obtain the mode constants (e.g., phase velocity, attenuation rate, and excitation factor) which

Note: From this point on, let $F_3(\theta)$ be denoted by $F(\theta)$ for convenience.
are the parameters of interest in computing field strengths. The total field at any point along the guide may be computed from the vector sum of the contributions from each individual mode.

As in reference 1, the MODESRCH procedure requires that the upper boundary of the waveguide be assigned arbitrary electron and ion density distributions with height as well as variable collision frequency profiles with height. Allowance is also made for the anisotropy of the ionosphere. The lower boundary of the guide is taken to be a smooth homogeneous earth characterized by an adjustable conductivity and permittivity. Earth curvature is also included via the artifice of the modified refractive index (e.g., Equation (4)).

The (X,Y,Z) coordinate system used is such that Z is directed into the ionosphere, the X-Z plane is the plane of propagation, and invariance in the Y-axis is assumed.

C. General Steps in the Solution of the Modal Equation

(1) For VLF/LF cases the input electron and ion density profiles vs height are modified by eliminating those heights above which and below which the corresponding particle densities will have negligible effects on the modal solutions. For ELF cases, profiles are not cutoff at the top or the bottom heights but instead the complete input profile is used in subsequent calculations.

(2) The modal solutions of Equation (7), the complex eigenangles ($\Theta = \Theta_{\text{real}} + i\Theta_{\text{imaginary}}$), are determined within a bounded region of the complex "THETA" plane defined by a set of limiting THETA values. This bounded region will be called the "contour rectangle" and the limiting THETA values are denoted by the corners of the rectangle (e.g., A,B,C, and D). These parameters are portrayed in Figure 1.
(3) Four fixed angles (θT_i, i=1,2,3,4) are chosen in the complex THETA plane. These angles are somewhat outside the corners A, B, C, and D of the contour rectangle. For VLF and LF calculations these angles are the given values of θ at which full-wave solutions are to be carried out for use in a Lagrange interpolation procedure. At ELF (e.g., < 1000 Hz) the purpose of these angles is to identify the location of the contour rectangle in the THETA plane.

(4) At VLF and at LF, the differential equations for the ionospheric terms \( X(\theta) \) are integrated at the angles θT_i, to give full-wave values of \( X(\theta T_i) \) at the base of the modified electron-ion density profiles. The results obtained by integrating to the bottom of the profile (e.g., Z=Z_b) will be designated as \( X_b(\theta T_i) \).

(5) For VLF/LF cases a further upward integration of the \( X(\theta T_i) \) equations from the height Z_b to a reference height 'd' is then performed assuming a free space environment over a curved earth. This reference height may be taken to be an effective reflection height. At this height there will be
no poles in the function $\tilde{X}(\theta)$ for the region of interest in the complex THETA plane. The determination of this height is important because the general root-finding algorithm used to locate the zeros of $F(\theta)$ requires that there be no poles in the vicinity of the zeros to be found. This reference height is also appropriate for performing Lagrange interpolation on the function $\tilde{X}(\theta)$.

(6) At the reference height 'd', for VLF and LF, the contour rectangle is subdivided into small mesh squares. The value of $F(\theta)$ at each corner of each mesh square is determined by Lagrange interpolation of $\tilde{X}(\theta)$ based on the given values of $\tilde{X}(\theta_i)$.

Recall from Equation (7) that the function $F(\theta)$ is computed from $\tilde{X}(\theta), \tilde{n}(\theta)$ and $\tilde{d}(\theta)$ so that for any angle, $\theta_j$ the value of $\tilde{X}(\theta_j)$ is found from the Lagrangian interpolation of $\tilde{X}(\theta_i), i=1,2,3,4$ whereas $n(\theta_j)$ and $d(\theta_j)$ are calculated directly.

(7) For ELF (e.g., <1000 Hz), the angles used in the full-wave integration of the $\tilde{X}(\theta)$ equation consists of the usual corners of the contour rectangle plus all additional angles formed from the four corners of each mesh square into which the contour rectangle is also divided. In this case the integration is extended downward to the earth's surface (e.g., $Z=0$) and no further integration back up to the height 'd' is done. No Lagrangian interpolation is involved since the full-wave solution is computed for all angles of interest to the problem. It is important to realize at this point that the MODESRCH procedure has not been formulated so as to eliminate the poles for ELF cases. This may result in no solution being found to the modal equation due to the presence of poles. Because of this result, it may be advisable to compute modes for the ELF cases by the methods of reference 1. This problem has not; however, occurred in practice.
(8) The zeros of \( F(\theta) \) at 'd' for VLF/LF (or at '0' for ELF) are then found by applying the root-finder algorithm to the mesh squares.

(9) Since \( F(\theta) \), as determined from step (8) is approximate, the exact solution of \( F(\theta) \) is then obtained by applying the usual Newton-Raphson procedures where the full-wave solution is used to obtain \( \bar{z}(\theta) \).

(10) With the determination of the exact solutions of the \( \theta \)'s (eigenangles), the required mode constants and excitation factors, may be calculated.

D. Optimization of Input Ionospheric Density Profiles

Computing efficiency for VLF and LF may be improved by cutting off the top and bottom heights of the input profile.

(1) Exponential Profiles (electrons only)

An exponential, electrons only, ionospheric profile may be specified as input to the MODESRCH procedure. The profile is computed as described in Wait's NBS TN 300, reference 3, where

\[
\omega_\phi(z) = 2.5 \times 10^5 \exp[\beta(z-h')] = 3.18254 \times 10^9 N(z)/v(z)
\]

and \( \beta \) is in inverse kilometres, \( z \) and \( h' \) are in kilometres, \( N(z) \) is the electron density in electrons per cubic centimetre and \( v(z) \) is the collision frequency in collisions per second. The exponential collision frequency is given in reference 3 as

\[
v(z) = 1.86 \times 10^{11} \cdot \exp(-0.15z)
\]

The top of the profile may be modified by choosing a value for the variable SCLHTS. The relationship is that:

\[
\text{Top Height} = \text{Integer } (H' + \frac{\text{SCLHTS}}{\beta} + 0.5)
\]

The nearest integral value of altitude as given by 'Top Height' is used to set the top of the profile. All heights above this value are neglected in the MODESRCH computations.
The bottom of the exponential profile is set by assigning a minimum value (i.e., ENMIN) of electron density in electrons/cm$^3$. All altitudes, at which the corresponding density is less than the value of ENMIN, will be neglected.

(2) VLF/LF - Tabular Electron-Ion Density Profiles

When the particle density profiles are in tabular form, the top and bottom of the profile may be cut off using the following criteria, as presented in reference 4. Here the particle profiles may consist of ions as well as of electrons.

The (isotropic) squared index of refraction is written as:

$$n^2(z) = 1 - A - jB$$  \hspace{1cm} (14)

where

\begin{align*}
A &= A_e + A_+ + A_- \\
B &= B_e + B_+ + B_-
\end{align*}

or

$$A = \frac{N_e e^2}{M_e\varepsilon_0 (\omega^2 + \nu_e^2)} + \frac{N_+ e^2}{M_+\varepsilon_0 (\omega^2 + \nu_+^2)} + \frac{N_- e^2}{M_-\varepsilon_0 (\omega^2 + \nu_-^2)}$$ \hspace{1cm} (15)

$$A_e \quad A_+ \quad A_-$$

and

$$B = \frac{\nu_e}{\omega} A_e + \frac{\nu_+}{\omega} A_+ + \frac{\nu_-}{\omega} A_-$$ \hspace{1cm} (16)

with the identification that

\begin{align*}
\nu_e &= \text{electron-neutral particle collision frequency (s}^{-1}\text{)} \\
\nu_+ &= \text{positive ion-neutral particle collision frequency (s}^{-1}\text{)} \\
\nu_- &= \text{negative ion-neutral particle collision frequency (s}^{-1}\text{)} \\
N_e &= \text{electron density (cm}^{-3}\text{)} \\
N_+ &= \text{positive ion density (cm}^{-3}\text{)} \\
N_- &= \text{negative ion density (cm}^{-3}\text{)}
\end{align*}
\[ \omega = \text{frequency (radians s}^{-1}) \]
\[ m_\text{e} = \text{mass of the electron (kg)} \]
\[ m_+ = \text{mass of the positive ions (kg)} \]
\[ m_- = \text{mass of the negative ions (kg)} \]
\[ \varepsilon_0 = \text{permittivity of free space} \]

The top part of the ionospheric profile which may be neglected in the calculations is defined as that altitude \(z\) above which:

\[ B(z) > \text{Maximum (TEMPO-B, } 20 \frac{\omega}{v_\text{e}}) \] (17)

where 'TEMPO-B' is assigned a value (e.g., 2.0).

An additional requirement for cutting off the top of the profile is that the electron density must be as great as 'TOPEN'. The variable 'TOPEN' is assigned a value (e.g., 1000 electrons/cm\(^2\)). The top of the profile is taken to be the higher of the two heights corresponding to 'B' (equation (17)) and to 'TOPEN'.

The bottom of the input ionospheric profile is neglected (for VLF/LF cases) at those altitudes for which the computed value of \(B\) in equation (16) is less than 'CUTOFF'. The variable 'CUTOFF' is assigned a value (e.g., 0.0001). The relationship for cutoff is:

\[ B(z) < \text{CUTOFF} \] (18)

(3) ELF Tabular Electron-Ion Density Profiles

In the case of ELF the profile top and bottom are not cut off and the complete profile is used in the calculations.

E. Integration of the Differential Equations for \( \frac{\mathbf{\tau}}{C} = (\mathbf{R} + \mathbf{1})/C \)

Through the Ionosphere

The differential equations for the ionosphere reflection matrix, \( \mathbf{R} \), are given in reference 5.
The differential equations are of the form

\[ \frac{dR}{dz} = -\frac{ik}{2} \left\{ S(21) + \frac{S(22)}{2} - R S(11) - \frac{S(12)}{2} R \right\} \]

where \( z \) is the height variable and \( S(kl) \) are 2x2 partitions of a 4x4 matrix given by:

\[
S = \begin{bmatrix}
T_{11} + T_{44} + T_{14}/C + CT_{41} & -T_{12}/C - T_{42} & -T_{11} + T_{44} + T_{14}/C - CT_{41} & -T_{12}/C - T_{42} \\
-T_{31} - T_{34}/C & C + T_{32}/C & T_{31} - T_{34}/C & -C + T_{32}/C \\
-T_{11} + T_{44} - T_{14}/C + CT_{41} & T_{12}/C - T_{42} & T_{11} + T_{44} - T_{14}/C - CT_{41} & T_{12}/C - T_{42} \\
T_{51} + T_{54}/C & C - T_{32}/C & -T_{31} + T_{34}/C & -C - T_{32}/C
\end{bmatrix}
\]

where

\[ C = \cos \theta \]
\[ \theta = \text{complex angle of incidence (deg).} \]

The intermediate matrix \( T \) relates the electromagnetic field components to their derivatives. In turn, \( T \) is defined in terms of the susceptibility matrix \( M \):

where \( T \) is defined as

\[
T = \begin{bmatrix}
-\frac{SM_{31}}{T + M_{33}} & \frac{SM_{32}}{T + M_{33}} & 0 & \frac{C^2 + M_{33}}{T + M_{33}} \\
0 & 0 & 1 & 0 \\
\frac{M_{23}M_{31}}{T + M_{33}} - M_{21} & C^2 + M_{22} - \frac{M_{23}M_{32}}{T + M_{33}} & 0 & \frac{SM_{23}}{T + M_{33}} \\
\frac{1 + M_{11}}{1 + M_{33}} - \frac{M_{13}M_{31}}{T + M_{33}} & \frac{M_{23}M_{13}}{T + M_{33}} - M_{12} & 0 & \frac{SM_{13}}{T + M_{33}}
\end{bmatrix}
\]

where \( S = \sin (\theta) \).
Finally, the susceptibility $M$ matrix is defined as

$$
M = \frac{-p}{U(U^2 - y^2)} \begin{bmatrix}
(U^2 - y^2 - 2(H - z)/r_e) (-jU_mY - \lambda_mY^2) & (jU_mY - \lambda_mY^2) \\
(jU_mY - \lambda_mY^2) & (U^2 - m^2Y^2 - 2(H - z)/r_e) (-jU_nY - \lambda_nY^2) \\
(-jU_mY - \lambda_nY^2) & (jU_nY - \lambda_nY^2) & (U^2 - n^2Y^2 - 2(H - z)/r_e)
\end{bmatrix}
$$

where

- $j = -1$
- $P = (\omega_p/\omega)^2$
- $U = 1 - j(\nu/\omega)$
- $Y = \mu_0\omega_m/\omega$ henrys
- $\omega_m$ = magnetic gyrofrequency (s$^{-1}$)
- $\omega_p$ = plasma frequency (s$^{-1}$)
- $\xi = \cos \theta_D \cos \varphi_a$
- $m = \cos \theta_D \sin \varphi_a$
- $n = -\sin \theta_D$
- $\nu$ = collision frequency (s$^{-1}$)
- $\omega$ = wave frequency (s$^{-1}$)
- $\mu_0$ = free space permeability (henrys/m)
- $\theta_D$ = magnetic dip angle (deg)
- $r_e$ = earth's radius (km)
- $H$ = altitude where the index of refraction is unity (km)
- $\varphi_a$ = direction of propagation east of magnetic north (deg)
- $z$ = altitude (km)

The matrix $M$ described above is general in that it represents a single ionospheric constituent. Three species are considered by this model: electrons and positive and negative ions. Thus for each species there will
be a matrix $M_i$, and the matrix elements needed for computing the $T$ matrix elements are found from:

$$M = \sum_{i=1}^{3} M_i$$  \hspace{1cm} (25)$$

Earth curvature terms have been added to the diagonal elements of $M$, by the terms $-2\left(\frac{\dot{z} - z}{r_e}\right)$.

Equation (19) may be rearranged to form the following differential equations for $\frac{X}{Z} = (R + 1)/C$, (see Equation (8)), giving:

$$\frac{dX}{dz} = -ik \frac{X}{Z} A + B \frac{X}{Z} + C \frac{X}{Z} D \frac{X}{Z}$$  \hspace{1cm} (26)$$

where $k$ is the wave number, $z$ is height, and

$$A = (S_{21} - S_{22} + S_{11} - S_{12})/C = \begin{pmatrix} 4T_{41} & 0 \\ 0 & 4 \end{pmatrix}$$  \hspace{1cm} (27)$$

$$B = S_{22} + S_{12} = 2 \begin{pmatrix} T_{44} & CT_{41} & -T_{42} \\ 0 & 0 & -C \end{pmatrix}$$  \hspace{1cm} (28)$$

$$C = -S_{11} + S_{12} = 2 \begin{pmatrix} T_{11} & -CT_{41} & 0 \\ T_{31} & 0 & -C \end{pmatrix}$$  \hspace{1cm} (29)$$

$$D = -C S_{12} = \begin{pmatrix} C(T_{11} - T_{14}) - T_{14} + CT_{41} & T_{12} + CT_{42} \\ -CT_{31} + T_{34} & C^2 - T_{32} \end{pmatrix}$$  \hspace{1cm} (30)$$

Equation (26) may be integrated numerically by the Runge-Kutta method. The integration is carried out in the direction of decreasing ionospheric height. Error testing is based on comparing the results of integrating through a height interval with one Runge-Kutta step with the result of integrating through the same height interval using two steps. If the difference is greater than a specified tolerance, the step size in that interval is cut in half. The initial set of integration steps is taken to be the input profile segments. However, the integration steps used in one integration are taken to be the initial set of steps for the next integration.
Note that since the elements of $\overline{X}(\theta)$ are complex, the integration is that of eight real coupled differential equations for the real and imaginary parts of the elements of $\overline{X}(\theta)$.

The classical procedure of assuming a semi-infinite homogeneous ionosphere at the top of the profile is followed. The solution for reflection from this semi-infinite medium is taken from reference 6 and is used as the starting solution for the integration.

This method of integration is identified as the "full-wave" procedure for obtaining ionospheric reflection coefficients. The values obtained from the integration are denoted as $\overline{X}(\theta T_i)\mid_b$, where $z=b$ is the lower cutoff height of the integration.

F. Integration of $\overline{X} = (R + 1)/C$ Through Free Space Over a Curved Earth

As was stated in Section II, C-5, the general root-finding algorithm used to locate the zeros of $F(\theta)$ requires that there be no poles in the vicinity of the zeros to be found. This requires that there be no poles in $\overline{X} = (R + 1)/C$ in this region of the $\theta$-plane. This can be assured by integrating $\overline{X}$ from the bottom of the profile upward through free-space over a curved earth to an effective "reflection height", defined by the condition

$$\sum_{\theta\text{-points}} \sum_{\text{elements}} \left| \frac{d\overline{X}(\theta T_i)}{dC} \right|^2 \rightarrow \text{Minimum}$$

where the summation is over the four elements of $\overline{X}$ and over a set of points (e.g., $\theta T_i \mid_i = 1, 2, 3, 4$) in the $\theta$-plane distributed over the region in which zeros of $F(\theta)$ are to be located. This condition is also appropriate for defining a level at which Lagrange interpolation can best be performed.
In practice the set of points \((9T_i)\) are used in defining the "effective reflection height" and these are taken to be the "given" points in the \(\theta\)-plane used for Lagrange interpolation.

As in previous waveguide formulations, e.g., reference 1, the earth curvature effect is approximated by assuming a dielectric medium which varies linearly in the square of the refractive index over a flat earth.

Considering the usual right handed cartesian coordinate system, where \(+Z\) is taken vertically upward, formulation for such an integration from the bottom of the profile at "Zb" to another height "Zt", can be obtained following reference 7. The derivation of the equations is given in Appendix B. The final equations are listed and discussed as follows.

Just below the free space-ionosphere boundary (e.g., Zb), let the total electric wave field be composed of two upward (e.g., +Z direction) traveling incident components and two downward traveling reflected components. They are defined by \(E^i_{11, y}\) and \(E^r_{11, y}\) respectively. \(E^i_{11}\) is in the X-Z plane and \(E^r_{11, y}\) is parallel to the y-axis. The total-field components just below the boundary are given in reference 7, page 118 as

\[
\begin{align*}
E_x &= (E^i_{11} - E^r_{11}) \cos \theta_i \\
E_y &= E^i_{y} + E^r_{y} \\
H_x &= (E^r_{y} - E^i_{y}) \cos \theta_i \\
H_y &= E^i_{11} + E^r_{11}
\end{align*}
\]

(32)

where \(\theta_i\) is the angle in the x-y plane which the incident wave normal makes with the vertical z axis. The following identifications are made:
\[ IR_1 = \frac{E^r_y}{E^i_y} \]
\[ IR_2 = \frac{E^r_y}{E^i_y} \]
\[ IR_4 = \frac{E^r_{11}}{E^i_{11}} \]
\[ IR_4 = \frac{E^r_{11}}{E^i_y} \]

Equations (32) and (33) are combined with the differential equations to be satisfied by the wave fields at oblique incidence (reference 7, pp. 140), to give two independent sets of relationships. Either of these sets of field variables may be set equal to zero without affecting the other, so that the corresponding waves are propagated independently.

For the first set of equations the electric field is everywhere parallel to the y-axis and the waves are said to be "horizontally polarized". For the second set of equations the electric field is parallel to the X-Z plane and the waves are said to be vertically polarized, although the electric vector is not in general vertical.

The equations become (as derived in Appendix B):

For Horizontal Polarization:

\[ (-A/C)h_1(\zeta) + (-B/C)h_2(\zeta) + E^i_y\left(\frac{IR_1 + 1}{C}\right) + E^i_{11} \left(\frac{IR_4}{C}\right) = 0 \]
\[ (-A/C) \left\{ ch_1(\zeta) + Kh_1'(\zeta) \right\} + (-B/C) \left\{ ch_2(\zeta) + Kh_2'(\zeta) \right\} + 2E_y = 0 \]  

For Vertical Polarization:

\[ (-G/C)h_1(\zeta) + (-G/C)h_2(\zeta) + E^i_{11} \left(\frac{IR_4 + 1}{C}\right) + E^i_y \left(\frac{IR_4}{C}\right) = 0 \]
\[ (-G/C) \left\{ ch_1(\zeta) + [Kh_1'(\zeta) + Lh_1(\zeta)]/n^2 \right\} + \]
\[ (-G/C) \left\{ ch_2(\zeta) + [Kh_2'(\zeta) + Lh_2(\zeta)]/n^2 \right\} + 2E^i_{11} = 0 \]
The coefficients $A$, $B$, $Q$, and $G$ are arbitrary constants to be determined. The functions $h_1(\xi)$ and $h_2(\xi)$ satisfy the differential equation

$$\frac{d^2 w}{d^2 \xi} + \xi w = 0 \quad (38)$$

and are the modified Hankel functions of order 1/3 (e.g., $h_1(\xi)$ and $h_2(\xi)$) as defined by the Computation Laboratory at Cambridge, Massachusetts (reference 8). The primes on these quantities denote derivatives with respect to the argument. Also

$$\xi = (k/a)^{2/3} [C^2 + a(Z-H)] \quad (39)$$

$$C = \cos(e)$$

$$a = 2/r_e$$

$$k = 2\pi/\lambda$$

$$K = i(\alpha/k)^{1/3}$$

$$L = i(\alpha/2K)$$

$$i = \sqrt{-1}$$

where

$H$: is the altitude where the index of refraction is unity.

$\lambda$: is the free space wave length.

$n$: is the index of refraction of the medium.

$r_e$: is the radius of the earth.

Note that the $R$ terms may be written as:

$$\overline{1x_1} = \frac{1R_1 + 1}{C}$$

$$\overline{1x_0} = \frac{1R_1 + 1}{C}$$

$$\overline{1x_0} = \frac{1R_1}{C}$$

$$\overline{1x_1} = \frac{1R_1}{C} \quad (41)$$
The "effective reflection height" is found by integrating $\bar{X}(0)$ upward to heights at, say one kilometer intervals above the bottom of the profile. That height at which the minimizing condition, Equation (31), is satisfied is the "effective reflection height". This height is chosen as the reference height, "d", used in classical waveguide theory as the height at which $R$ and $\bar{R}$ (i.e., $\bar{x}$, $n$, and $d$) are computed and as the height at which Lagrange interpolation is performed. For evaluations of $F(\theta)$, $Z_t$ is then always taken to be at $z=d$.

The upward free space integration of the ionospheric $\bar{X}(0)$ parameters is carried out in the following steps.

STEP I:

Two sets of boundary conditions (B.C.) are applied to equations (34), (35), (36) and (37) at the $Z_b$ level. These sets are

SET I: $E_{11}^i = 1, E_y^i = 0$

SET II: $E_{11}^i = 0, E_y^i = 1$

Here $\bar{X}_b(\theta)$ is the result of the Runge-Kutta integration of Equation (26) at what will be denoted as the "from level", ("f"). Where

$$\zeta_f = (k/a)^{2/3} [c^2 + a(Z_b - H)]$$

(42)

For Horizontal Polarization at the "from level" ($Z_b$), the B.C. give:

For B.C. SET I:

$$\left( \frac{A}{c} \right)_I = i \bar{X}_f \cdot \left( \frac{a^2}{\Delta \zeta} \right)$$

(43)

$$\left( \frac{B}{c} \right)_I = -i \bar{X}_f \cdot \left( \frac{a^2}{\Delta \zeta} \right)$$

(44)
For B.C. SET II:

\[
\begin{align*}
\left(\frac{A}{C}\right)_{II} &= (\bar{X}_{\perp} \cdot a_{22} - 2a_{12})/\Delta_1 \\
\left(\frac{B}{C}\right)_{II} &= (2a_{11} - \bar{X}_{\perp} \cdot a_{21})/\Delta_1
\end{align*}
\]

where

\[
\begin{align*}
a_{11} &= h_1(\zeta_f), \\
a_{12} &= h_2(\zeta_f) \\
a_{21} &= Ch_1(\zeta_f) + Kh_1(\zeta_f) \\
a_{22} &= Ch_2(\zeta_f) + Kh_2(\zeta_f) \\
\Delta_1 &= a_{11} a_{22} - a_{21} a_{12}
\end{align*}
\]

For vertical Polarization at the "from level" \((Z_b)\), the B.C. give:

For B.C. SET I:

\[
\begin{align*}
\left(\frac{Q}{C}\right)_{I} &= (\bar{X}_{\|} \cdot a_{22} - 2a_{12})/\Delta_2 \\
\left(\frac{G}{C}\right)_{I} &= (2a_{11} - \bar{X}_{\|} \cdot a_{21})/\Delta_2
\end{align*}
\]

For B.C. SET II:

\[
\begin{align*}
\left(\frac{Q}{C}\right)_{II} &= \bar{X}_{\|} \cdot \frac{a_{22}}{\Delta_2} \\
\left(\frac{G}{C}\right)_{II} &= -\bar{X}_{\|} \cdot \frac{a_{21}}{\Delta_2}
\end{align*}
\]
where:

\[
\begin{align*}
    a_{11} &= h_1(\zeta_f), \quad a_{12} = h_2(\zeta_f) \\
a_{21} &= Ch_1(\zeta_f) + \left[ Kh_1(\zeta_f) + Lh_1(\zeta_f) \right]/n_f^2 \\
a_{22} &= Ch_2(\zeta_f) + \left[ Kh_2(\zeta_f) + Lh_2(\zeta_f) \right]/n_i^2 \\
\Delta_2 &= a_{11} \cdot a_{22} - a_{21} a_{12}, \quad n_f^2 = 1 + \alpha(Z_b-H)
\end{align*}
\]

STEP II:

The values of \( E_{11}^i \) and \( E_y^i \) at the "to" level (e.g., at the height \( Z_t \)) are determined from the values of the coefficients; \( A/C, B/C, Q/C, \) and \( G/C, \) for the SET I and SET II boundary conditions.

Here the \( \zeta \) parameter at the "to" level will be denoted by:

\[
\zeta_t = (k/u)^{2/3} \left[ C^2 + \alpha(\zeta_t \cdot H) \right]
\]

For Horizontal Polarization at the \( Z_t "to" \) level the results are

\[
((E_y^i)_t)_I = \left( \frac{A}{C}_I \right) \cdot a_{21} + \left( \frac{B}{C}_I \right) \cdot a_{22}/2
\]

\[
((E_y^i)_t)_II = \left( \frac{A}{C}_II \right) \cdot a_{21} + \left( \frac{B}{C}_II \right) \cdot a_{22}/2
\]

where

\[
\begin{align*}
    a_{21} &= Ch_1(\zeta_t) + Kh_1(\zeta_t) \\
a_{22} &= Ch_2(\zeta_t) + Kh_2(\zeta_t)
\end{align*}
\]
For **Vertical Polarization** at the $Z_t$ "to" level the results are:

\[
((E_{1\text{i}}^i)_t)_{I} = ((Q/C)_I \cdot a_{21} + (G/C)_I \cdot a_{22})/2
\]

\[
((E_{1\text{i}}^i)_t)_{II} = (Q/C)_{II} \cdot a_{21} + (G/C)_{II} \cdot a_{22})/2
\]

where

\[
a_{21} = Ch_1(\xi_t) + [Kh_1(\xi_t) + Lh_1(\xi_t)]/n_t^2
\]

\[
a_{22} = Ch_2(\xi_t) + [Kh_2(\xi_t) + Lh_2(\xi_t)]/n_t^2
\]

\[
L = i(g_{\Delta k}^t), \quad n_t^2 = 1 + \alpha(Z_t-H)
\]

**STEP III:**

Knowing the values of

\[
((E_{1\text{i}}^i)_t)_{I,\text{II}} \quad \text{and} \quad ((E_{1\text{i}}^i)_t)_{I,\text{II}}
\]

at the "to" level from sets of the coefficients $A/C$, $B/C$, $Q/C$ and $G/C$ the elements of the ionospheric $\mathbf{X}^t$ (i.e., $\mathbf{R}$) can be found at the "to" level.

For **Horizontal Polarization** at the "to" level the results are:

\[
(\mathbf{X}_{1\text{i}t}) = [V_1((E_{x\text{i}}^i)_t)_{\text{II}} - V_2((E_{x\text{i}}^i)_t)_{I}]/W_{1}
\]

\[
(\mathbf{X}_{1\text{i}t}) = [V_2((E_{y\text{i}}^i)_t)_{I} - V_1((E_{y\text{i}}^i)_t)_{\text{II}}]/W_{1}
\]

where:

\[
W_{1} = ((E_{1\text{i}}^i)_{I} \cdot ((E_{1\text{i}}^i)_{\text{II}} - ((E_{1\text{i}}^i)_{\text{II}} \cdot ((E_{y\text{i}}^i)_{I})
\]

\[
V_{1} = (\frac{A}{C})_{I} h_{1}(\xi_t) + (\frac{B}{C})_{I} h_{2}(\xi_t)
\]

\[
V_{2} = (\frac{A}{C})_{II} h_{1}(\xi_t) + (\frac{B}{C})_{II} h_{2}(\xi_t)
\]
For Vertical Polarization at the "to" level the results are:

\[
\begin{align*}
(\mathbf{\Pi})_t &= [V_1((E^i_I)_t)_{II} - V_2((E^i_I)_t)_I]/W_2 \\
(\mathbf{\Pi})_t &= [V_2((E^i_I)_t)_I - V_1((E^i_I)_t)_{II}]/W_2
\end{align*}
\]

(63) (64)

where

\[
W_2 = ((E^i_I)_t)_I \cdot ((E^i_I)_t)_{II} - ((E^i_I)_t)_{II} \cdot ((E^i_I)_t)_I
\]

\[
V_1 = (Q/C)_I h_1(c_t) + (G/C)_I h_2(c_t)
\]

\[
V_2 = (Q/C)_{II} h_1(c_t) + (G/C)_{II} h_2(c_t)
\]

(65)

Thus having the values of the ionospheric reflection coefficients \( \Xi(\theta) \) at \( Z_b \) the resulting values of these terms at the height \( Z_t = d \) are obtained by applying the minimizing condition, e.g., so that the function \( F(\theta) \) may be computed in a form where poles are absent.

The values of the terms \( \Pi R_{\parallel}, \_R_{\perp}, \_R_{\perp} \) and \( \_R_{\parallel} \) at height \( Z \) may be obtained from the results of the above equation in terms of \( \Xi(\theta) \) by the relations of Equation (41).

G. The Ground Reflection Coefficient Matrices

The ground reflection coefficients (e.g., \( \_R_{\parallel} \) and \( \_R_{\perp} \)) are needed in the computation of the \( F(\theta) \) equation. The particular form of these coefficients, as used in the MODESRCH procedure, is identified by the ratios given in Appendix A by the equations A-9 and A-10:
\[
\begin{align*}
\ln_1 \overline{\ln_1} + 1 &= \overline{\ln_1} C \quad (66) \\
\ln_1 \overline{\ln_1} + 1 &= \overline{\ln_1} C \quad (67)
\end{align*}
\]

The equations needed to determine the values of \( \overline{\ln_1}, \overline{\ln_1} \) (in terms of \( \ln_1, \ln_1, \ln_d, \) and \( \ln_d \)) at a reference height "d" may be obtained from equations of Section F.

Needed substitutions to those equations are:

\[
Z = 0 \text{ for } Z = Z_b
\]  \quad (68)

\[
\left(\ln_{\perp}\right)^f = \left(\ln_{\perp}\right)^f = 0
\]

\[
\frac{\overline{\ln_{\perp}} + 1}{C \overline{\ln_{\perp}}} = \left(\ln_{\perp}\right) \bigg|_{Z=0} \quad (69)
\]

\[
\frac{\overline{\ln_{\perp}} + 1}{C \overline{\ln_{\perp}}} = \left(\ln_{\perp}\right) \bigg|_{Z=0} \quad (70)
\]

Where the Fresnel reflection coefficients for the ground-free space interface are

\[
\overline{\ln_{\perp}} = \frac{C n_g^2 - \sqrt{n_g^2 - S^2}}{C n_g^2 + \sqrt{n_g^2 - S^2}} \quad (71)
\]

and

\[
\overline{\ln_{\perp}} = \frac{C - \sqrt{n_g^2 - S^2}}{C + \sqrt{n_g^2 - S^2}} \quad (72)
\]

with

\[
n_g^2 = \left(\varepsilon - j \sigma/\omega\right)/\varepsilon_0 \quad (73)
\]
and

\[ \varepsilon = \text{dielectric constant of the ground.} \]
\[ \sigma = \text{conductivity of the ground.} \]
\[ \omega = \text{angular wave frequency.} \]
\[ C = \cos \theta, S = \sin \theta. \]

Well behaved forms of \( \frac{\varepsilon_{\|}}{\|_{\|}} \bigg|_{z=0} \) and \( \frac{\varepsilon_{\perp}}{\|_{\perp}} \bigg|_{z=0} \) at the "from" level (e.g. \( z = 0 \)), are derived in appendix C, and are given as:

\[
\begin{align*}
1^n_{\|} &= 1, \\
1^d_{\|} &= \frac{1}{2} \left[ C - \left( \frac{\sqrt{n_g^2 - s^2}}{n_g^2} \right) \right], \\
1^n_{\perp} &= \frac{1}{\sqrt{n_g^2 - s^2}}, \\
1^d_{\perp} &= \frac{1}{2} \left[ \left( C/\sqrt{n_g^2 - s^2} - 1 \right) \right].
\end{align*}
\]

At the "from" level (e.g., \( z = 0 \)), Equations (45) and (46) become:

For Horizontal Polarization:

\[
\begin{align*}
(A/C)_{II} &= (1^n_{\perp} \cdot a_{22} - 21^d_{\perp} a_{12})/(\Delta_1 \cdot 1^d_{\perp}) \quad (75) \\
(B/C)_{II} &= (z \cdot 1^d_{\perp} \cdot a_{11} - 1^n_{\perp} \cdot a_{21})/(\Delta_1 \cdot 1^d_{\perp}) \quad (76)
\end{align*}
\]

where:

\[
\begin{align*}
a_{11} &= h_1(\varepsilon_0), \quad a_{12} = h_2(\varepsilon_0) \\
a_{21} &= C h_1(\varepsilon_0) + K h_1(\varepsilon_0) \quad \varepsilon_0 = (\kappa/\alpha)^{2/3} \left[ C^2 - \sigma H \right] \\
a_{22} &= C h_2(\varepsilon_0) + K h_2(\varepsilon_0) \quad \alpha = 2/r_e \\
\Delta_1 &= a_{11} \cdot a_{22} - a_{21} \cdot a_{12}
\end{align*}
\]
For Vertical Polarization:

Equations (48) and (49) become:

\[
(Q/C)_I = (n_d \cdot a_{22} - 2 \cdot n_r \cdot a_{12})/\Delta 2 \cdot n_r \cdot a_{21})/(\Delta 2 \cdot n_r) 
\]  

(78)

\[
(G/C)_I = (2 \cdot n_r \cdot a_{11} - n_d \cdot a_{21})/(\Delta 2 \cdot n_r) 
\]  

(79)

where:

\[
a_{11} = h_1(\xi_0), \quad a_{12} = h_2(\xi_0)
\]

\[
a_{21} = Ch_1(\xi_0) + ((Kh_1(\xi_0) + Lh_1(\xi_0))/n_0^2
\]

\[
a_{22} = Ch_2(\xi_0) + ((Kh_2(\xi_0) + Lh_2(\xi_0))/n_0^2
\]

\[
\Delta 2 = a_{11} - a_{21} \quad a_{12}, \quad n_0^2 = 1 - \alpha H
\]

At the "to" level (e.g., z=\(d\)), Equation (55) becomes, for Horizontal Polarization:

\[
((E_y)_d)_{II} = ((A/C)_{II} \cdot a_{21} + (B/C)_{II} \cdot a_{22})/2
\]  

(81)

where

\[
a_{21} = Ch_1(\xi_d) + Kh_1(\xi_d)
\]

\[
\xi_d = (k/\alpha)^{2/3} \left[ c^2 + \alpha(d - H) \right]
\]

(82)

\[
a_{22} = Ch_2(\xi_d) + Kh_2(\xi_d)
\]

and Equation (57) becomes, for Vertical Polarization:

\[
((E_y)_{II}) = (Q/C)_{II} \cdot a_{21} + (G/C)_{II} \cdot a_{22})/2
\]  

(83)

where

\[
a_{21} = Ch_1(\xi_d) + \left[ Kh_1(\xi_d) + h_1(\xi_d) \right]/n_d^2
\]

(84)

\[
a_{22} = Ch_2(\xi_d) + \left[ Kh_2(\xi_d) + Lh_2(\xi_d) \right]/n_d^2
\]

\[
\alpha_d^2 = 1 + \alpha(d - H)
\]
At the "to" level (e.g. \( Z = d \)) define:

\[
\frac{\partial N_1}{\partial d} \bigg|_{Z=d} = \frac{\partial N_1}{\partial D_1}
\]

and

\[
\frac{\partial N_1}{\partial D_1} \bigg|_{Z=d} = \frac{\partial N_1}{\partial D_1}
\]

Equations (61) and (62) for Horizontal Polarization give:

\[
J_{N_1} = (A/C)_{\|} h_1(\xi_d) + (B/C)_{\|} h_2(\xi_d)
\]

(87)

\[
J_{D_1} = ((E_y)'d)_{\|}
\]

(88)

and Equations (64) and (65) for Vertical Polarization give:

\[
J_{N_1} = (Q/C)_{\perp} h_1(\xi_d) + (G/C)_{\perp} h_2(\xi_d)
\]

(89)

\[
J_{D_1} = ((E_z)'d)_{\perp}
\]

(90)

The results of Equations (87) - (90) can be rearranged by reference to Equation (10)

where

\[
\frac{\partial N_1}{\partial D_1} = \left( \frac{1}{\frac{1}{R_1} + \frac{1}{R_1}} \right)
\]

and

\[
\frac{\partial N_1}{\partial D_1} = \left( \frac{1}{\frac{1}{R_1} + \frac{1}{R_1}} \right)
\]

(91)

(92)

\[
\frac{1}{C(\frac{\partial N_1}{\partial D_1}) - 1} = \frac{J_{D_1}}{C_1N_1 - \frac{\partial D_1}{\partial D_1}}
\]

(91)

\[
\frac{1}{C(\frac{\partial N_1}{\partial D_1}) - 1} = \frac{J_{D_1}}{C_1N_1 - \frac{\partial D_1}{\partial D_1}}
\]
Inspection of the equations will reveal that $1_{D_1}$, $1_{D_1}$, $1_{N_1}$, and $1_{D_1}$ contain no poles for any finite value of $\theta$.

The resulting equations of this section are equivalent to those given in reference 9.

H. Excitation Factors, Height Gains and Auxiliary Functions

(1) Excitation Factors

The modal excitation factor and the modal height gain functions are two parameters needed in computing electric field strengths. The excitation factor formulas, as presented in reference 10, are summarized in the table 1. The column headings only apply to excitation of the electric field components $E_z$, $E_y$ and $E_x$ and the row headings apply to excitation by a vertical dipole ($\lambda_V$), horizontal dipole end on ($\lambda_E$) and a horizontal dipole broadside ($\lambda_B$). The direction of $z$ is taken positive into the ionosphere. Positive $x$ is the direction of propagation and $y$ is normal to the plane of propagation.
### Table 1 - EXCITATION FACTORS

<table>
<thead>
<tr>
<th>Field Component</th>
<th>$E_z$</th>
<th>$E_y$</th>
<th>$E_x$</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Exciter</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$\lambda_V$</td>
<td>$B_1 \frac{(1+i_{R_{II}})^2(1-i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
<td>$-B_1 \frac{R_{II}(1+i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
<td>$B_1 \frac{(1+i_{R_{II}})^2(1-i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
</tr>
<tr>
<td>$\lambda_E$</td>
<td>$B_2 \frac{(1+i_{R_{II}})^2(1-i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
<td>$-B_2 \frac{R_{II}(1+i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
<td>$B_2 \frac{(1+i_{R_{II}})^2(1-i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
</tr>
<tr>
<td>$\lambda_B$</td>
<td>$B_2 \frac{R_{II}(1+i_{R_{II}})(1+i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
<td>$-B_2 \frac{(1+i_{R_{II}})^2(1-i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
<td>$B_2 \frac{R_{II}(1+i_{R_{II}})(1+i_{R_{II}})}{S_{II_{R_{II}}} D_{II}}$</td>
</tr>
</tbody>
</table>

The $R$ and $\bar{R}$'s are the elements of the reflection matrix looking into the ionosphere and towards the ground from the reference level "d" within the guide.

The reference height "d" is that value where solutions to $F(\theta) = 0$ are determined.

$B_1$ and $B_2$ are given by:

$$B_1 = S^{5/2} \left. \frac{\partial F}{\partial \theta} \right|_{\theta = \theta_n}$$  \hspace{1cm} (93)  

$$B_2 = -\frac{B_1}{S}$$  \hspace{1cm} (94)
where \( S \) is the sine of the eigenangle and the denominator is the derivative of the modal equation (i.e. eq. 7) at an eigenangle, \( \theta \).

The definitions of the \( D_{ij} \)'s are given in the next section.

To compute electric field strengths the excitation factors must be supplemented with definitions of the height gains.

(2) Height Gain Terms

The height gain relationships at height \( Z \) are described below.

From Equation (39):

\[
\zeta = \left( \frac{k}{\alpha} \right)^{2/3} \left[ c^2 + \alpha (Z - H) \right] \tag{95}
\]

At any height \( Z \).

The height gain for the vertical electric field \( (E_z) \) is \( f_{||}(z) \) and is given by:

\[
f_{||}(Z) = \left[ (Q/C)_I h_1(\zeta) + (G/C)_I h_2(\zeta) \right] \exp[(Z - d)/r_e] \tag{96}
\]

where \( d \) is the reference height for solving \( \Phi(\theta) = 0 \) for the eigenangle and where \( (Q/C)_I \) and \( (G/C)_I \) are given by Equations (48) and (49) respectively. The term "\( r_e \)" is the radius of the earth.

Also, the height gain for the horizontal electric field component \( (E_y) \) normal to the plane of propagation at height \( Z \) is given by:

\[
f_{\perp}(Z) = \left[ (A/C)_{\perp I} h_1(\zeta) + (B/C)_{\perp I} h_2(\zeta) \right] \tag{97}
\]

where \( (A/C)_{\perp I} \) and \( (B/C)_{\perp I} \) are given by Equations (45) and (46) respectively.

The height gain for the horizontal electric field components \( (E_x) \) in the plane of propagation at height \( Z \) is given by:

\[
g(Z) = \frac{1}{\imath k} \frac{d}{dZ} \left[ f_{||}(Z) \right] = -i \exp \left( \frac{Z - d}{a} \right) \left[ \left( \frac{2}{ak} \right)^{1/3} \left[ (Q/C)_I h_1'(\zeta) + (G/C)_I h_2'(\zeta) \right] + \left( \frac{2}{ak} \right) \left[ (Q/C)_I h_1(\zeta) + (G/C)_I h_2(\zeta) \right] \right] \tag{98}
\]
At height \( Z = d \) (i.e. the reference height) the argument is:

\[
\zeta_d = \left(\frac{k}{\alpha}\right)^{2/3} [c^2 + \alpha(d - H)]
\]

and

\[
D_{11} = f_1^2(\zeta_d), \quad D_{12} = f_1(\zeta_d) f_\perp(\zeta_d)
\]

\[
D_{22} = f_\perp^2(\zeta_d)
\]

(3) **Modified Excitation Terms**

Various combinations of the modal excitation factor and the height gain function are required for computing field strengths by different mode-sum programs. Reference 11 is an example.

Table 2 illustrates the case where the excitation factors as described in Table 1 and the height gains from Equations (96), (97) and (98) are lumped together and defined as modified excitation factors. Table 2 illustrates combinations obtained for computing the vertical electric field \( E_z \). These particular combinations will be discussed further in Section V-C-2 for the option NPUNCH = 7.
Vertical source:

\[ X_{VZ} = B_1 \frac{(1 + \overline{R}_1)^2 (1 - \overline{R}_1 \overline{R}_1)}{\overline{R}_1 \cdot D_{11}} \cdot f_1(Z_t) \cdot f_1(Z_r) \]

Horizontal broadside source:

\[ X_{BZ} = \frac{\overline{R}_1 (1 + \overline{R}_1) (1 + \overline{R}_1)}{D_{12}} \cdot f_1(Z_t) \cdot f_1(Z_r) \]

Horizontal end-on source:

\[ X_{EZ} = \frac{(1 + \overline{R}_1)^2}{\overline{R}_1 \cdot D_{11}} \cdot (1 - \overline{R}_1 \overline{R}_1) \cdot g(Z_t) \cdot f_1(Z_r) \]

\( Z_t \) is height of the transmitter and 
\( Z_r \) is the height of the receiver

The excitation factor usually considered is the vertical component of the electric field vector for vertical receiving and transmitting point dipoles, where both receiver and transmitter are at the ground (e.g., \( z = 0 \)). The excitation factor \( X_{VZ} \) is denoted in this case by \( X_0 \). Wait's excitation factor is computed as \( A_0 = -0.5i k h_r X_0 \), where \( k \) is the wave number in \( \text{km}^{-1} \), \( h_r \) is \text{REFLHT} and \( i = \sqrt{-1} \).

(4) **Fast Mode Conversion**

Reference 12 presents equations and a computer program for calculating the vertical electric field strength at a receiver by the procedure of mode conversion. An input parameter required for these calculations...
is denoted as 'FOVR'. The mode conversion program requires that the reference height used for computing ionospheric and ground reflection coefficients be at the ground (i.e., \(z=0\)).

The reference height used in the MODESRCH computer program is at some height "\(d\)" (\(d\neq 0\)). Therefore the equation needed to obtain the required 'FOVR' term is given by

\[
\text{FOVR} = \frac{(1 + \frac{\text{IR}}{\text{RI}})(1 - \frac{\text{IR}}{\text{RI}})}{(1 + \frac{\text{IR}}{\text{RI}})(\frac{\text{IR}}{\text{RI}})} \cdot \frac{f(d)}{f_1(d)}
\]

or

\[
\frac{(1 + \frac{\text{IR}}{\text{RI}})(\frac{\text{IR}}{\text{RI}})}{(1 + \frac{\text{IR}}{\text{RI}})(1 - \frac{\text{IR}}{\text{RI}})} \cdot \frac{f(d)}{f_1(d)}
\]

(101)

I. Use of Lagrange Interpolation

The general root-finding algorithm, (section III), used to locate the zeros of a function, \(f(\theta)\), requires that values of \(f(\theta)\) and its derivatives wrt \(\theta\) be available upon call for any value of \(\theta\) in the region of interest. In Section F a criterion was given for choosing a reference height, '\(d\)', so as to minimize the variation of \(\frac{\lambda}{\sqrt{2}} = (\frac{\lambda}{\sqrt{2}} + 1)/C\) wrt \(\theta\) thereby placing the dominant waveguide effects into \(n\) and \(d\) (i.e., into \(R\)). (See Equation 7.) On the other hand, far more computation is required to evaluate \(\frac{\lambda}{\sqrt{2}}\) than to evaluate \(n\) and \(d\). This suggests the utilization of an approximate representation of \(\frac{\lambda}{\sqrt{2}}\) to be used within the general root-finder, and then following the zeros of \(f(\theta)\) (see Equation 7) to their locations which will be consistent with the exact formulation for \(\frac{\lambda}{\sqrt{2}}\).

The general root-finder requires that the function \(f(\theta)\) be analytic with no poles and therefore an approximate representation of \(\frac{\lambda}{\sqrt{2}}\) must be analytic with no poles. A polynomial representation has been chosen, which can be represented by the Lagrange interpolation formula
\[ \bar{X}(\theta)_{\text{approx.}} = \sum_{i=1}^{n} \left( \prod_{\substack{k=1 \atop k \neq i}}^{n} (\theta - \theta_k) \right) p_i \quad \text{and} \quad p_i = \frac{\bar{X}(\theta_i)}{\prod_{\substack{k=1 \atop k \neq i}}^{n} (X_i - X_k)} \tag{102} \]

where \( \bar{X}(\theta_i) \) are evaluations of \( \bar{X} \) using the exact formulation for \( \bar{X} \) at \( n \) selected angles, \( \theta_i \), distributed over the \( \theta \)-plane in the region of interest.

Since \( F(\theta) \) (see Equation 7) in both approximate and exact (complete) forms is analytic with no poles, each zero of \( F(\theta)_{\text{approx.}} \) may be followed uniquely to a corresponding zero of \( F(\theta)_{\text{exact}} \). The continuous transition from one to the other may be represented, for example, by

\[ F(\theta) = F(\theta) \cdot \cos^2 \beta + F(\theta) \cdot \sin^2 \beta \tag{103} \]

with \( \beta \) varying continuously from 0 to 90°. In practice \( \beta \) is incremented in finite steps with a Newton-Raphson iteration at each step. How far the zero of \( F(\theta) \) "travels" in the \( \theta \)-plane as \( \beta \) is varied from 0 to 90° depends; however, on how well \( F(\theta) \) (i.e., \( \bar{X}(\theta) \)) is represented by the Lagrange interpolation. It is possible, in principle, that a zero may travel from the region of interest to infinity, or vice-versa. There are no guarantees and although, in most cases in the VLF and LF range, the use of Lagrange interpolation may save on computation by an order of magnitude, it nevertheless represents the "weakest link" in formulating a self-contained mode search. In some cases, especially in the ELF range, it is necessary to use the (exact) formulation for \( F \).

J. Comments of the Differentiation of \( F(\theta) \) wrt \( \theta \)

The excitation factor components given in Section H require the computation of the derivative \( F(\theta) \) wrt \( \theta \) for each eigenvalue of \( \theta \). In
addition, the general root-finder, (section III), requires, in its later stages, some evaluations of \( dF/d\theta \), and the following of zeros by the use of Newton-Raphson iteration described in Section I requires computation of \( dF/d\theta \). Traditionally this derivative has been evaluated by computing \( F \) at two closely-spaced values of \( \theta \), using a large number of significant digits in the computation. An alternative is to analytically differentiate \( F \), which implies analytic differentiation of \( f \), \( n \) and \( d \). This alternative was chosen for the mode search program.

Note that differentiation of the numerical (Runge-Kutta) integration described in Section E is performed by the same type numerical integration, with the integrand analytically differentiated wrt \( \theta \).

In certain sections of MODESRCH where \( F(\theta) \) is computed in terms of Lagrange interpolation, the formulation of the derivative wrt \( \theta \) is given by

\[
\frac{dF(\theta)}{d\theta} = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \left\{ \frac{1}{n} \prod_{k=1}^{n} \left( F(\theta) - F_k(\theta) \right) \right\} \cdot \left\{ P_i + P_j \right\}
\]  

where

\[
P_k = F(\theta_k) \quad \frac{n}{\prod_{k=1}^{n} (X_k - X_{k+1})},
\]

and \( n \) is the number of points.
III. **THE GENERAL ROOT-FINDER**

There is a need to formulate a procedure to determine the complex zeros of a complex function. For the method presented here, the search for such zeros is based on the fact that the lines of constant phase of any complex function, \( F(\theta) \), may be discontinuous only at points where \( F(\theta) = 0 \) or \( F(\theta) \to \infty \). The function \( F(\theta) \), to be discussed, is such that it contains no poles and only \( F(\theta) = 0 \) is considered. Also, the lines of constant phase around \( \phi(\theta) = 0 \) progress only in a counter-clockwise direction.

Let:

\[
F(\theta) = F_r(\theta_r, \theta_i) + j F_i(\theta_r, \theta_i)
\]

where

\[
\theta = \theta_r + j \theta_i
\]

Also

\[
F(\theta) = \left[ F_r(\theta_r, \theta_i)^2 + F_i(\theta_r, \theta_i)^2 \right]^{1/2} e^{j\phi}
\]

where

\[
\phi = \tan^{-1}\left( \frac{F_i(\theta_r, \theta_i)}{F_r(\theta_r, \theta_i)} \right)
\]

and

- \( F_r(\theta) \) is the real part of the complex function \( F(\theta) \)
- \( F_i(\theta) \) is the imaginary part of the complex function \( F(\theta) \)
- \( \theta_r \) is the real part of the complex number \( \theta \)
- \( \theta_i \) is the imaginary part of the complex number \( \theta \).

If \( F(\theta) \) has no poles, this implies that the line of any particular constant phase value \( \phi = \phi_c \), radiating from a zero of \( F(\theta) \), must cross a closed contour containing that zero at least once. Furthermore, no other zero of \( F(\theta) \) may be on this phase line. These relations are illustrated in Figure 2.
Figure 2. Constant Phase Lines of $F(\theta)$.

A line of constant phase (e.g., $\phi = \phi_c$) which crosses the contour may be followed inward until it leads to a zero of $F(\theta)$ or until the line again reaches the contour.

The procedure is as follows.

(A) Let $\phi_c = 0^\circ$ (or $180^\circ$), this implies that

$$F_I(\theta_r, \theta_i) = 0 \quad (109)$$

Also let $\phi_c = 90^\circ$ (or $270^\circ$), this implies that

$$F_R(\theta_r, \theta_i) = 0 \quad (110)$$

(B) Construct a "contour rectangle" in the complex $\theta$ plane. Denote the corners of this rectangle by:

TLEFT - value of real part of Theta at left edge of rectangle.
TRIGHT - value of real part of Theta at right edge of rectangle.
TBOT - value of imag. part of Theta at bottom edge of rectangle.
TTOP - value of imag. part of Theta at top edge of rectangle.
Define the following parameters:

- **TMESH** - equal to about half the average spacing between zeros within the rectangle.
- **TOL** - tolerance to which zeros are to be found.

If two zeros are closer than 'TOL', the zeros will not be found by this method.

(C) Divide the 'contour rectangle' up into small mesh squares where the dimensions of the squares is such that the length of a side is smaller than the average separation of the zeros of $F(\theta)$ (e.g., size of a mesh side = 'TMESH').

(D) Normalize the value of the corners of the "contour rectangle" into mesh units. The corners are then denoted as JLT, JRT, JTOP, and JBOT. That is $\theta$'s for the "contour rectangle" are:

- $\theta_1$ at left edge: $JLT = \text{TLEFT}/\text{TMESH}$
- $\theta_2$ at right edge: $JRT = \text{TRIGHT}/\text{TMESH}$
- $\theta_3$ at top edge: $JTOP = \text{TTOP}/\text{TMESH}$
- $\theta_4$ at bottom edge: $JBOT = \text{TBOT}/\text{TMESH}$.

(L) Figure 3 illustrates the rectangular region of the complex $\theta$-plane within which the roots of $F(\theta) = 0$ are to be found.
Figure 3. "Contour Rectangle" with "MESH" Squares.

In the above example all the axes are marked with tic marks, with spacing $\Delta \theta = \text{TMESH}$. The mesh size within the rectangle is equal to TMESH.

In the above figure, JLT = 6, JRT = 20, JTOP = 1, JBOT = -3. Since some "slop" occurs at the rectangle boundary, the limits of the rectangle are made one TMESH larger on all sides than actually desired.

TMESH should be no larger than about one-half the average spacing between roots.

(F) The procedure then is to examine the character of $F(\theta)$ in a counter clockwise direction around the contour rectangle (see Figure 4). Intermediate steps of investigating mesh squares are also carried out in a counter clockwise direction. Again see Figure 4.
(G) Consider an individual mesh square (e.g., mesh no. j). The corners of the square may be labeled in terms of $F(\theta_r, \theta_i)$ where each edge is one-mesh unit in length. Figure 5 illustrates normalized coordinates of the mesh.

Figure 4. Directions for Determining Zeros of $F(\theta)$.

Figure 5. Normalized Coordinates of Each Mesh Square for $F(\theta_r, \theta_i)$. 

41
Note that for each unit mesh square the lower left hand corner is taken as the reference position.

A basic assumption put on \( F(\theta) \) is that \( F_I(\theta) \) and \( F_R(\theta) \) are to be linear along each edge of a mesh square.

The coordinates of the corners of mesh square No. 1 (see Figure 4) are given by:

\[
\begin{align*}
F(0,1) &= F(JLT,JTOP) \\
F(0,0) &= F(JLT,JTOP-1) \\
F(1,0) &= F(JLT+1,JTOP-1) \\
F(1,1) &= F(JLT+1,JTOP)
\end{align*}
\]

(11)

All other squares are identified by increasing or decreasing the \( \theta_p \) and/or \( \theta_i \) coordinates by increments of 1-mesh unit.

(H) The root-finder procedure continues as follows. First, mesh square No. 1 is examined. The values of \( F_I(0,1) \) and \( F_I(0,0) \) for mesh edge No. 1 are computed. If the signs of these two values are identical, there can be no line with constant value of \( F_I(\theta) = 0 \) passing through this edge.

Next, investigate mesh square No. 2 (see Figure 4). Compute the values of \( F_I(0,1) \) and \( F_I(0,0) \) for this mesh square. The value of \( F_I(0,1) \) for mesh square No. 2 is identical to that of \( F_I(0,0) \) for mesh square No. 1, however, the value of \( F_I(0,0) \) for mesh square No. 2 will be a new value.

The coordinates of mesh square No. 2 are identified by:

\[
\begin{align*}
F(0,i) &= F(JLT,JTOP-1) \\
F(0,0) &= F(JLT,JTOP-2) \\
F(1,0) &= F(JLT+1,JTOP-2) \\
F(1,1) &= F(JLT+1,JTOP-1)
\end{align*}
\]

(112)

If the signs of \( F_I(0,1) \) and \( F_I(0,0) \) for mesh square No. 2 are opposite, this implies that the line \( F_I(\theta) = 0 \) enters mesh square No. 2 somewhere along edge No. 1. Because of the linearity assumption of \( F_I(\theta) \) along
the edge of a mesh square, the intersection between the edge and the constant phase line $F_1(\theta) = 0$ is (as shown in Appendix D, eq. 13) to be at:

$$\theta_r = \theta_j = \left(\frac{-F_1(0,0)}{F_1(0,1)-F_1(0,0)}\right)$$

(113)

$\theta_r$ is identified as ENTER-R

$\theta_j$ is identified as ENTER-I

(I) It has now been determined that the constant line $F_1(\theta) = 0$ has entered mesh square No. 2. The problem is to find where this line exits from the square. Also it must be determined whether a zero to $F(\theta)$ lies within the mesh square. Several tests must be made:

1) Computations are made to obtain the values of $F_1(1,0)$ and $F_1(1,1)$ for mesh square No. 2. At this point values are known for $F_1(\theta)$ at all four corners of mesh square No. 2.

2) Test for there being two (hyperbolic) lines entering and leaving the mesh square along each line of which $F_1(\theta) = 0$.
   (a) If only one line, the parameter LTWO is set to zero.
   (b) If the values of $F_1(0,0)$ and $F_1(1,1)$ are each greater than zero; and the values of $F_1(1,0)$ and $F_1(0,1)$ are each less than zero then there are two hyperbolic lines of $F_1(\theta) = 0$ entering and leaving the mesh square. This condition is illustrated in Figure 6.

![Figure 6. Hyperbolic Branch Curves of $F_1(\theta)$ Within a Mesh Square.](image-url)
For this case, the parameter "LTWO" is set equal to 1.

(c) Also, if the values of $F_I(0,0)$ and $F_I(1,1)$ are each less than zero, and the values of $F_I(1,0)$ and $F_I(0,1)$ are each greater than zero, the result is identical to (b) above.

3) Test for there being at least one hyperbolic branch line entering and leaving the mesh square along which $F_R(\theta) = 0$.

(a) If the values of $F_R(0,0)$, $F_R(0,1)$, $F_R(1,0)$ and $F_R(1,1)$ are all of the same sign, then there is no line of $F_R(\theta) = 0$ passing through the mesh square and the variable "I90" is set to zero.

(b) If the condition of (a) above, is not met, then there is at least one branch line entering and leaving the mesh and the variable 'I90' is set equal to 1.

4) Test to see if there exists the conditions for the presence of hyperbolic lines at both $F_I(\theta) = 0$ and $F_R(\theta) = 0$ within the mesh square.

(a) If the condition is not present (e.g., "LTWO" is equal to 0 and "I90" is equal to 0) then determine which mesh edge (e.g., 2, 3, or 4) where the original entry line exits and proceed with an analysis of the appropriate mesh square. For example, as can be observed from Figure 4, edge No. 2 leads to mesh square No. 3, edge No. 3 leads to mesh square No. 5 and edge No. 4 leads to mesh square No. 1.

The line $F_I(\theta) = 0$ is followed continuously from mesh to mesh until the line exits from the "contour rectangle". When this occurs, the value of the leading edge (counter clockwise direction) of the mesh square,
through which the $F_I(\theta) = 0$ line exits, is noted. By identifying the mesh square, in which this line exit occurs, it is possible to later avoid re-entering this mesh square through this exit edge while following along the outer edge of the "contour rectangle". Figure 7 illustrates this situation. Note that it is possible that a solution to $F(\theta) = 0$ will be found as the line $F_I(\theta) = 0$ progresses through the various mesh squares. This case will be discussed in the following Section (b).

The next step after exit from mesh square No. $k$ is to examine mesh square No. $3$ for intersections of the line $F_I(\theta) = 0$ on edge No. 1 of mesh square No. $3$.

(b) If hyperbolic lines of both $F_I(\theta) = 0$ and $F_R(\theta) = 0$ do exist within mesh square No. $2$, then the following hyperbolic relations are examined.

Figure 7. Contour Rectangle, Mesh Squares and the line $F_I(\theta) = 0$.  
45
It is shown in Appendix D that the relationship for $F(\theta)$ based on a knowledge of $F(\theta)$ at each of the corners of a mesh square is given by

$$F(\theta) = a + b\theta_i + c\theta_r + d\theta_r\theta_i$$  \hspace{1cm} (114)

where

$$a = F(0,0)$$
$$b = F(0,1) - F(0,0)$$
$$c = F(1,0) - F(0,0)$$
$$d = F(0,0) + F(1,1) - F(0,1) - F(1,0)$$  \hspace{1cm} (115)

These coefficients describe the variation of $F(\theta)$ within the mesh square given the values of $F(\theta)$ at the corners of the square and that a linear variation exists between values of $F(\theta)$ along the edges of the square.

When $F_R(\theta) = 0$ (or $F_I(\theta) = 0$), Equation 114 is the equation of an equilateral hyperbola with vertical and horizontal asymptotes.

Now consider the constant phase line $F_I(\theta) = 0$. From Equation 114 this is

$$(a_i + b_i \theta_i) + (c_i + d_i \theta_i)\theta_r = 0$$

or

$$\theta_r = -\frac{a_i + b_i \theta_i}{c_i + d_i \theta_i}$$  \hspace{1cm} (116)

The following equation is also obtained from Equation 114.

$$(a_i + c_i \theta_r) + (b_i + d_i \theta_r)\theta_i = 0$$

or

$$\theta_i = -\frac{a_i + c_i \theta_r}{b_i + d_i \theta_r}$$  \hspace{1cm} (117)

The position of the crossings of the hyperbolic asymptotes (which are parallel to the sides of the square) for the lines $F_I(\theta) = 0$ are computed since both lines (hyperbolic branches) may enter and leave the square.
For the function $F_I(\theta) = 0$ the line of the vertical asymptote is given in Appendix D as

$$\theta_r = -\frac{b_i}{d_i}$$

(118)

where $\theta_r$ is identified as CENTER-R, while the line of the horizontal asymptote is given by

$$\theta_i = -\frac{c_i}{d_i}$$

(119)

where $\theta_i$ is identified as CENTER-I.

There are two possible points at which there may be crossings of the (hyperbolic) lines $F_R(\theta) = 0$ and $F_I(\theta) = 0$. A crossing point is chosen to be a zero of the function if it lies within the current mesh square and if it lies on the hyperbolic branch of $F_I(\theta) = 0$ currently being followed. Note that in the example being examined the hyperbolic branch line being examined enters mesh square No. 2 along edge No. 1 and exits along edge No. 2.

Both Equation (116) and Equation (117) must be examined since in any given case either form (but not both forms) may be indeterminate.

Equation (116) may be considered as two equations, that is for real and imaginary parts of the constants $a, b, c,$ and $d$:

$$\frac{(a_r+b_r \theta_i)}{(c_r+d_r \theta_i)} = \frac{a_i+b_i \theta_i}{c_i+d_i \theta_i}$$

(120)

and, equating the two right hand equations the result is

$$P_I \theta_i^2 + Q_I \theta_i + W_I = 0$$

(121)

where

$$P_I = b_r d_i - b_i d_r$$

$$Q_I = a_r d_i + b_r c_i - a_i d_r - b_i c_r$$

$$W_I = a_r c_i - a_i c_r$$

(122)
This equation is solved (see Appendix E) for $\theta_i$. Then, knowing $\theta_i$, $\theta_r$ can be determined from Equation (116). In the same manner Equation (117) gives

$$\theta_i = \frac{a_r + c_r \theta_r}{b_r + d_r \theta_r} = \frac{a_i + c_i \theta_r}{b_i + d_i \theta_r}$$

(123)

and equating the two right hand equations gives

$$p_2 \theta_r^2 + q_2 \theta_r + w_2 = 0$$

(124)

where

$$p_2 = c_r d_i - c_i d_r$$

$$q_2 = a_r d_i + c_r b_i - a_i d_r - c_i b_r$$

$$w_2 = a_r c_i - a_i c_r$$

(125)

this equation may be solved for $\theta_r$ (see Appendix E). Then, knowing $\theta_r$, $\theta_i$ can be determined from Equation (117).

If

$$|b_r d_i - b_i d_r| < |c_r d_i - c_i d_r|$$

(126)

Equations (116) and (121) are solved for $\theta_r$ and $\theta_i$. If the inequality of Equation (126) is not true then Equations (117) and (124) are solved for $\theta_r$ and $\theta_i$.

A solution of Equations (116) and (121) (or Equations (117) and (124)) is accepted if it lies within the current mesh square. That is, if the solution is denoted $\theta_s$ or $(\theta_{rs} + j \theta_{is})$, then $\theta_s$ must satisfy the relationships

$$0 \geq \theta_{rs} \leq 1 \text{ and } 0 \geq \theta_{is} \leq 1$$

(127)

to be a solution.
Also the solution must lie on the hyperbolic branch of $F_I(\theta) = 0$ which is currently being followed. The relationship is that (using Equation (113) and Equations (118) and (119):

\[
\text{if } (\text{ENTER } R - \text{CENTER } R) \cdot (\theta_{r_s} - \text{CENTER } R) > 0
\]

and

\[
\text{if } (\text{ENTER } I - \text{CENTER } I) \cdot (\theta_{i_s} - \text{CENTER } I) > 0
\]

then the solution of $F(\theta)$ is on the current branch line. If Equation (128) is not true then the solution $\theta_s$ is not on the current branch; however, the improper solution will be selected later when following another constant phase line of $F_I(\theta) = 0$.

Figure 8 illustrates the case where $\theta_s$ is a proper solution to $F(\theta) = 0$. Figure 9 illustrates the case where $\theta_s$ is not a proper solution to $F(\theta) = 0$.

If the value $\theta_s$ is a solution to $F(\theta) = 0$ the line $F_I(\theta) = 0$ is followed out of mesh square No. 2 through edge No. 2 into mesh square No. 3 through its edge No. 4 (see Figure 4). The line $F_I(\theta) = 0$ is then followed from mesh to mesh until the line $F_I(\theta) = 0$ exits from the contour rectangle (see Figure 10).
Figure 9. \( \theta_s \) is not the Proper Solution to \( F(\theta) = 0 \).

Figure 10. Contour Rectangle, Mesh Squares and Hyperbolic Lines of
\( F_1(\theta) = 0 \). \( \theta_s \) is a Solution of \( F(\theta) = 0 \).
At this point the value of the leading edge (counter clockwise direction) of the mesh square, through which the \( F_I(\theta) = 0 \) line exits, is noted. This mesh edge will also be along one of the contour rectangle sides. By identifying the mesh square in which this line occurs, it is then possible to later avoid re-entering this mesh through this exit edge while following the sides of the contour rectangle. This is important because if the search were allowed to re-enter at this location, the same zero of \( F(\theta) \) would be found twice.

The next step is to examine mesh square No. 3 for intersections of the line \( F_Y(\theta) \) and edge No. 1 of that mesh.

(J) The entire contour rectangle is now searched, by the methods described in Section I, so that finally all of the zeros of the function \( F(\theta) = 0 \), which lie within the contour rectangle, are known. These solutions must be considered approximate; however, in that the hyperbolic approximation has been used in solving for \( F(\theta) = 0 \).

(K) To alleviate this problem, use is made of the Newton-Raphson iteration method to remove the mesh approximation where:

\[
\Delta \theta = - \frac{F(\theta)}{\frac{dF(\theta)}{d\theta}} \bigg|_{\theta = \text{mesh}} \tag{129}
\]

If two approximate solutions are nearly equal, they may converge to a single value after application of the Newton-Raphson procedure.

In order to resolve the two solutions further analysis is needed. Let the two approximate solutions be \( \theta_{s,1} \) and \( \theta_{s,2} \). Consider a set of 9 mesh squares arranged in a 3x3 formation with the one containing the solution \( \theta_{s,1} \) at the center. There are 16 corner points associated with the corners of these squares (e.g. \( (\theta_x)_i \), \( i=1,2,3,...,16 \)). The functions \( F((\theta_x)_i) \) are then computed
for each of the 16 corners. These values will be denoted as \( F_{\text{mesh}}((\theta_x)_i) \). The set of functions given by the following equation are also computed,

\[
F_{\text{no}} \left( ((\theta_x)_i) \right) = F((\theta_{s,i}) + \frac{dF}{d\theta} \mid_{\theta_{s,i}} (\theta_{x_i} - \theta_{s,i}) \tag{130}
\]

\((i=1,2,3,\ldots,16)\)

The next step is to compute the following function at each of the 16 corners:

\[
F_{\text{total}} \left( ((\theta_x)_i) \right) = F_{\text{mesh}} \left( ((\theta_x)_i) \right) \cos^2 \theta + F_{\text{no}} \left( ((\theta_x)_i) \right) \sin^2 \theta \tag{131}
\]

where \( \theta \) is incremented from \( 0^\circ \) to \( 90^\circ \) in whatever increments are required to resolve the zeros of \( F(\theta) \). Considering the 9 squares one at a time the hyperbolic equation (Eq. 114) is solved to get 0, 1, or 2 solutions in each of the squares. Only those solutions \( (\theta_{q,j}) \) which exist inside of a given square are kept. The final solution \( (\theta_{p,1}) \) is chosen to be that one solution from the set \( (\theta_{q,j}) \) which is closest to the original \( \theta_{s,1} \).

The same type of analysis is then carried out for the solution \( \theta_{s,2} \). The final solutions \( \theta_{p,1} \) and \( \theta_{p,2} \) are iterated by the Newton-Raphson procedure to give the exact solutions. If these two solutions again converge to a single number, the value of \( \theta \) in Equation 131 is incremented and the entire procedure is again carried through with the values \( \theta_{p,1} \) and \( \theta_{p,2} \) replacing the values of \( \theta_{s,1} \) and \( \theta_{s,2} \). Note that completeness (in that the full set of zeros has been determined) is assured in Steps I and J whereas exact values must await Step K.
IV STRUCTURE AND DESCRIPTION OF THE MODESRCH COMPUTER PROGRAM

(A) Program Structure

The MODESRCH computer program finds the waveguide mode constants of the earth-ionosphere waveguide for radio propagation at ELF-VLF-LF. These mode constants are determined in the form of modal: eigenangles, attenuation rates, phase velocities and excitation factors.

The individual subroutines of the MODESRCH program are listed, in Table 3. In this table, those subroutines are divided into special sub-groups which are in accordance with their function in the total program structure.

Figures 11 through 14 illustrate the structure of the MODESRCH computer program. In particular, Figure 11 presents the general flow of the program showing the order in which the various subroutines and sub-groups are executed. This sequence is indicated, in the figure, by the numerals 1 through 6. Figure 12 shows the structure of the Ionospheric Full-Wave Integration procedure. Figure 13 gives the Free Space Integration Structure along with the structure of the routines for computing the values of the function $F(\theta)$ and the derivative of $F(\theta)$ with respect to $\theta$. Figure 14 illustrates the structure of the Root-Finder procedure.
### TABLE 3
**LIST OF PROGRAM ROUTINES FOR MODESrch**

#### MAIN FLOW
- MAIN
- WVGUID
- PRFLIN
- SETRH
- NEOUT
- NPOUT

#### ROOT-FINDER
- FZEROS
- NOMESH
- FINDF
- QUAD

#### F-FUNCTION
- FCTVAL
- FDFDT
- FINAL
- LAGRNG
- RBARS
- MDHNKL

#### FULL WAVE INTEGRATION
- INTEG
- INITLR
- QUARTC
- TMTRX
- SMTRX
- ENNU
- DIFFEQ
- ROUT

#### FREE SPACE INTEGRATION
- FSINTG
- MDHNKL

#### AUXILIARY ROUTINES
- CASIN
- CANG
- XFER
Figure 11. Structure of MODESRCH Computer Program.
Figure 12. Structure of the Ionospheric Full-Wave Integration Procedure for the MODESRCH Computer Program.
Figure 14. Structure of the Root Finder Procedure for the MODESRCH Computer Program.
(B) Computer Program Description

The function of each program subroutine is listed below:

(1) **MAIN ROUTINE:**

This routine controls the sequence of input and execution. It provides for NAMELIST input.

(2) **SUBROUTINE PRFLIN:**

This routine provides for input of electron density and collision frequency profile cards and for limiting the extent of the top and bottom of the electron density profile. Input of ion profiles is also provided for.

(3) **SUBROUTINE WVGUID:**

This routine controls the running sequence of the total computer program. Specific operations are to:

(a) Initialize the full-wave solution of the ionospheric integration routines.

(b) If the propagation frequency is greater than 1000 Hz and the variable 'IEXACT' is set equal to zero, then a reference height, (RK), is set through the routine SET RK. If either of the above is not true, the reference height is taken as zero.

(c) Initialize the ground reflection coefficient equations.

(d) Initialize the free-space integration calculations.

(e) Find the zeros (i.e., eigenangles) of the function F(θ) (see Equation 7), through the routine FZEROS. Lagrange interpolation is used if the frequency is greater than 1000 Hz or if the variable IEXACT is greater than zero. Otherwise, full wave solutions of the ionospheric reflection coefficients are computed.
(f) If the variable IEXACT is equal to zero or the frequency is greater than 1000 Hz, then the use of the routine FINAL is used to iterate on the eigenangle solutions of $F(\theta) = 0$ as the function is changed from that using Lagrange Interpolation to that using exact full-wave solutions of the ionospheric reflection coefficients.

(g) Order the eigenangles according to the value at the real part of the eigenangles.

(h) Compute and punch the eigenangles if the variable 'NEIGEN' is set equal to one. This is done through the routine NEOUT.

(i) Compute and punch the mode constant parameters if the variable 'NPUNCH' is set greater than zero. This is done through the routine NPOUT.

(4) **SUBROUTINE SETRH:**

(a) This routine selects the values at THETA, (e) at which full-wave solutions of the ionospheric reflection coefficients are to be carried out for use in Lagrange interpolation.

(b) This routine also selects the height, to the nearest whole kilometer, at which the function:

$$\sum_{\text{e-points}} \left( \sum_{\text{K-elements}} \left| \frac{d(R+1)}{dC} \right|^2 \right)$$

is minimized. In the above expression $C = \cos(\theta)$, $R$ is the reflection coefficient matrix and the summations are over all e-points at which full-wave solutions are
to be carried out for use in Lagrange interpolation. The summation is also over all elements of the \( R \) reflection matrix. Derivatives of the full-wave solutions are not computed; rather, derivatives of the Lagrange interpolation formula are used. Note that the full-wave solutions are carried out to the bottom of the profile, using the ionospheric reflection coefficient integration procedures (i.e., subroutine INTEG) after which, the full wave solution is then carried out in an upward direction through free space over a curved earth (by use of the free space integration routine FSINTG) to various heights which are increased successively by intervals of one kilometer. That height at which the minimum occurs is printed out as the 'REFLECTION HEIGHT' and is used; thereafter, as the Reference Height at which the function, \( F(\theta) \) is to be computed.

(5) SUBROUTINE NEOUT:
This routine provides for punched card output of eigenangles.

(6) SUBROUTINE NPOUT:
This routine provides for computing and printing tables and punched card output associated with the 'NPUNCH' option.

(7) SUBROUTINE INTEG:
This subroutine performs an integration of the differential equations for the ionosphere reflection matrix using Runge-Kutta integration formulas. The integration variables are
the elements of the matrix \((R+I)/C\) where \(R\) is the reflection matrix described in Equation 2. If 'IDERIV' is set non-zero, the derivatives of the \((R+I)/C\) elements \(\text{wrt} \ C = \cos(\text{THETA})\) are also integration variables. The set of heights at which pairs of integration steps begin and end is stored in the array 'RKHTS'. Sizes of the steps are determined by comparing the values of the elements of \((R+I)/C\) after a pair of steps and values of the same variables after a single double-size comparison step. The \((R+I)/C\) variables used in the comparison step are denoted as 'X'. If the difference is too large, the step size pairs are cut in half by adding a new height to the list in 'RKHTS'. In order to enhance stability in other parts of the program, no heights are deleted from the list unless the list is reset by a call to the SET RK subroutine.

(8) SUBROUTINE ENNU:
Computation of electron density and collision frequency (and ion density if appropriate) as a function of height. If profile was input from formatted cards, logarithmic interpolation is used. See also notes in subroutine PRFLIN.

(9) SUBROUTINE INITLR:
This routine computes the coefficients of the Booker Quartic equation and solves for its roots. The routine then computes the value of \((R+I)/C\). For reflection from a sharply bounded anisotropic ionosphere of semi-infinite extent where \(R\) is the reflection coefficient matrix. The solution is used as the initial condition for Runge-Kutta integration.
The solution is based on the material of reference 6. Derivatives wrt C=COS(THETA) are also computed if the variable 'IDERIV' is set non-zero.

(10) **SUBROUTINE QUARTIC:**
This routine obtains the solution for the roots of a fourth-order polynomial (QUARTIC Equation). A summary of pertinent equations is given in reference 6.

(11) **SUBROUTINE TMTRX:**
This routine computes the elements of the 'M' matrix of Equation 23. Those combinations of 'M' matrix elements used in the 'l' matrix which do not include use of THETA are the final output of this routine. Computations of quantities which are not functions of height or of THETA are carried out in the routine INIT T.

(12) **SUBROUTINE SMTRX:**
This routine computes the coefficients used in the differential equations for \((R+1)/C\). These are stored in common area 'S MTRX' and are analogous to 'S' matrix elements given by Equation 20. Derivatives of these coefficients wrt C=COS(THETA) are also computed if 'IDERIV' is set non-zero.

(13) **SUBROUTINE DIFFEQ:**
This routine computes the derivative of the differential equations for \((R+1)/C\), where \(R\) is the ionospheric reflection matrix. The equations are derived from the differential equations for 'R' as given in reference 5. Storage into the array 'R MTRX' is made so that the equations may be used with either the main set of integration variables,
or with the comparison set, X. Differential equations for the derivatives of \((R+1)/C\) wrt \(C = \cos(\theta)\) are also used if 'IDERIV' is set non-zero. Note that the variable called 'R' in this routine is actually \((R+1)/C\).

(14) **SUBROUTINE RBARS:**

This subroutine computes values of variables which may be used to form the elements of the RBAR matrix, where RBAR represents reflection of an ELM wave from the earth's surface. Namely, \(RBAR_{11} = \frac{DEN_{11}}{C \times NUM_{11} - DEN_{11}}\), or \(\frac{1.0}{RBAR_{11} + 1.0}/C = \frac{NUM_{11}}{DEN_{11}}\), and \(RBAR_{22} = \frac{DEN_{22}}{C \times NUM_{22} - DEN_{22}}\), or \(\frac{1.0}{RBAR_{22} + 1.0}/C = \frac{NUM_{22}}{DEN_{22}}\). Derivatives wrt \(C = \cos(\theta)\) are also computed if 'IDERIV' is set non-zero. Note that the equations are formulated in such a way that a smooth transition is made from the 'CURVED EARTH' form to the 'FLAT EARTH' form. See also notes in subroutine MDHNKL regarding definitions of Hankel function parameters. The value of earth's radius is such as to make \(2.0/RE = 3.14E-4\). Computation at \(\theta = 90^\circ\) is not excluded.

(15) **SUBROUTINE MDHNKL:**

This subroutine computes values of variables which may be combined to form modified Hankel functions as described in 'TABLES OF THE MODIFIED HANKEL FUNCTIONS OF ORDER ONE THIRD AND OF THEIR DERIVATIVES', by the staff of the computation laboratory, Harvard University Press, 1945. The notation is such that the actual \(H_1 = H_1 \times \exp(-E)\). The actual \(H_2 = H_2 \times \exp(E)\). Actual derivatives wrt argument, \(Z\), are \(H_1' \times \exp(-E)\) and \(H_2' \times \exp(E)\).
SUBROUTINE FSINTG:

This routine performs an integration of the differential equations for the ionosphere reflection matrix through a free space region over a curved earth. The integration may be performed in either a positive or negative height direction, but in this program the integration is always upward. The integration variables are the elements of the matrix \((R+I)/C\) where \(R\) is the reflection matrix. The solution is based on Budden, reference 7, in particular the material on pp. 118, 327-329, 336-338, and 343-345. If 'IDERIV' is set non-zero, the derivatives of \((R+I)/C\) elements wrt \(C=\cos(\theta)\) are also integration variables. Computation at \(\theta=90\) is not excluded. Also, the equations are formulated in such a way that a smooth transition is made from the 'CURVED EARTH' form to the 'FLAT EARTH' form for appropriate values of \(\theta\). The initial and final values of the integration variables are stored in the common area 'INTEGR'. This routine carries out the computation of height-gain coefficients \(Q,G,A,B\) for two conditions on the upgoing wave at each height=\(Z\), namely \(E_{1}=1, E_{2}=0\) and \(E_{1}=0, E_{2}=1\). Each of the coefficients \(Q,G,A,B\) is divided by \(C=\cos(\theta)\). Computation of the upgoing fields \(E_{1}\) and \(E_{2}\) at height=\(Z\) for the two conditions described above are also done. The integration proceeds from height \(Z_{b}\) to height \(Z_{t}\) when 'FSINTG' is called.
(17) **SUBROUTINE LAGRNG:**

This routine performs Lagrange interpolation of reflection coefficients in the COS (THETA) plane. Also to obtain derivatives of interpolated values WRT COS (THETA) the Lagrange interpolation formula itself is differentiated WRT COS (THETA).

(18) **SUBROUTINE FINAL:**

The routine for following the eigen-angles as the F function is changed from that using Lagrange interpolation to that using exact full-wave solution values of reflection coefficients.

(19) **SUBROUTINE FZEROS:** (TLEFT, TRIGHT, TBOT, TTOP, TMESH, TOL, MPRINT, ZEROS, NR Z)

For Root-Finder FZEROS is a routine for finding the zeros of a complex function, F, which lie within a specified rectangular region of the complex (THETA) plane, provided the function has no poles in the vicinity of the rectangle.

**EXPLANATION OF PARAMETERS--**

- **TLEFT** - value of real part of THETA at left edge of rectangle.
- **TRIGHT** - value of real part of THETA at right edge of rectangle.
- **TBOT** - value of imag part of THETA at bottom edge of rectangle.
- **TTOP** - value of imag part of THETA at top edge of rectangle.

66
TMESH - set equal to about half the average spacing between zeros within the rectangle. A smaller value may be used as a safety measure, but too small a value will result in excessive run time.

TOL - tolerance to which zeros are to be found. If two zeros are closer than 'TOL', the root-finder will stop with an error message.

MPRINT - normally set to zero. A non-zero value leads to print-out for debugging.

ZEROS - output list of (complex) values of THETA at which zeros are found.

NR Z - the number of zeros found.

SUBROUTINES TO BE PROVIDED--

FCT VAL - (THETA,F) - to return the value of the function, F, at the point in the complex plane specified by 'THETA'.

F DFDT - (THETA,F,DFDT) > same as 'FCT VAL' except that the derivative, DFDT, of the function WRT THETA must also be returned.

(20) SUBROUTINE QUAD:

For Root-Finder finds the solution for the real roots of a Quadratic Equation of the form A*X**2+2.0*B*X+C=0.0, where X is called 'SOL' in this routine. The number of real roots found is given by 'NR SOL'. A value of 1 for 'NR SOL' results from the Quadratic Equation approaching linearity. Used by subroutines FZEROS and NO MESH.
(21) SUBROUTINE NO MESH:
For Root-Finder the routine for finding exact (in the sense of NO MESH approximation) locations of zeros of the function, F, for which a complete, but approximate, set was found in subroutine FZEROS.

(22) SUBROUTINE FINDF:
This routine calls upon the routine FCT VAL to compute F(\theta).

(23) SUBROUTINE FCT VAL:
This routine is for computing F function values at mesh points, called on principally by subroutines FZEROS and NO MESH. This is the modified F function which has no poles and which has no zeros at \theta=90.

(24) SUBROUTINE F DFDT:
This routine for computing function F(\theta) values and their derivatives WRT THETA at arbitrary values of THETA. Called on principally by subroutines NO MESH and FINAL. This is the modified F function which has no poles and which has no zeros at \theta=90.

(25) SUBROUTINE XFER:
Routine for transferring one array into another.

(26) SUBROUTINE R OUT
Output of Runge-Kutta integration variables, used only for debugging.

(27) FUNCTION CASIN
Routine for computing the complex sine function of a complex argument.
(28) FUNCTION CANG:

Routine for computing the arc tangent of a complex angle.
V. RUNNING THE PROGRAM

A. Control Cards

These cards contain mnemonic words denoting the program operations to be executed. These words will be assumed to begin in column 1 of the card. In the text of this outline, these words will be enclosed in apostrophes. Blanks will be denoted by a space indicated by ' ' to clarify positioning of blank columns. These cards are summarized in Table 4.

(1) 'PROFILE-i' initiates reading of the ionospheric profile. An alternative method for specifying an ionospheric profile is presented in Section B.

The value of i indicates the number of species to be used. It may have one of two possible values: 1 or 3. If the position for i is left blank, i = 1 is assumed.

The 'PROFILE-i' card must be followed by an alphanumeric card which may be used to identify the profile.

The profile cards contain the following parameters punched according to the format (F7.2,4X,5(1X,E9.2)): height in kilometers and the specie densities in number per cubic centimeter. The first specie is electrons. The cards are input in order of descending altitude and the profile is terminated with a dummy height with value less than zero. A maximum of 101 profile cards may be used.

In the case of i = 3, the program assumes that the first two species read are electrons, \( N_e \) and singly charged positive ions, \( N_i^+ \). The required third specie is thus singly charged negative ions, \( N_i^- \), and their density is computed and printed by the program as \( N_i^- = N_i^+ - N_e \).
In the integration of the ionospheric reflection elements through the ionosphere, the program interpolates exponentially between input values. Thus, for example, an exponential profile can be input with only the top and bottom densities specified.

(2) 'COLFREQ' initiates reading of an ionospheric collision frequency profile. An alternative form for specifying exponential collision frequencies is discussed in Section B. This profile, if used, must follow the 'PROFILE_i' input because i specifies the number of species.

The 'COLFREQ' card is immediately followed by the profile. The input format and order is the same as used for 'PROFILE_i', except that collision frequencies in collisions per second for electrons, and when i is greater than 1 for positive ions and negative ions, must be input. The profile is input in order of descending height and is terminated by a dummy height of value less than zero. A maximum of 50 collision frequency profile cards may be used. As with the ionospheric specie profiles, the program interpolates exponentially between input values and there need be no correspondence between the altitudes used to define the two profiles.

When 'COLFREQ' is used, the collision frequency profile can be over-ridden later in the job by the exponential collision frequency specification described in Section B-1. Note that in this case the exponential parameters, 'COEFNU' and 'EXPNU', must be input through NAMELIST.

(3) 'ID' indicates that the next card is identification for the data being run. This identifying card is merely read and printed. It may contain any alphanumeric information desired.

(4) '_&DATUM' and '_&END' initiate and terminate reading of Namelist. The form of this input is described below.
All cards appearing between '_&DATUM' and '_&END' must have a blank in column 1. In the sample data decks this will be indicated by '_'.

Variable names are punched with no imbedded blanks. The name is followed by an equal sign (=) and the value of the variable punched in any valid format. Integer variables may only be input with integer numbers and real variables may be input with either integer numbers or floating point numbers. The value of the variable must be followed by a comma. All blanks in the values punched are treated as zeros. A comma is the only valid delimiter. For example, 'I=1_' would result in I=10. If a variable name is repeated, the second value will be used.

If the variable defines an array, the array may be filled by a string of values separated by commas. There need not be as many values specified in the string as there are in the dimension of the array. The string of variables may be continued on additional cards by not repeating the variable name. Alternatively, the value of single elements may be specified by subscripting the variable name before the equal sign such as 'ARRAY(3) = -6.E3,'.

Complex variables must be input in order of real and imaginary parts. Arrays of complex variables are input the same as described above except that twice as many values are required. For example, 'CARRAY = 1., 3., -1., -3.,' is used to input CARRAY(1) = (1., 3.) and CARRAY(2) = (-1., -3.).

(5) 'QUIT' signals normal termination of a job. This is not required provided a /* card is used at the end of the input deck.

(6) A reasonable order of input is shown in Table 4.
TABLE 4
SUMMARY OF CONTROL CARDS

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>PROFILE_i</td>
<td>read ionospheric profiles (i=1 or 3)*</td>
</tr>
<tr>
<td>COLFREQ_i</td>
<td>read ionospheric collision frequency profiles</td>
</tr>
<tr>
<td>ID</td>
<td>read and print identification card(80 card columns)</td>
</tr>
<tr>
<td>_&amp;DATUM</td>
<td>begin Namelist read</td>
</tr>
<tr>
<td>_&amp;END</td>
<td>end Namelist read and begin Waveguide calculations</td>
</tr>
<tr>
<td>QUIT</td>
<td>end of job (optional)</td>
</tr>
</tbody>
</table>

/*
*Note: If i not specified, 1 is assumed.
*/

B. Input Variables
(1) General Input variables needed for computations of the modal solutions.

'FREQ' is the frequency in kilohertz.

'MRATIO' is the ratio of the mass of the specie to the mass of an electron. Default values are for an electron, 'MRATIO = 1'. For positive or negative ions 'MRATIO = 58000'.

An exponential collision frequency may be specified for each specie. 'COEFNU' is the collision frequency, in collisions per second, at the ground. 'EXPNU' is the height variation in inverse kilometers. The collision frequency at an altitude of z kilometers in this case is defined by:

\[ \nu(z)j = \text{COEFNU}(j) \times \exp(\text{EXPNU}(j) \times z) \]

where j=1 is for electron-neutral particle collisions, j=2 is for positive ion-neutral particle collisions and j=3 is for negative ion-neutral particle collisions.
The parameter 'NUFLAG' allows for a choice of exponential collision frequency profiles to be used in the calculations. In this instance values are not assigned to COEFNU(j) and EXPNU(j); instead default values, which are built into the MODESRCH program are used in the calculations. With 'NUFLAG = 0', the values of the exponential parameters 'COEFNU' and 'EXPNU' (as described in Wait's TN 300, Reference 3) are used. If 'NUFLAG = 1', the values of these exponential parameters (as computed by Moler, Reference 13) are used. The special values are:

'NUFLAG = 0' (Wait):
- COEFNU(1) = 1.816E+11, coll/sec
- COEFNU(2) = 4.540E+9, coll/sec
- COEFNU(3) = 4.540E+9, coll/sec
- EXPNU(1) = -0.15 km⁻¹
- EXPNU(2) = -0.15 km⁻¹
- EXPNU(3) = -0.15 km⁻¹

'NUFLAG = 1' (Moler):
- COEFNU(1) = 4.303E+11 coll/sec
- COEFNU(2) = 1.076E+10 coll/sec
- COEFNU(3) = 1.076E+10 coll/sec
- EXPNU(1) = -0.1622 km⁻¹
- EXPNU(2) = -0.1622 km⁻¹
- EXPNU(3) = -0.1622 km⁻¹

If an exponential electron collision frequency profile is to be used other than those described by Wait or Moler, 'NUFLAG' must be set = 0 and the proper values assigned to 'COEFNU(j)' and 'EXPNU(j)'.

It is important to note that if the collision frequency profile is entered as tabulated values via the control card 'COLFREQ' and then in the same job step, additional calculations are wanted for new collision
frequency values, as input through the Namelist variables 'COEFNU(j)' and
EXPNU(j)'; then, 'NUFLAG' must be set equal to zero and new values
assigned to 'COEFNU(j)' and 'EXPNU(j)'.

An exception to this procedure occurs when the new values of
'COEFNU(j)' and 'EXPNU(j)' are those denoted as "Moler's values". In this
case the parameter 'NUFLAG' is set equal to 1 and no values need be
assigned to 'COEFNU(j)' or 'EXPNU(j)' in that this is done automatically
within the program.

'RHO' is the distance in Mm from the transmitter. This variable
is used only in punched output. It is useful to further identify punched
data that is being generated along a propagation path.

The geomagnetic field is specified by three variables. 'AZIM' is
the clockwise angle between magnetic north and the horizontal propagation
direction in degrees east of north. 'CODIP' is the magnetic co-dip angle
in degrees from the vertical down. The magnetic equator is specified by
'CODIP = 90'. 'MAGFLD' is the magnetic field intensity in webers per
square meter.

The ground conditions are specified by two variables. 'SIGMA' is
the conductivity in mhos per meter and 'EPSR' is the relative dielectric
constant.

'H' is the height in kilometers at which the modified refractive
index is unity. This is the height to which the eigenangles are referred.
'REFLHT' is the height in kilometers which is used to normalize
the excitation factor defined by Wait. 'REFLHT' must not be zero.

'GMAX' is the maximum separation (in degrees) of the theta angles
at which full wave solutions, to be used in Lagrange interpolation, are
carried out.
'SEP' is a real variable which defines the minimum separation between eigen solutions.

'RTOL' is the variable which determines the precision of the Runge-Kutta integration for the ionospheric reflection elements.

'NPUNCH' determines the specific parameters to be punched on output cards.

'CARDPN' is an option for punching or not punching on cards the outputs from the 'NPUNCH' option.

'CARDPT' is an option for printing, or not printing, the values computed using the 'NPUNCH' option.

'NEIGEN' is the option for punching eigenangles on the cards.

'TALT' is the height of the transmitting antenna in km.

'RALT' is the height of the receiving antenna in km.

Note: Input of TALT and RALT is only applicable to the calculation of modal excitation factors as described in Section V-C-2. Computed output values, as a function of 'RALT' and 'TALT', are obtained only through the option NPUNCH = 7.

'TMESH' is the angular dimension of the mesh squares contained in the contour rectangle. This value is in degrees. 'TMESH' should always be smaller than the average separation of the eigenangles. Also 'TMESH' should be chosen smaller than the separation between the values of 'RANGER' (maximum) and 'RANGER' (minimum) or the values of 'RANGEI' (maximum) and 'RANGEI' (minimum). Unless 'TMESH' is assigned a value via NAMELIST, it is computed within the program as:

\[ \text{TMESH} = \sqrt{3.75/freq_{kHz}} \]  

(132)

'LUB' is a real variable used to terminate the iterative process. 'LUB' is used to test the iterative change in the magnitude of the complex
eigenangles. The iteration is stopped when the change in the magnitude of the complex eigenangle is less than or equal to 'LUB'. Unless 'LUB' is assigned a value via NAMELIST, it is computed within the program as:

\[ \text{LUB} = \sqrt{15.0/\text{freqkHz}}(0.01) \] (133)

'IEXACT' is an option that determines whether Lagrangian interpolation or the "full wave" solution is used in initial stages of the program. IEXACT = 0 is for Lagrangian. IEXACT = 1 is for "full wave".

(2) **Exponential profiles (electrons only)**

An exponential, electrons only, ionospheric profile may be specified. The profile is computed as described in Wait's NBS TN 300, Reference 3, where

\[ \omega_r(z) = 2.5 \times 10^5 \exp(\beta(z-h')) = 3.18254 \times 10^9 \frac{N(z)}{\nu(z)} \] (11)

and \( \beta \) is in inverse kilometers, \( z \) and \( h' \) are in kilometers, \( N(z) \) is the electron density in electrons per cubic centimeter and \( \nu(z) \) is the collision frequency in collisions per second. The values used to compute \( \nu(z) \) are 'COEFNU(l)' and 'EXPNU(l)' and are given in Section V-B-l. Note that the term 'BETA' is \( \beta \) and 'HPRIME' is \( h' \).

The equation for computing the electron density (\( N(z) \), electrons/cm\(^3\)) at height \( z \) is given by:

\[ N(z) = 7.8554 \times 10^{-5} \cdot \text{COEFNU(1)} \cdot \left[ \exp \left( (\text{BETA} + \text{EXPNU(1)})z - \text{BETA} \cdot h' \right) \right] \] (134)

'SCLHTS' is a real variable which indicates the number of scale heights above \( h' \) which will be used to denote the top of the profile when the \( \beta, h' \) exponential electron density profile is used.
The top of the profile may be modified by choosing a value for the variable SCLHTS. The relationship is that:

\[
\text{Top Height} = \text{Integer} \left( H\text{PRIME} + \frac{\text{SCLHTS}}{\text{BETA}} + 0.5 \right) \quad (135)
\]

The nearest integral value of altitude as given by 'Top Height' is used to set the top of the profile. All heights above this value are neglected in the MODESRCH computations.

The bottom of the exponential profile is set by assigning a minimum value (i.e. ENMIN) of electron density in electrons/cm\(^3\). All altitudes (z), at which the corresponding density is less than 'ENMIN' will be neglected.

(3) **VLF/LF - Tabular Electron-Ion Density Profiles.**

Computing efficiency for VLF and LF may be improved by cutting off the top and bottom heights of the input ionospheric profile.

When the particle density profiles are input to the program via 'PROFILE\(i\)' (i = 1 implies electrons only) (i = 3 implies electrons and ions), the top and bottom of the profile may be cut off using the criteria, as presented in Section II, D-2. In that section it is stated that the top part of the input ionospheric profile which may be neglected in the calculations is defined as that altitude(z) above which:

\[
B(z) > \text{Maximum (TEMPO-B, 20 } \frac{\omega}{v_e} \text{)}
\]

where 'TEMPO-B' is assigned a value (e.g. 2.0) and B(z) is given by eq. 16. Also an additional requirement for cutting off the top of the profile is that the electron density must be as great as 'TOPEN'. The variable 'TOPEN' is assigned a value (e.g. 1000 electrons/cm\(^2\)). The top of the profile is taken to be the higher of the two heights corresponding to B(z), (equation 17), and to 'TOPEN'.

78
The bottom of the input ionospheric profile is neglected (for VLF/LF cases) at those altitudes for which the computed value of $B$ in equation 16 is less than 'CUTOFF'. The variable 'CUTOFF' is assigned a value (e.g. 0.0001). The relationship for Cutoff is:

$$B(z) < \text{CUTOFF} \quad (18)$$

If it is not desired to cut off the profile bottom for certain calculations then the Namelist parameter 'ICUT' must be set equal to zero.

(4) ELF Tabular Electron-Ion Density Profiles

The same criteria for the variable 'PROFILE_i' as was presented for the VLF/LF case is applicable to ELF, (i.e. $i = 1$ or blank implies electrons only and $i = 3$ corresponds to electrons and ions).

In the case of ELF the profile top and bottom are not cut off and the Namelist variable 'ICUT' is not applicable.

(5) Namelist Variables with Initial Values.

Initial values of the Namelist variables are presented in Table 5.
### TABLE 5
NAMELIST VARIABLES AND INITIAL VALUES

<table>
<thead>
<tr>
<th>NAME</th>
<th>VALUE</th>
<th>UNITS</th>
<th>NAME</th>
<th>VALUE</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>NPROF</td>
<td>None</td>
<td>-</td>
<td>ICUT</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>FREQ</td>
<td>0.0</td>
<td>kHz</td>
<td>NUFLAG</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>H</td>
<td>50.0</td>
<td>km</td>
<td>COEFNU(3)</td>
<td>1.816E+11, 4.540E9</td>
<td>cml/sec</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>4.540E9</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.15, -0.15,</td>
<td>km^{-1}</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.15</td>
<td></td>
</tr>
<tr>
<td>REFLHT</td>
<td>50.0</td>
<td>km</td>
<td>EXPNU(3)</td>
<td>-0.15, -0.15,</td>
<td>km^{-1}</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-0.15</td>
<td></td>
</tr>
<tr>
<td>RHO</td>
<td>0.0</td>
<td>Mm</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AZIM</td>
<td>None</td>
<td>degrees</td>
<td>SEP</td>
<td>0.1</td>
<td>degrees</td>
</tr>
<tr>
<td>CODIP</td>
<td>None</td>
<td>degrees</td>
<td>RTOL</td>
<td>0.001</td>
<td>degrees</td>
</tr>
<tr>
<td>MAGFLD</td>
<td>None</td>
<td>WEBERS/m^2</td>
<td>GMAX</td>
<td>5.0</td>
<td>degrees</td>
</tr>
<tr>
<td>SIGMA</td>
<td>4.64</td>
<td>mho/m</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>EPSR</td>
<td>81.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BETA</td>
<td>None</td>
<td>km^{-1}</td>
<td>TOPEN</td>
<td>1000.</td>
<td>electrons/cm^3</td>
</tr>
<tr>
<td>HPRIME</td>
<td>-99.0</td>
<td>km</td>
<td>ENMIN</td>
<td>0.1</td>
<td>electrons/cm^3</td>
</tr>
<tr>
<td>SCLHTS</td>
<td>3</td>
<td>km</td>
<td>RANGE</td>
<td>None</td>
<td>degrees</td>
</tr>
<tr>
<td>MRATIO(3)</td>
<td>1,58000.0</td>
<td>58000.0</td>
<td>RANGEI</td>
<td>None</td>
<td>degrees</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>TEMPOB</td>
<td>2.0</td>
<td></td>
</tr>
<tr>
<td>CUTOFF</td>
<td>0.0001</td>
<td></td>
<td>TALT</td>
<td>0.0</td>
<td>km</td>
</tr>
<tr>
<td>NEIGEN</td>
<td>0</td>
<td></td>
<td>RALT</td>
<td>0.0</td>
<td>km</td>
</tr>
<tr>
<td>NPUNCH</td>
<td>1</td>
<td></td>
<td>IEXACT</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>CARDPT</td>
<td>0</td>
<td></td>
<td>TMESH</td>
<td>-1.0</td>
<td>degrees</td>
</tr>
<tr>
<td>CARDPN</td>
<td>1</td>
<td></td>
<td>LUB</td>
<td>-1.0</td>
<td></td>
</tr>
</tbody>
</table>

**Notes:** Numbers in parenthesis after Name indicate dimensions. Lack of decimal point indicates integer variables.
The Eigenangle ($\theta$)

The modal solutions of equation 1, which are denoted as complex eigenangles ($\theta = \theta_{\text{real}} + i\theta_{\text{imaginary}}$), are determined within a bounded region of the complex "THETA" plane defined by a set of limiting THETA values. This bounded region will be called the "contour rectangle" and the limiting THETA values are denoted by "RANGER" (minimum and maximum) and 'RANGEI' (maximum and minimum). These parameters are portrayed in Figure 15.

Where Point:

- A: RANGER (minimum), RANGEI (maximum)
- B: RANGER (maximum), RANGEI (maximum)
- C: RANGER (minimum), RANGEI (minimum)
- D: RANGER (maximum), RANGEI (minimum)

The small squares inside the "contour rectangle" are called the "Mesh Squares". 'RANGER' is the real part in degrees of the minimum and maximum values assigned to the contour rectangle. 'RANGEI' is the imaginary part in degrees of the minimum and maximum values assigned to the contour rectangle.

![Complex Theta Plane](image-url)

Figure 15. Showing the Contour Rectangle ABCD and Mesh Squares
i.e. RANGER = Minimum, maximum,
RANGEI = Maximum, minimum

Note: The minimum value must be given before the maximum value for RANGER while the maximum value must be given first for RANGEI.

During program execution, the contour rectangle, as described by RANGER and RANGEI, is automatically divided into small sub-rectangles. This is done to increase computation efficiency. The division process is applied only to the RANGER values while the RANGEI values remain the same as those input through NAMELIST.

The algorithm is that the number of segments 'NS' into which the original contour rectangle is divided is given by:

\[ NS = \text{TRUNCATION} \left\lfloor \frac{\text{RANGER(max)} - \text{RANGER(min)} - \text{LUB}}{\text{GMAX}} \right\rfloor + 1 \] \hspace{1cm} (136)

The number of degrees in each segment is given by:

\[ \Delta^\circ = \frac{\text{RANGER (max)} - \text{RANGER(min)}}{NS} \]

C. Program Outputs

For purposes of clarity in describing program outputs, the following symbols will be used:

\[ k = \text{wave number in km}^{-1} \]
\[ h_r = \text{reference height for Wait's excitation factor (REFLHT)} \]
\[ i = \sqrt{-1} \]
\[ \theta = (\theta_r, \theta_i) = \text{final mode solution referenced to } H \]
\[ \theta' = (\theta'_r, \theta'_i) = \theta \text{ referenced to the ground} \]
\[ \alpha = -8686 \ \text{k} \ \text{Im}(\sin \theta') = \text{attenuation rate in dB/Mm} \]
\[ v/c = 1/\text{Re}(\sin \theta') = \text{phase velocity/speed of light} \]

Note: The above procedure for sub-dividing the RANGER value is done only for VLF/LF calculations and is not applied for calculations at ELF.
\lambda_{mn} = \text{excitation factor}

X_{mn} = \text{modified excitation factor}

\Lambda_{mn} = \text{normalized excitation factor} = -0.5ikh_{r}X_{mn}

The terms, \(X_{mn}, \Lambda_{mn}\) and \(\lambda_{mn}\), are elements of 3 by 3 complex matrices. The individual elements are subscripted first by \(m = V, E\) or \(B\) to specify a vertical, end fire or broadside dipole and second by \(n = Z, Y\) or \(X\) to specify the field component excited. Thus, \(X_{VZ}\) is the \(z\) component of the electric field excited by a vertical dipole source. The terms \(X_{mn}\) and \(\Lambda_{mn}\) are also functions of the transmitter and receiver altitudes (TALT and RALT). These are the excitation factors mentioned in Section II, H. The terms \(X_{0}\) and \(\Lambda_{0}\) specify \(X_{VZ}\) and \(\Lambda_{VZ}\) for both transmitter and receiver at the ground. In both printed and punched output, each \(X_{0}, X_{mn}\) and \(\lambda_{mn}\) is given as magnitude and phase in radians and each \(\Lambda_{0}\) and \(\Lambda_{mn}\) is given as 20 log (magnitude) and phase in radians. Note that in the MODESRCH computer program all the \(\lambda_{mn}\) terms are used whereas only \(X_{VZ}, X_{EZ}, X_{BZ}, X_{0}\) and \(\Lambda_{0}\) of the other excitation terms are utilized.

The output from the computer program consists of the following:

(1) Computer Listing Output
   
   (a) ID Information
   (b) PROFILE Information
   (c) NAMELIST Information
   (d) Values of 'TMESH' and 'LUB'
   
   Also, heights and electron densities of top and bottom of profile.
   (e) Value of TEMPO-B at top of the profile, value of Wait's \(\Omega_{R}\) at top of the profile or height of the profile where Wait's \(\Omega_{R} = 2.5 \times 10^{5}\)
(f) Reflection height -- height at which ionospheric reflection coefficients are interpolated and the height at which the ground reflection coefficients are computed (e.g., the height where \( F(\theta) \) is computed).

(g) Theta range for each 'contour rectangle'.

(h) \( H \): the height at which the eigenangle solutions are computed.

(i) Mode No. \( n \)

(j) \( \Theta(\theta_r, \theta_i) \): The final mode solution referenced to \( H \) (\( \theta_r \) and \( \theta_i \) are in degrees).

(k) \( \alpha(\theta'_n) \): Attenuation rate dB/Mm where \( \alpha = -8666K \cdot \text{Im} \cdot \text{SIN} \cdot \theta' \) and where \( K \) is the wave number in km\(^{-1}\).

(l) \( V/C(\theta'_n) \): The phase velocity/speed of light

\[
V/C = \frac{1}{\text{Real}(\text{SIN} \cdot \theta')}
\]

(m) Magnitude and Phase of the modal excitation factor as defined by Wait. (\( \Lambda_0 \) (dB), \( \phi_0 \) (radians)).

(n) \( \Theta(p, \theta'_i) \): The \( \theta \) eigenangle referenced to the ground (\( \theta'_p, \theta'_i \) are in degrees).

(o) Polarization mixing ratio:

\[
\rho = \left| \frac{(1 - \text{I}_R^1 \text{I}_R^1)}{(\text{I}_R^1 \text{I}_R^1)} \right| = \left| \frac{\text{I}_R^1 \text{I}_R^1}{(1 - \text{I}_R^1 \text{I}_R^1)} \right| \quad (138)
\]

When \( \text{NEIGEN} = 1 \) that ionospheric height where the electron density is equal either to the value of the input variable \( \text{ENMIN} \) (for \( \beta, h' \) electron-density profiles) or to the lowest height of the input profile (for tabular electron density profiles) is printed. This parameter is
denoted as "D" and corresponds to the variable "D" in the reference WAVEGUIDE program where D is defined as the height in kilometers below which ionospheric integration is terminated.

Also when NEIGEN = 1, individual eigenangles (referenced to the height H) are printed.

If NEIGEN = 0, the above variables are not printed.

(2) Punched Output

When NEIGEN = 1 the value of "D" as described above is punched and also the individual eigenangle solutions are punched. These parameters are punched in the format of usual NAMELIST variables. When NEIGEN = 0, these variables are not punched.

Additional punched output from the program is obtained with the variable NPUNCH.

NPUNCH options for obtaining output cards:

NPUNCH=1*** gives output cards for horizontally homogeneous mode-sum or WKB-sum in terms of (T'S).
NPUNCH=2*** gives output cards in terms of SNVLF output.
NPUNCH=7*** gives output cards for input into ELF-WKB mode-sums.
NPUNCH=8*** gives output cards for input into mode conversion in terms of (FOVR).
NPUNCH=9*** gives output cards for input into mode conversion in terms of both (T'S) and (FOVR).
NPUNCH=0*** the above variables are not computed.
CARDPN=0*** gives usual mode summary print out but no cards.
CARDPN=1*** gives usual mode summary print and cards.
CARDPT=0*** causes the output cards to not be listed.
CARDPT=1*** causes the output cards to be listed.

85
When NPUNCH equals 1, 2, 7, 6, or 9, each entry into '&DATUM' results in a card containing RHO, FREQ, AZIM, CODIP, MAGFLD, SIGMA and EPSR in the format:

('R',F7.3,' F',F8.4,' A',F8.3,' C',F8.3,' M',E10.3,' S',E10.3,' E',F5.1)

and each exit via '&END' results in a blank card. In other program descriptions, the first card is referred to as the RFACMSE header card.

NPUNCH = 1 generates two cards per mode in the order

(1, e', T₁, T₂) and (2, e', T₃, T₄)

with format (I1, 2F9.5, 4E15.8).

Where:

\[ T₁ = S₁² \frac{(1 + \text{R}₁ \text{R}₄)}{\frac{\partial F}{\partial \theta}} \left( 1 - \text{R}₁ \text{R}₄ \right) \frac{\text{D}_{11}}{\text{D}_{11}} \]

\[ T₂ = S₂² \frac{(1 + \text{R}₁ \text{R}₄)}{\frac{\partial F}{\partial \theta}} \left( 1 - \text{R}₁ \text{R}₄ \right) \frac{\text{D}_{22}}{\text{D}_{22}} \]

\[ T₃ = S₃² \frac{(1 + \text{R}₄ \text{R}₄)}{\frac{\partial F}{\partial \theta}} \left( 1 + \text{R}₄ \text{R}₄ \right) \frac{\text{D}_{12}}{\text{D}_{12}} \]

\[ T₄ = \frac{\text{R}₄}{\text{R}₄} \]  

(139)
0' = complex eigenangle referenced to the ground and is punched as \( \theta_r \) (degrees) and \( \theta_i \) (degrees). Examination of Table 1 and equation 139 shows that the excitation "X" values can be obtained from the "T" values. The "T" values are punched as complex numbers.

NPUNCH = 2 generates a single card per mode containing 
(a, V/C, \( |\Lambda_o|, \phi_{\Lambda o} \)) with format (4F10.5). Here \( \Lambda_o \) is in dB above one UV/meter and \( \phi_{\Lambda o} \) is in radians. This format is denoted as SNVLF.

NPUNCH = 7 generates two cards per mode. The first card contains \( \theta' \) with a comma after the real and imaginary parts. Here \( \theta_r' \) and \( \theta_i' \) are in degrees. The second card contains \( X_{\text{VZ}}, X_{\text{BZ}} \) and \( X_{\text{EZ}} \) for transmitter and receiver heights of RALT and TALT. There is a comma after each term. This form of output is suitable for a Namelist type input.

These values are punched as magnitude and phase of the complex excitation number. Two comments which are pertinent to the NPUNCH = 7 output are:

(a) The values of the \( X_{\text{mz}} \)'s are computed as functions of the transmitter and receiver heights (TALT and RALT).

(b) If the punched output is to be used as input in the WDB-ELF Field Strength computer program (reference 14), then the NAMELIST terms TALT and RALT, input to the MODESrch program must be zero for compatibility.

NPUNCH = 8 generates one card per mode. The card contains the variables (\( \theta' \), \( |X_o| \), \( \phi_{Xo} \), FOVR, HOFWR) using the format (OP2F9.5,1PE14.5, OPF9.5,1P2E16.8,OPF7.2) Here \( \theta' \) is punched as \( \theta_r' \) and \( \theta_i' \) each in degrees. \( \phi_{Xo} \) is in radians and 'FOVR' is a complex number. The variable 'HOFWR' is the profile height (km) which corresponds to a value of Wait's \( \Omega - R = 2.5\times10^5 \).
Examples of Program Inputs and Outputs

Example I illustrates some of the options for data input. This example is for an exponential electron density profile defined by 'BETA' and 'HPRIME' parameters. In this case the exponential collision frequency profile, corresponding to that described by Wait, (Reference 3), is included by setting the 'NUFLAG' variable to zero. Also, putting NPUNCH = 1 or greater causes the mode constants to be printed in the output listing and if 'CARDPN=1', (which is the default value) functions of the excitation factors are punched. The value NEIGEN = 1 gives both a listing and punched cards for the eigenangles at a height of 50 km. The printed output is also shown in example I.

Example II lists the output cards as computed from 'NPUNCH = 1' as presented in example I. Note that the first card gives the values for RHO, FREQ, AZIM, CODIP, MAGFLD, SIGMA and EPSR as used in the modal calculations. The card will be denoted as the RFACMSE card. Also the complex eigenangles at the ground are also punched on the cards as $\theta_1^r$ and $\theta_1^i$. There are two cards per mode corresponding to $T_1$(Real), $T_1$(Imaginary) and $T_2$(Real) and $T_2$(Imaginary) on card No. 1. Card No. 2 contains $T_3$(Real) and $T_3$(Imaginary), $T_4$(Real), and $T_4$(Imaginary).

Where 'NLIGEN = 1' as in example I, the values of the eigenangles at a height of $H = 50$ km are also punched one per card as $\theta$(real) and $\theta$(imaginary). The value of D as punched corresponds to the bottom of the electron density profile.

Example III lists the punched card output obtained from the input parameters of example I when 'NPUNCH = 2'. Here the first card is the usual RFACMSE card. The second card gives the propagation frequency. The following cards give, for each mode, the values of the attenuation
rate (dB/Mm); the ratio V/C; the magnitude of Wait's excitation factor (dB); and the phase of Wait's excitation factor (radians).

Example IV lists the punched card output obtained from the input parameters of example I where 'NPUNCH = 8'. This output is suitable for input into the mode conversion computer program described in reference 12. The first output card is the RFACMSE card. The following cards give for each mode the values of $\theta'$ (real) (degrees), $\theta'$ (imaginary) (degrees); the excitation component from a vertical exciter for the vertical electric field at $z=0$, (magnitude (not dB) and phase (radians)); the real and imaginary parts of the complex number 'FOVR'; and the height, in km, used for mode conversion, 'HOFWR'.

Example V presents input data where the electron density and positive ion density are entered in tabular form. Note that in this instance the collision frequencies for electrons, positive and negative ions are all entered automatically within the program through the variable 'NUFLAG = 1'. This corresponds to the use of these terms as computed by Moler (reference 13).

Example VI shows the printed output obtained from the input of example V. Note that for this case the profile top is not cutoff whereas the bottom is cutoff through the variable 'CUTOFF'.

Example VII lists the punched card output obtained from example V for NPUNCH = 9. Note that this punch option gives a combination of NPUNCH = 1 and NPUNCH = 8. There are three cards per mode.

Example VIII presents input data similar to example V. In this case the variable 'TOPEN' has been set to a small number so as to illustrate the output messages obtained when the top of the profile is cutoff.
Example IX shows the input when both particle density (electrons and positive ions) and collision frequency profiles (electron, positive ions and negative ions) are input in tabular form. This procedure will generally be the case for computations at ELF.

Example X illustrates the usual printed output obtained for NPUNCH greater than 0. Note that since this is an ELF case only one mode is found.

Example XI lists the output obtained for NPUNCH = 7. This output is compatible with a program which computes WKB-Fields at ELF (reference 14). Note however in this regard the NAMELIST variables 'TALT' and 'RALT' must both be zero. The punched card output for NPUNCH = 7 contains the following: the RFACMSE card; a card with \( \theta^V \) (degrees) and \( \theta^L \) (degrees); and a single card with the magnitude (not dB) and phase (radians) of the vertical excitation factors for vertical, end-or and broadside exciters.

Example XII shows the output obtained at 10 kHz for propagation over sea water at night. Whereas Examples XIII and XIV illustrate the results for propagation for the same conditions at 60 kHz. Note that considerably more modes are required to compute the total field at 60 kHz than at 10 kHz.
**Input:**

```
DATUM
BETA=9.5, HPRIME=87.0, SCLHIS=6.0, ENMIN=0.1, FREQ=27.0,
NUFLAG=0, NPUNCH=1, NIGEN=1, RANGE=70.0, 87.0, RANGE1=0.0, 1.0,
AZIM=283.0, CDIP=4.1, MAFLD=5.4E-5, EPSK=15.0, SIGMA=1.0E-2,
&END
```

<table>
<thead>
<tr>
<th>TMESH</th>
<th>0.373</th>
<th>LUB</th>
<th>0.075</th>
</tr>
</thead>
</table>

**Printed Output:**

```
COMPUTEC PROFILE PARAMETERS:

H   NE   NU   X   Y   Z
99.0 2.34E03 6.45E04 2.24E02 5.44E01 3.365E-31 1.0E08
70.0 1.000E-01 4.543E06 1.10E-02 5.44E01 2.675E31 7.005E01

OMEGA-R = 2.5E05 AT TOP HEIGHT = 87.00

REFLECTION HT SET TO 90.64 KM  THETA RANGE = 82.750 TO 67.000
REFLECTION HT SET TO 90.64 KM  THETA RANGE = 78.500 TO 76.750
REFLECTION HT SET TO 90.64 KM  THETA RANGE = 74.250 TO 73.500
REFLECTION HT SET TO 91.64 KM  THETA RANGE = 70.300 TO 74.250

H = 50.00

<table>
<thead>
<tr>
<th>MODE</th>
<th>THEATA</th>
<th>ATTEM</th>
<th>VUVERC</th>
<th>WAIT MAG</th>
<th>WAIT ANG</th>
<th>THETAP</th>
<th>POL</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>69.889</td>
<td>-0.342</td>
<td>2.117</td>
<td>3.994E9</td>
<td>-25.931</td>
<td>0.188</td>
<td>69.762</td>
</tr>
<tr>
<td>2</td>
<td>69.304</td>
<td>-0.167</td>
<td>1.664</td>
<td>0.985E6</td>
<td>-9.350</td>
<td>0.048</td>
<td>89.036</td>
</tr>
<tr>
<td>3</td>
<td>82.790</td>
<td>0.824</td>
<td>6.335</td>
<td>1.601E4</td>
<td>2.350</td>
<td>0.253</td>
<td>86.355</td>
</tr>
<tr>
<td>4</td>
<td>62.790</td>
<td>0.214</td>
<td>4.335</td>
<td>1.301E4</td>
<td>5.048</td>
<td>0.253</td>
<td>86.355</td>
</tr>
<tr>
<td>5</td>
<td>80.304</td>
<td>0.293</td>
<td>4.265</td>
<td>1.065E3</td>
<td>6.406</td>
<td>0.396</td>
<td>83.403</td>
</tr>
<tr>
<td>6</td>
<td>79.520</td>
<td>0.224</td>
<td>5.552</td>
<td>1.038E3</td>
<td>-9.021</td>
<td>0.508</td>
<td>82.146</td>
</tr>
<tr>
<td>7</td>
<td>73.820</td>
<td>0.289</td>
<td>9.300</td>
<td>1.019E3</td>
<td>11.037</td>
<td>0.721</td>
<td>78.320</td>
</tr>
<tr>
<td>8</td>
<td>73.005</td>
<td>0.340</td>
<td>7.290</td>
<td>1.034E3</td>
<td>-17.624</td>
<td>0.407</td>
<td>77.956</td>
</tr>
<tr>
<td>9</td>
<td>73.277</td>
<td>0.224</td>
<td>5.580</td>
<td>1.055E3</td>
<td>1.900</td>
<td>0.407</td>
<td>74.958</td>
</tr>
<tr>
<td>10</td>
<td>72.312</td>
<td>-0.507</td>
<td>13.330</td>
<td>1.041E3</td>
<td>-5.372</td>
<td>0.236</td>
<td>73.792</td>
</tr>
</tbody>
</table>

D = 70.64

<table>
<thead>
<tr>
<th>LIGEN</th>
<th>ATTEM</th>
<th>VUVERC</th>
</tr>
</thead>
<tbody>
<tr>
<td>85.889</td>
<td>-0.342</td>
<td>2.117</td>
</tr>
<tr>
<td>83.364</td>
<td>-0.167</td>
<td>1.664</td>
</tr>
<tr>
<td>82.750</td>
<td>-0.214</td>
<td>2.235</td>
</tr>
<tr>
<td>82.750</td>
<td>-0.214</td>
<td>2.235</td>
</tr>
<tr>
<td>80.004</td>
<td>-0.253</td>
<td>4.265</td>
</tr>
<tr>
<td>79.520</td>
<td>-0.224</td>
<td>3.528</td>
</tr>
<tr>
<td>76.320</td>
<td>-0.269</td>
<td>5.300</td>
</tr>
<tr>
<td>76.269</td>
<td>0.049</td>
<td>7.298</td>
</tr>
<tr>
<td>72.777</td>
<td>-0.244</td>
<td>8.500</td>
</tr>
<tr>
<td>72.312</td>
<td>-0.507</td>
<td>13.330</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>1.0413</td>
</tr>
</tbody>
</table>

**Example I**

TIME REQUIRED FOR THIS PROFILE = 83.20 SECONDS
<table>
<thead>
<tr>
<th>R</th>
<th>0.2</th>
<th>F</th>
<th>27.2202</th>
<th>A</th>
<th>283.2226</th>
<th>E</th>
<th>21.0292</th>
<th>M</th>
<th>5,499E-05</th>
<th>S</th>
<th>1,271E-22</th>
<th>E</th>
<th>15.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>89.76163</td>
<td>-5.921301</td>
<td>5.3134163E-04</td>
<td>1.7315403E-07</td>
<td>-2.943112205E-10</td>
<td>5.263712533E-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>89.60211</td>
<td>-2.7922252</td>
<td>8.4369767E-03</td>
<td>1.333735534E-07</td>
<td>5.7034333E-09</td>
<td>2.8308286E-09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>89.55392</td>
<td>-1.079442</td>
<td>8.313931495E-03</td>
<td>1.48095814E-07</td>
<td>8.40600E-09</td>
<td>2.2165471E-09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>89.55002</td>
<td>-1.078442</td>
<td>8.313931495E-03</td>
<td>1.48095814E-07</td>
<td>8.40600E-09</td>
<td>2.2165471E-09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>89.55502</td>
<td>-1.079452</td>
<td>3.89146036E-03</td>
<td>1.48095814E-07</td>
<td>8.40600E-09</td>
<td>2.2165471E-09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>89.55502</td>
<td>-1.079452</td>
<td>3.89146036E-03</td>
<td>1.48095814E-07</td>
<td>8.40600E-09</td>
<td>2.2165471E-09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>89.55502</td>
<td>-1.079452</td>
<td>3.89146036E-03</td>
<td>1.48095814E-07</td>
<td>8.40600E-09</td>
<td>2.2165471E-09</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

\[
\begin{align*}
\Theta_r & \quad \Theta'_r & \quad T_r & \quad T'_r & \quad T_r & \quad T'_r & \quad T_r & \quad T'_r & \quad T_r & \quad T'_r
\end{align*}
\]

\[D = 70.64,\]
\[F\text{IGEN} = \]
\[35.199, -23.342,\]
\[33.199, -23.342,\]
\[32.199, -23.342,\]
\[31.199, -23.342,\]
\[30.199, -23.342,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]
\[7.219, -0.224,\]

**EXAMPLE II**
PRINTED OUTPUT:

TMESH = 0.373  LUB = 0.0075

WARNING--THE TOP OF THE PROFILE CAN NOT BE CUT OFF. THE TOP OF THE PROFILE MAY BE TO LOW.

TEMPO-B AT TOP = 2.344E 01  WAVES OMEGA-R AT TOP = 7.982E 07
THE BOTTOM PROFILE HEIGHT IS = 39.414 AT B(CUTOFF) = 0.1JOE-03

OMEGA-R EQUALS 2.5E05 AT TOP HEIGHT = 80.98

<table>
<thead>
<tr>
<th>REFLECTION HT SET TO</th>
<th>THETA RANGE</th>
<th>ATTEN</th>
<th>VOHCRC</th>
<th>WAIT MAG</th>
<th>WAIT ANG</th>
<th>THETAP</th>
<th>PUL</th>
</tr>
</thead>
<tbody>
<tr>
<td>82.41 KM</td>
<td>82.750 TO 61.000</td>
<td>3.143</td>
<td>0.9539</td>
<td>-45.101</td>
<td>0.036</td>
<td>69.068</td>
<td>-6.300</td>
</tr>
<tr>
<td>83.41 KM</td>
<td>78.500 TO 62.750</td>
<td>2.148</td>
<td>0.9924</td>
<td>-20.294</td>
<td>0.169</td>
<td>89.739</td>
<td>-5.604</td>
</tr>
<tr>
<td>84.41 KM</td>
<td>74.250 TO 70.500</td>
<td>3.784</td>
<td>0.9991</td>
<td>-4.393</td>
<td>-0.055</td>
<td>88.701</td>
<td>-1.666</td>
</tr>
</tbody>
</table>

TIME REQUIRED FOR THIS PROFILE = 169.08 SECONDS

EXAMPLE VI
Punched Output (NPUNCH=9):

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.12</td>
<td>0.23</td>
<td>0.34</td>
<td>0.45</td>
<td>0.56</td>
<td>0.67</td>
<td>0.78</td>
<td>0.89</td>
<td>0.90</td>
<td>0.11</td>
</tr>
<tr>
<td>2</td>
<td>0.21</td>
<td>0.32</td>
<td>0.43</td>
<td>0.54</td>
<td>0.65</td>
<td>0.76</td>
<td>0.87</td>
<td>0.98</td>
<td>0.09</td>
<td>0.10</td>
</tr>
<tr>
<td>3</td>
<td>0.30</td>
<td>0.41</td>
<td>0.52</td>
<td>0.63</td>
<td>0.74</td>
<td>0.85</td>
<td>0.96</td>
<td>0.07</td>
<td>0.18</td>
<td>0.29</td>
</tr>
<tr>
<td>4</td>
<td>0.40</td>
<td>0.51</td>
<td>0.62</td>
<td>0.73</td>
<td>0.84</td>
<td>0.95</td>
<td>0.06</td>
<td>0.17</td>
<td>0.28</td>
<td>0.39</td>
</tr>
<tr>
<td>5</td>
<td>0.50</td>
<td>0.61</td>
<td>0.72</td>
<td>0.83</td>
<td>0.94</td>
<td>0.05</td>
<td>0.16</td>
<td>0.27</td>
<td>0.38</td>
<td>0.49</td>
</tr>
<tr>
<td>6</td>
<td>0.60</td>
<td>0.71</td>
<td>0.82</td>
<td>0.93</td>
<td>0.04</td>
<td>0.15</td>
<td>0.26</td>
<td>0.37</td>
<td>0.48</td>
<td>0.59</td>
</tr>
<tr>
<td>7</td>
<td>0.70</td>
<td>0.81</td>
<td>0.92</td>
<td>0.03</td>
<td>0.14</td>
<td>0.25</td>
<td>0.36</td>
<td>0.47</td>
<td>0.58</td>
<td>0.69</td>
</tr>
<tr>
<td>8</td>
<td>0.80</td>
<td>0.91</td>
<td>0.02</td>
<td>0.13</td>
<td>0.24</td>
<td>0.35</td>
<td>0.46</td>
<td>0.57</td>
<td>0.68</td>
<td>0.79</td>
</tr>
<tr>
<td>9</td>
<td>0.90</td>
<td>0.01</td>
<td>0.12</td>
<td>0.23</td>
<td>0.34</td>
<td>0.45</td>
<td>0.56</td>
<td>0.67</td>
<td>0.78</td>
<td>0.89</td>
</tr>
</tbody>
</table>

Example VII
### PROFILE

<table>
<thead>
<tr>
<th>NELCNU</th>
<th>NITEF</th>
<th>SEC</th>
<th>MM</th>
</tr>
</thead>
<tbody>
<tr>
<td>100.00</td>
<td>9.74E 02</td>
<td>1.04E 03</td>
<td></td>
</tr>
<tr>
<td>95.00</td>
<td>6.4E 02</td>
<td>6.65E 02</td>
<td></td>
</tr>
<tr>
<td>90.00</td>
<td>3.49E 02</td>
<td>4.39E 02</td>
<td></td>
</tr>
<tr>
<td>85.00</td>
<td>1.90E 02</td>
<td>3.63E 02</td>
<td></td>
</tr>
<tr>
<td>80.00</td>
<td>5.10E 01</td>
<td>4.15E 02</td>
<td></td>
</tr>
<tr>
<td>75.00</td>
<td>3.80E 00</td>
<td>4.47E 02</td>
<td></td>
</tr>
<tr>
<td>70.00</td>
<td>3.40E 02</td>
<td>4.50E 02</td>
<td></td>
</tr>
<tr>
<td>65.00</td>
<td>1.14E 02</td>
<td>5.60E 02</td>
<td></td>
</tr>
<tr>
<td>60.00</td>
<td>2.77E 03</td>
<td>7.65E 02</td>
<td></td>
</tr>
<tr>
<td>55.00</td>
<td>3.60E 03</td>
<td>1.09E 03</td>
<td></td>
</tr>
<tr>
<td>50.00</td>
<td>2.06E 03</td>
<td>1.54E 03</td>
<td></td>
</tr>
<tr>
<td>45.00</td>
<td>1.21E 03</td>
<td>2.11E 03</td>
<td></td>
</tr>
<tr>
<td>40.00</td>
<td>6.73E 04</td>
<td>2.73E 03</td>
<td></td>
</tr>
<tr>
<td>35.00</td>
<td>3.66E 04</td>
<td>3.25E 03</td>
<td></td>
</tr>
<tr>
<td>30.00</td>
<td>1.67E 04</td>
<td>3.53E 03</td>
<td></td>
</tr>
<tr>
<td>25.00</td>
<td>7.70E 05</td>
<td>3.63E 03</td>
<td></td>
</tr>
<tr>
<td>20.00</td>
<td>3.43E 05</td>
<td>3.63E 03</td>
<td></td>
</tr>
<tr>
<td>15.00</td>
<td>1.30E 05</td>
<td>3.63E 03</td>
<td></td>
</tr>
<tr>
<td>10.00</td>
<td>6.58E 06</td>
<td>3.72E 03</td>
<td></td>
</tr>
<tr>
<td>5.00</td>
<td>3.43E 06</td>
<td>4.42E 03</td>
<td></td>
</tr>
<tr>
<td>0.0</td>
<td>2.17E 06</td>
<td>5.50E 03</td>
<td></td>
</tr>
</tbody>
</table>

**INPUT:**

```
TYPEN=100.0,
AZIM=283.0,COORDP=21.0,MAGFLD=5.44E-5,
FREQ=27.0,EPSP=15.0,SIGMA=A.0E-2,;UFLAG=1,NPUNCH=9,
RANGER=70.0,87.0,RANGEI=0.0,-1.0,
END
```

**TMESH = 0.373 LUB = 0.0075**

**EDITED OUTPUT (PARTIAL):**

*THE TOP OF THE PROFILE IS SET TO BE--*

89.10 3.28E 02 4.24E 02 8.24E 01

*AT THE TOP OF THE PROFILE B = 1.724E 01 OMEGA-R = 4.584E 06*

*THE BOTTOM PROFILE HEIGHT IS = 39.414 AT B(CUTOFF) = 0.100E-03*

**EXAMPLE VIII**
### PROFILE 3

**SATELLITE MIGHT AMBIENT PROFILE**

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>Pres. (Pa)</th>
<th>Temp. (K)</th>
<th>Viscosity (Pa·s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>60.00</td>
<td>1.00E+05</td>
<td>1.00E+05</td>
<td>1.00E+05</td>
</tr>
<tr>
<td>120.00</td>
<td>2.00E+05</td>
<td>2.00E+05</td>
<td>2.00E+05</td>
</tr>
<tr>
<td>180.00</td>
<td>3.00E+05</td>
<td>3.00E+05</td>
<td>3.00E+05</td>
</tr>
<tr>
<td>240.00</td>
<td>4.00E+05</td>
<td>4.00E+05</td>
<td>4.00E+05</td>
</tr>
<tr>
<td>300.00</td>
<td>5.00E+05</td>
<td>5.00E+05</td>
<td>5.00E+05</td>
</tr>
</tbody>
</table>

**INPUT:**

```
C60 FREQ 3
20.00 1.05E-02 4.50E-01 4.50E-01
25.00 3.50E-01 9.00E-01 9.00E-01
30.00 7.50E-01 1.50E-00 1.50E-00
35.00 1.25E-00 2.10E-00 2.10E-00
40.00 1.75E-00 2.60E-00 2.60E-00
45.00 2.25E-00 3.20E-00 3.20E-00
50.00 2.75E-00 3.80E-00 3.80E-00
55.00 3.25E-00 4.40E-00 4.40E-00
60.00 3.75E-00 5.00E-00 5.00E-00
65.00 4.25E-00 5.60E-00 5.60E-00
70.00 4.75E-00 6.20E-00 6.20E-00
75.00 5.25E-00 6.80E-00 6.80E-00
80.00 5.75E-00 7.40E-00 7.40E-00
85.00 6.25E-00 8.00E-00 8.00E-00
90.00 6.75E-00 8.60E-00 8.60E-00
95.00 7.25E-00 9.20E-00 9.20E-00
100.00 7.75E-00 9.80E-00 9.80E-00
```

**EXAMPLE IX**
PRINTED OUTPUT:

\[ \text{TMESH} = 7.371 \quad \text{LUB} = 9.1414 \]

\[
\begin{array}{cccccccc}
\text{NUDE} & \text{THETA} & \text{ATTEN} & \text{V0VERC} & \text{WAIT MAG} & \text{WAIT ANG} & \text{THETAP} & \text{PUL} \\
1 & 83.117 & 33.751 & 1.029 & 0.84794 & -5.030 & -0.150 & 83.246 & -34.507 & 4.92227 \\
\end{array}
\]

\[
\text{TIME REQUIRED FOR THIS PROFILE} = 60.56 \text{ SECONDS}
\]

EXAMPLE X

PUNCHED OUTPUT (NPUNCH=7):

\[
\begin{array}{cccccccc}
\text{XVZ} & \text{PHI}_{XVZ} & \text{XEZ} & \text{PHI}_{XEZ} & \text{XBZ} & \text{PHI}_{XBZ} \\
\end{array}
\]

EXAMPLE XI
INPUT:

PRINTED OUTPUT:

TIME REQUIRED FOR THIS PROFILE = 13.041 SECONDS

EXAMPLE XII
INPUT:

HAWAII TO CALIFORNIA (NIGHTIME)
LATUM
A/I=-58.0, PUMEP=34.0, MAGFLU=4.31E-5,
BETA=3.0, HPXME=3.0, SCLHTS=6.0,
FREQ=60.0.
RANGE=60.3, 3.3, RANGEI=0.0, -2.0,
END

PRINTED OUTPUT (PARTIAL):

TMESH = 0.2,00 LUR = 0.0000
COMPUTED PROFILE PARAMETERS:

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>Z</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.01E+02</td>
<td>1.01E+02</td>
<td>1.01E+02</td>
</tr>
</tbody>
</table>

OMEGA-R = 2.0E05 AT TOP HEIGHT = 68.00

REFLECTION HT SET TO 69.42 KM
THETA RANGE = 85.000 TO 90.000

REFLECTION HT SET TO 69.42 KM
THETA RANGE = 80.000 TO 90.000

EXAMPLE XIII
E. Error Messages

(1) EIGENANGLES AT $\theta_1$ AND $\theta_2$ ARE TO BE RESOLVED (program execution stops).

Solution: Assign to RANGER and RANGEI values which constitute a small "contour rectangle" which incloses the above two eigenangles and re-execute the program. An alternative is to decrease 'GMAX'.

(2) ERROR IN CONTROL CARD (Program execution stops).

Solution: Check control cards (i.e., PROFILE, COLFREQ, &DATUM, or &END)

(3) TOP OF PROFILE CANNOT BE SET -- At the bottom of the PROFILE, B is greater than the value of TEMPO-B (program execution stops).

Solution: Problem is due to incompatibilities between values of TEMPO-B and the actual electron and ion density values.

(4) WARNING -- THE TOP OF THE PROFILE CANNOT BE CUTOFF. THE TOP OF THE PROFILE MAY BE TOO LOW (program execution continues).

Reason: In top card of electron density profile the computed B value is less than the input value of TEMPO-B.

(5) THE PROFILE BOTTOM CANNOT BE CUTOFF. ALL B-VALUES (AS A FUNCTION OF HEIGHT) ARE LESS THAN CUTOFF. THIS IMPLIES THAT THE PROFILE IS VERY CLOSE TO FREE SPACE (Program execution stops).

Solution: Check values of 'CUTOFF', and individual values of electron and ion densities. If o.k., put 'ICUT' = 0 and re-execute.
(6) THE BOTTOM OF THE PROFILE CANNOT BE CUTOFF USING 'CUTOFF' = XXX AT HEIGHT = YYY BECAUSE THE VALUE OF B AT THE BOTTOM OF THE PROFILE IS LARGER THAN 'CUTOFF' (Program execution continues).

(7) NO MODES (program execution continues).
No eigenangle solutions located inside the contour rectangle as chosen. Solutions: Change RANGER and RANGEI values and re-execute the program.

(8) INVALID MODE AT θ (program execution continues).
Solution: Cut down the size of the contour rectangle in the region of θ.

(9) STEP TOO SMALL IN INTEG OR TOO MANY STEPS IN INTEG (program execution stops).
Solution: Examine PROFILE (e.g. too dense at top or too sparse at bottom) and other ionospheric parameters such as magnetic field, collision frequency, etc. First decrease 'GMAX' and re-execute the program. If problem persists, increase 'ENMIN' and again re-execute.

(10) PROBLEM IN SORTINC Q VALUES OR Q FAILS TO CONVERGE IN QUARTIC (program execution stops).
Solution: Again look at ionospheric parameter values. RANGER (minimum) is probably too small. Increase the value of RANGER (minimum) and re-execute program.

(11) X MODES FOUND ON SAME PHASE LINE (program execution continues).
Solution: Information only.
(12) WARNING X MODES FOUND ON Y PHASE LINES (program execution continues).

Solution: Information only.

(13) If run time is very long it is probably related to #8 above and thus solution is the same as for #8.

(14) NO EXIT FROM MESH SQUARE (program execution stops)

Solution: The value of TMESH should be changed slightly (1%) and the program rerun. If message continues - STOP.

(15) ONLY X MODES FOUND ON Y PHASE LINES (program execution stops).

Solution: Reduce TMESH by 1/2 and re-execute program.

(16) PROBLEMS IN SUBROUTINE NO MESH (program execution stops).

Solutions: Reduce TMESH by 1/2 and re-execute program. If message continues - STOP.
VI. REFERENCES


13. Moler, W., Naval Electronics Laboratory Center, San Diego, CA, private communication.

APPENDIX A

Modification of the Modal Equation

Given:

\[ F_1(\theta) = \left( 1 - \frac{\pi N_{II}}{\pi D_I} \right) \left( 1 - \frac{1}{\pi D_{II}} \right) - \pi R_{II} \cdot \pi R_{II} \cdot \frac{\pi N_{II}}{\pi D_{II}} \cdot \frac{1}{\pi D_{II}} \]  \hspace{1cm} (A-1)

where \( R = R(\theta) \) and \( \bar{R} = \bar{R}(\theta) \).

Let

\[ y_{R_{II}} = \frac{\pi N_{II}}{\pi D_I} \cdot \bar{R}_{II} = \frac{\pi N_{II}}{\pi D_{II}} \]  \hspace{1cm} (A-2)

where \( \pi N_{II} \), \( \pi D_{II} \), \( \pi N_{II} \) and \( \pi D_{II} \) are in such form that they contain no poles in the region of interest in the \( \theta \)-plane.

Then

\[ F_1(\theta) = \left( 1 - \frac{\pi N_{II}}{\pi D_I} \right) \left( 1 - \frac{1}{\pi D_{II}} \right) - \pi R_{II} \cdot \pi R_{II} \cdot \frac{\pi N_{II}}{\pi D_{II}} \cdot \frac{1}{\pi D_{II}} \]  \hspace{1cm} (A-3)

and

\[ F_2(\theta) = \pi D_{II} \cdot \pi D_{II} F_1(\theta) = \left( \pi D_{II} - \pi N_{II} \right) \left( \pi D_{II} - \pi N_{II} \right) \]  \hspace{1cm} (A-4)

Also

\[ F_2(\theta) = \begin{bmatrix} y_{R_{II}} - y_{N_{II}} C \left( \frac{R_{II} + 1}{C} \right) + y_{N_{II}} \end{bmatrix} \]  \hspace{1cm} (A-5)

\[ \left[ y_{D_{II}} - y_{N_{II}} C \left( \frac{R_{II} + 1}{C} \right) + y_{N_{II}} \right] \]  \hspace{1cm} (A-5)

\[ \cdot \left[ C^2 \frac{R_{II}}{C} \cdot \frac{R_{II}}{C} \cdot y_{N_{II}} \cdot y_{N_{II}} \right] \]  \hspace{1cm} (A-5)

108
Then

\[ F_3(\theta) = \frac{r_2(\theta)}{c^2} = \frac{r_{D_\perp}r_{D_\parallel}F_1(\theta)}{c^2} = \]

\[
\left\{ \begin{array}{l}
\left[ \frac{1}{n_{\parallel}} + \frac{1}{D_{\parallel}} \right] - \frac{\left( R_{\parallel} + 1 \right)}{c^2} \cdot \left[ \frac{1}{n_{\perp}} + \frac{1}{D_{\perp}} \right] - \frac{\left( R_{\perp} + 1 \right)}{c^2} \\
- \left[ \frac{1}{n_{\parallel}} \cdot \frac{1}{n_{\perp}} \cdot \frac{R_{\parallel}}{c} \cdot \frac{1}{R_{\perp}} \right]
\end{array} \right\} \]  \hspace{1cm} (A-6)

Let

\[ \bar{x} = \frac{R + 1}{c} \]

\[
\left( \frac{1}{n_{\parallel}} + \frac{1}{D_{\parallel}} \right) = \frac{1}{n_{\parallel}} \left( \frac{1}{n_{\perp}} + \frac{1}{D_{\perp}} \right) = \frac{1}{n_{\perp}}
\]  \hspace{1cm} (A-7)

\[ i_{n_{\parallel}} = i_{d_{\parallel}} \quad i_{n_{\perp}} = i_{d_{\perp}} \]

Then

\[ F_3(\theta) = \frac{r_{D_\parallel}r_{D_\perp}F_1(\theta)}{c^2} = \]

\[
\left[ \frac{1}{n_{\parallel}} \cdot \frac{1}{n_{\perp}} \cdot \frac{R_{\parallel}}{c} \cdot \frac{1}{R_{\perp}} \right] - \left[ \frac{1}{n_{\parallel}} \cdot \frac{1}{n_{\perp}} \cdot \bar{x}_{\parallel} \right] - \left[ \frac{1}{n_{\perp}} \cdot \frac{1}{n_{\perp}} \cdot \bar{x}_{\perp} \right] - \left[ \frac{1}{n_{\perp}} \cdot \frac{1}{n_{\perp}} \cdot \bar{x}_{\perp} \right] \]

\hspace{1cm} (A-8)

Note that:

\[ \frac{n_{\parallel}}{d_{\parallel}} = \frac{n_{\parallel} + D_{\parallel}}{c} = \frac{n_{\parallel}}{n_{\parallel}} + 1 = \left( \frac{1}{\bar{R}_{\parallel}} + 1 \right) \frac{1}{\bar{c}} = \frac{R_{\parallel} + 1}{\bar{c}} \] \hspace{1cm} (A-9)

\[ \frac{n_{\perp}}{d_{\perp}} = \frac{n_{\perp} + D_{\perp}}{c} = \frac{n_{\perp}}{n_{\perp}} + 1 = \left( \frac{1}{\bar{R}_{\perp}} + 1 \right) \frac{1}{\bar{c}} = \frac{R_{\perp} + 1}{\bar{c}} \] \hspace{1cm} (A-10)

Also note that no poles in \( N \) or \( D \) implies that there are no poles in \( n \) or \( d \).
APPENDIX B

Derivation of the Equations for Free Space Integration

In the MODESRCH routine the differential equations for the ionospheric and ground reflection coefficient matrices have to be integrated through a free space region over a curved earth. In the following derivation of the necessary differential equations for the ionospheric reflection matrices a number of references are made to Budden's work, reference 7.

A right handed Cartesian system is assumed with the X and Y axes lying in a horizontal plane boundary between the bottom of the ionosphere and the free space below. The positive Z direction is upward. A plane wave is incident upon the ionosphere from below with the wave normal in the X-Z plane at an angle $\theta_1$ to the Z axis.

The differential equations describing wave fields at oblique incidence (Op. cit., P. 140, eqns. 9.49-9.54) may be separated into two independent sets. Either of these sets of field variables may be set equal to zero without affecting the other, so that the corresponding waves are propagated independently.

For the first set the electric field is everywhere parallel to the Y-axis, and the waves are said to be 'horizontally polarized'. The equations are:

$$\frac{\partial E_y}{\partial z} = i k H_x \quad (B-1)$$
$$i k S E_y = -i k H_z \quad (B-2)$$
$$\frac{\partial H_x}{\partial z} + i k S H_z = i k n^2 E_y \quad (B-3)$$
where \( i = \sqrt{-1} \)
\[ S = \sin \theta_1 \]
\[ k = \frac{2 \pi}{\lambda} \]
\[ \lambda = \text{free space wavelength} \]
\[ n = \text{index of refraction of the medium} \]

and \( H = \mathcal{Z}_o H \)
where \( \mathcal{Z}_o = E_x/H_y \)

\( H_z \) may be eliminated from eqns. 2 and 3 to give
\[ \frac{\partial H_x}{\partial Z} = i k (n^2 - S^2) E_y \quad (B-4) \]

Elimination of \( H_x \) between eqns. 1 and 4 gives
\[ \frac{d^2 E_y}{dZ^2} + k^2 q^2 E_y = 0 \quad (B-5) \]
where \( q^2 = (n^2 - S^2) \)

For the second set of equations the electric field is parallel to the X-Z plane and the waves are said to be 'vertically polarized' even though the electric vector is not, in general, vertical.

The equations are:
\[ \frac{\partial E_x}{\partial Z} + i k S E_z = -i k H_y \quad (B-6) \]
\[ \frac{\partial H_y}{\partial Z} = -i k n^2 E_x \quad (B-7) \]
\[ -i k S H_y = i k n^2 E_z \quad (B-8) \]

The field component \( E_z \) may be eliminated from eqns. (6) and (8) to give
\[ \frac{\partial E_x}{dZ} = -i k \frac{q^2}{n^2} H_y \quad (B-9) \]
Elimination of $E_x$ between eqns. 7 and 9 gives

$$\frac{d^2 H_y}{dz^2} - \frac{1}{n^2} \frac{d}{dz} \left( \frac{d(n^2)}{dz} \right) H_y + k^2 q^2 H_y$$  \hspace{1cm} (B-10)$$

Just below the free space-ionosphere boundary the total electric
wave field is composed of two upward traveling incident components and two
downward traveling reflected components. They are $E_{\parallel,y}^i$ and $E_{\parallel,y}^r$ respectively, where $E_{\parallel}$ is in the X-Y plane. Following Budden (Op Cit. Pg. 118
eq. 8.72) the equations for free space integration of reflection at
oblique incidence may be derived.

The total field components just below the boundary are

$$E_x = (E_{\parallel,y}^i - E_{\parallel,y}^r) \cos \theta_I$$  \hspace{1cm} (B-11,a)$$

$$E_y = E_{\parallel,y}^i + E_{\parallel,y}^r$$  \hspace{1cm} (B-11,b)$$

$$H_x = (E_{\parallel,y}^r - E_{\parallel,y}^i) \cos \theta_I$$  \hspace{1cm} (B-11,c)$$

$$H_y = E_{\parallel,y}^i + E_{\parallel,y}^r$$  \hspace{1cm} (B-11,d)$$

Define:

$$R_{\perp} = E_{\parallel,y}^r/E_{\parallel,y}^i$$  \hspace{1cm} (B-12,a)$$

$$R_{\parallel} = E_{\parallel,y}^r/E_{\parallel,y}^i$$  \hspace{1cm} (B-12,b)$$

$$R_{\parallel} = E_{\parallel,y}^r/E_{\parallel,y}^i$$  \hspace{1cm} (B-12,c)$$

$$R_{\perp} = E_{\parallel,y}^r/E_{\parallel,y}^i$$  \hspace{1cm} (B-12,d)$$

where the $R$'s are functions of $\theta_I$.

For Horizontal Polarization:

Combine eqns. (11,b), (12,a) and (12,b) to get

$$E_y = E_y^i + E_y^i R_{\perp} + E_{\parallel,y}^i R_{\parallel}$$
or

\[ E_y = E_y^i (R_1 + 1) + E_{||}^i R_1 \]  \hspace{1cm} (B-13)

The wave equation for Horizontal Polarization (e.g., eq 5) is:

\[ \frac{d^2 E_y}{dz^2} + k^2 \frac{q^2}{2} E_y = 0 \]  \hspace{1cm} (B-5)

where \( q^2 = n^2 - S^2 \)

and \( S = \sin \theta_1 \)

To include a curved earth let

\[ n^2 = 1 + \alpha (Z-H) \]

where:

- \( n \) is the index of refraction of the medium,
- \( \alpha = 2/r_e \), where \( r_e \) is the radius of the earth.
- \( H \) is that height where \( n = 1 \).

then \( q^2 = 1 + \alpha (Z-H) - S^2 = c^2 + \alpha (Z-H) \)

where \( c = \cos \theta_1 \)

so that

\[ \frac{d^2 E_y}{dz^2} + k^2 \left[ c^2 + \alpha (Z-H) \right] E_y = 0 \]  \hspace{1cm} (B-14)

to get standard form:

Let \( \rho = \left( \frac{k_1}{\alpha} \right)^{2/3} \left[ c^2 + \alpha (Z-H) \right] \)  \hspace{1cm} (B-15)

then

\[ \frac{d\rho}{dz} = \left( \frac{k_1}{\alpha} \right)^{2/3} \alpha = \left( \nu^2 \alpha \right)^{1/3} \]

\[ \frac{d^2 \rho}{dz^2} = 0 \]
and then expanding gives

\[ \frac{d^2 E_y}{dZ^2} = \frac{d}{dZ} \left( \frac{d E_y}{dZ} \right) = \frac{d}{dZ} \left( \frac{d E_y}{d\rho} \frac{d\rho}{dZ} \right) \]

\[ = \frac{d^2 E_y}{d\rho^2} \left( \frac{d\rho}{dZ} \right)^2 + \frac{d E_y}{d\rho} \frac{d^2 \rho}{dZ^2} \]

now, substituting the above results into eq. B-14 gives

\[ \frac{d^2 E_y}{dZ^2} = (k^2 \alpha) \frac{2}{3} \frac{d^2 E_y}{d\rho^2} \]  

(B-16)

Also, since

\[ (k^2 \alpha)^{2/3} \left( \frac{k}{\alpha} \right)^{2/3} = k^2 \]

then

\[ k^2 q^2 = (k^2 \alpha)^{2/3} \left( \frac{k}{\alpha} \right)^{2/3} \left[ c^2 + \alpha (z-h) \right] \]

or

\[ k^2 q^2 = (k^2 \alpha)^{2/3} \rho \]

so that the wave equation (eq. 5) becomes upon substitution of the above:

\[ (k^2 \alpha)^{2/3} \left\{ \frac{d^2 E_y}{d\rho^2} + \rho \frac{d E_y}{d\rho} \right\} = 0 \]

or

\[ \frac{d^2 E_y}{d\rho^2} + \rho \frac{d E_y}{d\rho} = 0 \]  

(B-17)

The general solution to this equation is

\[ E_y(\rho) = A h_1(\rho) + B h_2(\rho) \]  

(B-18)

where A and B are arbitrary constants and \( h_1(\rho) \) and \( h_2(\rho) \) are modified Hankel functions of order 1/3. (See reference 8).
To relate $E_y$ and $H_x$:

Equation (1) is:

$$\frac{d E_y}{d z} = i \kappa H_x$$

or

$$H_x = \frac{-i}{k} \frac{d E_y}{d z}$$

$$= \frac{i}{k} \frac{d E_y}{d \rho} \frac{d \rho}{d z} = \frac{-i}{k} \left( k^2 \alpha \right)^{1/3} \frac{d E_y}{d \rho}$$

$$H_x = -k \left( \frac{\alpha}{k} \right)^{1/3} \frac{d E_y}{d \rho}$$

let $K = i \left( \frac{\alpha}{k} \right)^{1/3}$  \hspace{1cm} (B-19)

then $H_x = -K \frac{d E_y}{d \rho}$

or

$$H_x = -K \left( A h_1^i (\rho) + B h_2^i (\rho) \right)$$  \hspace{1cm} (B-20)

The primes on the above quantities denote derivatives with respect to the argument.

From Eq. 13,

$$E_y = C E_y^i \left( \frac{i R_i}{C} + 1 \right) + C E_i^i \left( \frac{i R_i}{C} \right) \hspace{1cm} (B-21)$$

and from eq. 11c

$$H_x = CE_y^r - CE_y^i$$

also from eq. 11,b

$$E_y^r = E_y - E_y^i$$
Substitution for $E_y^r$ into eq. (11,c) gives

$$C E_y - C E_y^i = H_x + C E_y^i$$

or

$$C E_y - H_x = 2C E_y^i \quad \text{(B-22)}$$

Substitutions into eq. 21 for $E_y$ from eq. 18 gives

$$A h_1(p) + B h_2(p) = C E_y^i \left( \frac{\vec{R}_1^i + \vec{R}_1^i}{C} \right) + C E_y^i \left( \frac{\vec{R}_1^i}{C} \right)$$

and substitution into eq. 22 for $E_y$ and $H_x$ from equations 18 and 20 gives

$$C \{ A h_1(p) + B h_2(p) \} + K \{ A h_1'(p) + B h_2'(p) \} = 2C E_y^i$$

or

$$\begin{align*}
(- \frac{A}{C}) h_1(p) + (- \frac{B}{C}) h_2(p) + E_y^i \left( \frac{\vec{R}_1^i + \vec{R}_1^i}{C} \right) + E_y^i \left( \frac{\vec{R}_1^i}{C} \right) &= 0 \\
(- \frac{A}{C})(Ch_1(p) + Kh_1'(p)) + (- \frac{B}{C})(Ch_2(p) + Kh_2'(p)) + 2E_y^i &= 0
\end{align*} \quad \text{(B-23)}$$

Equations 23 and 24 are the equations to be used for 'horizontal polarization' in the free space integration procedure.

For Vertical Polarization:

Applying previous equations (e.g., equations 11 and 12) gives:

$$H_y = E_y^i + E_y^r \quad \text{(B-11,d)}$$

$$E_x = (E_y^i - E_y^r) \cos \theta$$

$$\|R_{||} = E_y^r / E_y^i \quad \text{(B-12,c)}$$

$$\bot R_{||} = E_y^r / E_y^i \quad \text{(B-12,d)}$$
Combine eqns. (11,d), (12,c) and (12,d) to get
\[ H_y = E_{\parallel}^i + E_{\parallel}^i R_{\parallel} + E_{\perp}^i R_{\perp} \]
also combining eqns. (11,a), (12,c) and (12,d)
\[ E_x = (E_{\parallel}^i - E_{\parallel}^i R_{\parallel} - E_{\perp}^i R_{\perp}) C \]

In different form:
\[ H_y = C E_{\parallel}^i \left( \frac{R_{\parallel}+1}{C} \right) + C E_{\perp}^i \left( \frac{R_{\perp}}{C} \right) \]  \hspace{1cm} (B-25)

where \( C = \cos \Theta_{\parallel} \)

and from eqns. (11,a) and (11,d)
\[ E_x = E_{\parallel}^i C - H_y C + E_{\perp}^i C \]
or
\[ C H_y + E_x = 2E_{\parallel}^i C \]  \hspace{1cm} (B-26)

To obtain the wave equation for vertical polarization, follow Budden (op. cit. pg. 343). Note that in this case the electric vector has both vertical (\( E_z \)) and horizontal (\( E_y \)) components.

The wave equation for vertical polarization is given by equation 10 as:
\[ \frac{d^2 H_y}{dz^2} - \frac{1}{n^2} \frac{d}{dz} \left( \frac{n^2}{dH_y/dz} \right) + k^2 q^2 H_y = 0 \]  \hspace{1cm} (B-10)

Any linear second-order differential equation with a term containing the first derivative (e.g., \( dH_y/dz \)) can be reduced to its 'normal form', that is a form without a first derivative term, by a change in the dependent variable.

Let \( V = H_y/n \) \hspace{1cm} (B-27)
or \( H_y = nV \) \hspace{1cm} (B-28)
then equation 10 becomes:

\[
\frac{d^2V}{dz^2} + k^2 \left[ q^2 + \frac{1}{2k^2} \frac{d^2(n^2)}{dz^2} - \frac{3}{4k^2 n^4} \left( \frac{d(n^2)}{dz} \right)^2 \right] V = 0 \quad (B-29)
\]

where for a curved earth:

\[ q^2 = c^2 + \alpha(Z-H) \]

Note that:

\[
\frac{d(n^2)}{dz} = \frac{d}{dz} (1 + \alpha(Z-H)) = \alpha \quad (B-30)
\]

and

\[
\frac{d^2(n^2)}{dz} = 0 \quad (B-31)
\]

The wave equation, Eq. 29, then becomes, after substitution of eqns. 30, and 31:

\[
\frac{d^2V}{dz^2} + k^2 \left[ c^2 + \alpha(Z-H) - \frac{3}{4k^2 n^4} \right] V = 0 \quad (B-32)
\]

let

\[ \rho = \left( \frac{k}{\alpha} \right)^{2/3} [c^2 + \alpha(Z-H)] \quad (B-33) \]

Compare the resulting form of eq. 32 with eq. 17

Therefore the solution is

\[ V = Q h_1(\rho) + G h_2(\rho) \quad (B-34) \]

and since

\[
H_y = nV \quad (B-28)
\]

\[
H_y = n (Q h_1(\rho) + G h_2(\rho)) \quad (B-35)
\]
Note:
\[
\frac{dn^2}{dz} = 2 n \frac{dn}{dz} \quad \text{or} \quad \frac{dn}{dz} = \frac{1}{2n} \frac{dn^2}{dz}
\]

but from eq. 30
\[
\frac{dn^2}{dz} = \alpha
\]

\[
\therefore \frac{dn}{dz} = \frac{\alpha}{2n}
\]

(B-36)

Also from eq. 33,
\[
\frac{d\alpha}{dz} = \left(\frac{k}{\alpha}\right)^{2/3} \quad \alpha = \left(\frac{k^2\alpha^3}{\alpha^2}\right)^{1/3} = (k^2\alpha)^{1/3}
\]

(B-37)

To relate \(H_y\) and \(E_x\):

From eq. (7):
\[
\frac{d}{dz} H_y = -i k n^2 E_x
\]

or
\[
E_x = \frac{i}{k n^2} \frac{d}{dz} H_y = \frac{i}{k n^2} \frac{d}{dz} (nV)
\]

or
\[
E_x = \frac{i}{k n^2} \left(n \frac{dV}{dz} + \frac{dn}{dz} V\right)
\]

and
\[
E_x = -\frac{i}{k n^2} \left(n \frac{dV}{dp} \frac{dp}{dz} + \frac{dn}{dz} V\right)
\]

(B-38)

Substituting eqns. 36 and 37 into eq. 38 gives
\[
E_x = \frac{i}{k n^2} \left(n (k^2\alpha)^{1/3} \frac{dV}{dp} + \frac{\alpha}{2n} V\right)
\]

or
\[
E_x = \frac{1}{n} \left[i \left(\frac{\alpha}{k}\right)^{1/3} \frac{dV}{dp} + i \frac{\alpha}{2k} \frac{V}{n^2}\right]
\]

(B-39)
Now define
\[ K = i \left( \frac{\alpha}{k} \right)^{1/3} \]  \hfill (B-40)
and
\[ L = i \left( \frac{\alpha}{2k} \right) \]  \hfill (B-41)

Then
\[ E_x = \frac{1}{n} \left( K \frac{dv}{dz} + L \frac{V}{n^2} \right) \]  \hfill (B-42)

Since \( L \) is small let \( n^2 \approx 1 \)

Then
\[ \frac{1}{n} E_x = \frac{1}{n^2} \left( K \frac{dv}{dp} + L V \right) \]  \hfill (B-43)

Substituting for \( V \) from Eq. (34) gives:
\[ \frac{1}{n} E_x = \frac{1}{n^2} \left( K \left[ Q h_1' (p) + G h_2' (p) \right] + L \left[ Q h_1(p) + G h_2(p) \right] \right) \]  \hfill (B-44)

Substituting eq. 35 into eq. 25 for \( H_y \) gives

\[ n \left[ Q h_1(p) + G h_2(p) \right] = C E_\parallel i \left( \frac{i R_\parallel + 1}{C} \right) + C E_y i \left( \frac{i R_y}{C} \right) \]  \hfill (B-45)

Substituting eqns. 35 and 44 into eq. (26) for \( H_y \) and \( E_x \) gives:
\[ n \left[ C \left( Q h_1(p) + G h_2(p) \right) \right. \left. \div \frac{1}{n^2} \left( K \left[ Q h_1' (p) + G h_2' (p) \right] \right) + \frac{1}{n^2} \left( L \left[ Q h_1(p) + G h_2(p) \right] \right) \right] = 2E_\parallel i C \]  \hfill (B-46)

Now neglect the multiplier \( n \) so that \( \parallel \) and \( \perp \) will be alike for coupling to get from equation 45:

\[ (- \frac{Q}{C}) h_1(p) + (- \frac{G}{C}) h_2(p) + E_\parallel i \left( \frac{i K_\parallel + 1}{C} \right) + E_y i \left( \frac{i R_\parallel}{C} \right) = 0 \]  \hfill (B-47)
and from equation 46:

\[
\frac{(-q)}{C} \left( C h_1(p) + [K h_1(p) + L h_1(p)]/n^2 \right) + \frac{(-G)}{C} \left( C h_2(p) + [K h_2(p) + L h_2(p)]/n^2 \right) + 2 E_y^i = 0
\]

Equations 47 and 48 are the equations to be used for 'vertical polarization' in the free space integration procedure.

The four equations for ionospheric reflection at oblique incidence (e.g., eqns. 23, 24, 47 and 48) may be used to accomplish the full-wave integration of the four ionospheric reflection coefficients (R) through a free space medium over a curved earth. The integration is to be carried out from some starting height "Zf", where the values of the R's are known, to some other level "Zt" where the values of the R's are to be determined.

The free space integration procedure consists of three steps (I, II and III) which are as follows:

I. At the "from" level (i.e., Z = Zf) the known values of the elements of R may be substituted into equations (23, 24, 47, and 48) and then two sets of boundary conditions, B. C., applied successfully. That is for Set I assign \((E_y^i = 1, E_y^j = 0)\) and solve for the coefficients \(A_J\) and \(B_J\) from equations 23 and 24; and for the coefficients \(Q_J\) and \(G_J\) from equations 47 and 48. Next, for Set II assign \((E_y^i = 1, E_y^j = 0)\) and solve for the coefficients \(A_{II}\) and \(B_{II}\) from equations 23 and 24; and for the coefficients \(Q_{II}\) and \(G_{II}\) from equations 47 and 48.

The procedure includes a need for solving two simultaneous equations. That is if:

\[
a_{11} A + a_{12} B = V_1 \quad (B-49,a)
\]

\[
a_{21} A + a_{22} B = V_2 \quad (B-49,b)
\]
then
\[
\bar{A} = \frac{(V_1 a_{22} - V_2 a_{12})}{(a_{11} a_{22} - a_{21} a_{12})} \quad (B-50,a)
\]
\[
\bar{B} = \frac{(V_2 a_{11} - V_1 a_{21})}{(a_{11} a_{22} - a_{21} a_{12})} \quad (B-50,b)
\]

For Horizontal Polarization at the \( Z_f \) (from) level, the following identifications are made:

Identify terms of eq. 23 with eq. (49,a) and terms of eq. 24 with (49,b). That is let:
\[
\bar{A} = -\frac{A}{C}, \quad \bar{B} = -\frac{B}{C};
\]

where
\[
C = \cos \theta_1, \quad \rho_f: \text{ implies a function of } o \text{ at the "from" level.}
\]
\[
\rho_f = \left( \frac{k}{\alpha} \right)^{2/3} [C + \alpha(Z_f - H)]
\]
\[
a_{11} = h_1(\rho_f), \quad a_{12} = h_2(\rho_f)
\]
\[
a_{21} = C h_1(\rho_f) + K h_1^i(\rho_f)
\]
\[
a_{22} = C h_2(\rho_f) + K h_2^i(\rho_f)
\]
\[
V_1 = -E_y i \left( \frac{i R_f^1 + 1}{C} \right) - E_\parallel i \left( \frac{i R_f^1}{C} \right)
\]
\[
V_2 = -2 E_y i
\]

where from (19)
\[
K = i \left( \frac{\alpha}{k} \right)^{1/3}
\]
For Set I B.C.: \( E_{ly} = 1, E_{zy} = 0 \) Applying these boundary conditions and the relations of equation 51 to equation 23 and to equation 24 gives:

\[
\begin{align*}
&\left( -\frac{A}{C} \right)_{I} a_{11} + \left( -\frac{B}{C} \right)_{I} a_{12} = \left( \frac{R_{I}^{f}}{C} \right) \\
&\left( -\frac{A}{C} \right)_{I} a_{21} + \left( -\frac{B}{C} \right)_{I} a_{22} = 0
\end{align*}
\]

Solving these equations simultaneously by the application of equation 50(A,B) gives:

\[
\begin{align*}
\left( \frac{A}{C} \right)_{I} &= \frac{R_{I}^{f}}{C} \cdot \frac{a_{22}}{\Delta_1} \quad \text{(B-52,a)} \\
\left( \frac{B}{C} \right)_{I} &= -\frac{R_{I}^{f}}{C} \cdot \frac{a_{21}}{\Delta_1} \quad \text{(B-52,b)}
\end{align*}
\]

where

\[
\Delta_1 = (a_{11} a_{22} - a_{21} a_{12}) \quad \text{(B-53)}
\]

with \( a_{ij} \) as defined in equation 51.

For Set II B.C.: \( E_{li} = 0, E_{iy} = 1 \) Applying these B.C. conditions and the relations of equation 51 to equation 23 and equation 24 gives:

\[
\begin{align*}
&\left( -\frac{A}{C} \right)_{II} a_{11} + \left( -\frac{B}{C} \right)_{II} a_{12} = -\left( \frac{R_{II}^{f} + 1}{C} \right) \\
&\left( -\frac{A}{C} \right)_{II} a_{21} + \left( -\frac{B}{C} \right)_{II} a_{22} = 2
\end{align*}
\]
Solving these equations simultaneously by the application of equations 50(A,B) gives:

\[
\begin{align*}
\begin{bmatrix} A \\ C \end{bmatrix}_I & = \left( \frac{iR_1^f + 1}{C} \right) \begin{bmatrix} a_{22} - 2a_{12} \\ a_{21} \end{bmatrix} \frac{1}{\Delta_1} \\
\begin{bmatrix} B \\ C \end{bmatrix}_I & = \left( 2a_{11} - \left( \frac{iR_1^f + 1}{C} \right) a_{21} \right) \frac{1}{\Delta_1}
\end{align*}
\]  
(B-54,a) (B-54,b)

Where \( \Delta_1 \) is again defined in equation 53.

For Vertical Polarization at the \( Z_f \), "from", level the following identifications are made.

Identify terms of equation 47 with equation (49,a) and terms of equation 48 with (49,b). That is let:

\[
\begin{align*}
\bar{A} & = \left( \begin{array}{c} 0 \\ \frac{g}{C} \end{array} \right), \quad \bar{B} = \left( \begin{array}{c} 0 \\ \frac{g}{C} \end{array} \right) \\
a_{11} & = h_1(\rho_f), \quad a_{12} = h_2(\rho_f) \\
a_{21} & = C h_1(\rho_f) + \left[ K h'_1(\rho_f) + L h_1(\rho_f) \right]/n_f^2 \\
a_{22} & = C h_2(\rho_f) + \left[ K h'_2(\rho_f) + L h_2(\rho_f) \right]/n_f^2
\end{align*}
\]

Where from equation 40 and 41

\[
K = i\left( \frac{a}{K} \right)^{1/3}, \quad L = i\left( \frac{a}{2K} \right)
\]

and

\[
\begin{align*}
V_1 & = -E_B^i \left( \frac{iR_B^f + 1}{C} \right) - E_Y^i \left( \frac{iR_A^f}{C} \right) \\
V_2 & = -2E_B^i
\end{align*}
\]  
(B-55)
For Set I B.C.: \((E_x^I = 1, E_y^I = 0)\) Applying these B.C. conditions and the relations of equation 55 to equations 47 and 48 gives:

\[
\left( \frac{Q}{c} \right)_{11} + \left( \frac{G}{c} \right)_{12} = \left( \frac{R_x^f + 1}{c} \right)
\]

\[
\left( \frac{Q}{c} \right)_{12} + \left( \frac{G}{c} \right)_{22} = 0
\]

Solving these equations simultaneously by the application of equations 50(A,B) gives:

\[
\begin{align*}
\left( \frac{Q}{c} \right)_{11} &= \frac{1}{\Delta_2} \left[ \left( \frac{R_x^f + 1}{c} \right) a_{22} - 2a_{12} \right] \\
\left( \frac{G}{c} \right)_{11} &= \frac{1}{\Delta_2} \left[ 2a_{11} - \left( \frac{R_x^f + 1}{c} \right) a_{21} \right]
\end{align*}
\]

Where

\[
\Delta_2 = (a_{11} a_{22} - a_{21} a_{12})
\]

with \(a_{ij}\) as defined in equation 55

For Set II B.C.: \((E_x^I = 0, E_y^I = 1)\) Applying these boundary conditions and the relations of equation 55 to equations 47 and 48 gives:

\[
\left( \frac{Q}{c} \right)_{II} a_{11} + \left( \frac{G}{c} \right)_{II} a_{12} = \frac{R_x^f}{c}
\]

\[
\left( \frac{Q}{c} \right)_{II} a_{21} + \left( \frac{G}{c} \right)_{II} a_{22} = 0
\]

Solving this equation simultaneously by the application of equations 50(A,B) gives:

\[
\begin{align*}
\left( \frac{Q}{c} \right)_{II} &= \frac{R_x^f}{c} a_{22} / \Delta_2 \\
\left( \frac{G}{c} \right)_{II} &= - \frac{R_x^f}{c} a_{21} / \Delta_2
\end{align*}
\]

Where \(\Delta_2\) is again defined in equation 55
II. The values of the electric field terms $E_i^j$ and $E_y^i$ at the "to" level (e.g., at height $Z_t$) may be determined from the values of the coefficients $(A, B, F$ and $G)$ as determined in Step I for the Set I and Set II boundary conditions.

For Horizontal Polarization at the $Z_t$ "to" level, the following identifications are made.

Identify terms of equation 24 with equation 49,b. That is:

$$\bar{A} = (-\frac{A}{C}), \bar{B} = (-\frac{B}{C})$$

$$a_{21} = C h_1(\rho_t) + K h_1'(\rho_t); \quad V_2 = -2\left(E_y^i\right)_t$$

$$a_{22} = C h_2(\rho_t) + K h_2'(\rho_t)$$

Where

$$\rho_t = \left(k^2/3\right) [C^2 + \alpha(Z_t - H)]$$

$$K = i\left(\frac{1}{\alpha}\right)^{1/3}$$

For Set I B.C.: (e.g. $(E_i^j)_f = 1; (E_y^i)_f = 0$)

$$(-\frac{A}{C}) a_{21} + (-\frac{B}{C}) a_{22} = -2\left(E_y^i\right)_t$$

or

$$\left(E_y^i\right)_t = \left(\frac{a_{21} + a_{22}}{2}\right)$$

Where the values of $(A/C)_I$ and $(B/C)_I$ were determined by equation 52 and the $a_{ij}$ are given in equation 58.
For Set II B.C.: 
\( \begin{align*} 
(\text{e.g. } (E_\|)^i)_f &= 0, \quad (E_y)^i)_f = i \\
(- \frac{A}{C})_\| a_{21} + (- \frac{B}{C})_\| a_{22} &= -2 (E_y)^t_\| 
\end{align*} \)

or
\[
\begin{bmatrix} 
(E_y)^t_\| 
\end{bmatrix}_\| = \left( \frac{(\frac{A}{C})_\| a_{21} + (\frac{B}{C})_\| a_{22}}{2} \right)
\] (B-60)

Where the values of \((A/C)_\|\) and \((B/C)_\|\) were determined by equation 54 and the \(a_{ij}\) are given by equation 58.

For Vertical Polarization at the \(Z_t\) "to" level, the following identifications are made.

Identify terms of equation 48 with equation (49,b). That is:
\[
\begin{align*} 
\vec{A} &= \left( - \frac{Q}{C} \right), \quad \vec{B} = \left( - \frac{G}{C} \right) \\
a_{21} &= \left\{ C h_1(\rho_t) + \left[ \sum_h h_1(\rho_t) + L h_1(\rho_t) \right] / h^2 \right\} \\
a_{22} &= \left\{ C h_2(\rho_t) + \left[ \sum_h h_2(\rho_t) + L h_2(\rho_t) \right] / h^2 \right\} \\
V_2 &= -2 (E_\|)^i_t \\
L &= i \left(\frac{Q}{2K} \right) \text{ from equation 41} 
\end{align*} \] (B-61)

For Set I B.C.: \( \text{e.g. } (E_\|)^i)_f = 1; \quad (E_y)^i)_f = 0 \)

\[
\begin{align*} 
(- \frac{Q}{C}) I a_{21} + (- \frac{G}{C}) I a_{22} &= -2 (E_\|)^t I 
\end{align*} \)

or
\[
\begin{bmatrix} 
(E_y)^t I 
\end{bmatrix}_I = \left[ \frac{(\frac{Q}{C}) I a_{21} + (\frac{G}{C}) I a_{22}}{2} \right]
\] (B-62)

Where \((A/C)_I\) and \((G/C)_I\) are given by equation 56 and the \(a_{ij}\) are given by equation 61.
For Set II B.C.: 

\( (E^i)_{\text{II}} = 0; \quad (E^i)_{\text{II}} = 1 \)

\[
\left( -\frac{Q}{C}\right)_{\text{II}}a_{21} + \left( -\frac{G}{C}\right)_{\text{II}}a_{22} = -2 \left( (E^i)_{\text{II}} \right)
\]

or

\[
\left( (E^i)_{\text{II}} \right) = \frac{\left( \left( -\frac{Q}{C}\right)_{\text{II}}a_{21} + \left( -\frac{G}{C}\right)_{\text{II}}a_{22} \right)}{2}
\]

(B-63)

Where \((Q/C)_{\text{II}}\) and \((G/C)_{\text{II}}\) are given by equation 57 and the \(a_{ij}\) are given by equation 61.

III. Knowing the values of \(\left( (E^i)_{\text{II}} \right), \left( (E^i)_{\text{II}} \right), \left( (E^i)_{\text{I}} \right), \) and \(\left( (E^i)_{\text{I}} \right)\) at the "to" level, from both sets of coefficients \(A, B, Q\) and \(G\), as determined in Step II, the elements of the ionospheric \(R's\) can be found at the "to" level.

For Horizontal Polarization equation 23 at the "to" level is:

\[
\left( -\frac{A}{C}\right)h_1(\rho_t) + \left( -\frac{B}{C}\right)h_2(\rho_t) + \left[ E^i \right]_t \left( \frac{4R_i + 1}{C} \right)_t = 0
\]

(B-64)

Applying the B.C. of Set I for \((A/C)_{\text{I}}\) and \((B/C)_{\text{I}}\) to equation (64) and applying the B.C. of Set II for \((A/C)_{\text{II}}\) and \((B/C)_{\text{II}}\) to equation (64) gives two equations.
(\( (E_y)_t \))_{\text{I}} (\frac{\|R_1 + 1}{c})_t + (\( (E_y)_t \))_{\text{II}} (\frac{\|R_1}{c})_t
+ (- \frac{A}{C})_I h_1(\rho_t) + (- \frac{B}{C})_I h_2(\rho_t) = 0 \quad (B-65)

and

(\( (E_y)_t \))_{\text{II}} (\frac{\|R_1 + 1}{c})_t + (\( (E_y)_t \))_{\text{I}} (\frac{\|R_1}{c})_t
+ (- \frac{A}{C})_II h_1(\rho_t) + (- \frac{B}{C})_II h_2(\rho_t) = 0 \quad (B-66)

Identifying terms of equations 65 and 66 with equations 49,a and 49,b respectively gives:

\[
\begin{bmatrix}
B = (\frac{\|R_1 + 1}{c})_t \\
A = (\frac{\|R_1}{c})_t
\end{bmatrix}
\]

\[
a_{11} = (\( (E_y)_t \))_{\text{I}} \\
a_{12} = (\( (E_y)_t \))_{\text{II}} \\
a_{21} = (\( (E_y)_t \))_{\text{II}} \\
a_{22} = (\( (E_y)_t \))_{\text{I}}
\]

\[
V_1 = (\frac{A}{C})_I h_1(\rho_t) + (\frac{B}{C})_I h_2(\rho_t) \\
V_2 = (\frac{A}{C})_II h_1(\rho_t) + (\frac{B}{C})_II h_2(\rho_t)
\]

Then from equation (50,a):

\[
\frac{\|R_1}{c}_t = \frac{V_1 ((E_y)_t)_{\text{II}} - V_2 ((E_y)_t)_{\text{I}}}{\((E_y)_t\))_{\text{I}} (\( (E_y)_t \))_{\text{II}} - (\( (E_y)_t \))_{\text{I}} (\( (E_y)_t \))_{\text{II}}}
\]

And from equation (50,b):

\[
\frac{\|R_1 + 1}{c}_t = \frac{V_2 ((E_y)_t)_{\text{I}} - V_1 ((E_y)_t)_{\text{II}}}{\((E_y)_t\))_{\text{I}} (\( (E_y)_t \))_{\text{II}} - (\( (E_y)_t \))_{\text{I}} (\( (E_y)_t \))_{\text{II}}}
\]
For Vertical Polarization equation 47 at the "to" level is:

\[
\begin{align*}
&(-\frac{Q}{C})h_1^0 t + (\frac{G}{C})h_2(\rho t) + (E_\parallel \text{t}) \left(\frac{R_\parallel + 1}{C}\right) \\
&+ (E_y \text{t}) \left(\frac{1}{C}\right) t = 0
\end{align*}
\]  

\(\text{(B-70)}\)

Applying the conditions of Set I for \((Q/C)_I\) and \((G/C)_I\) to equation 70 and applying the conditions of Set II for \((Q/C)_{II}\) and \((G/C)_{II}\) to equation (70) gives the two equations:

\[
\begin{align*}
&\left(\left(E_\parallel \text{t}\right)_I \left(\frac{R_\parallel + 1}{C}\right) \right) + \left(\left(E_y \text{t}\right)_I \left(\frac{1}{C}\right) t\right) \\
&+ \left[(-\frac{Q}{C})_I h_1(\rho t) + (-\frac{G}{C})_I h_2(\rho t)\right] = 0 \tag{B-71}
\end{align*}
\]

and

\[
\begin{align*}
&\left(\left(E_\parallel \text{t}\right)_{II} \left(\frac{R_\parallel + 1}{C}\right) \right) + \left(\left(E_y \text{t}\right)_{II} \left(\frac{1}{C}\right) t\right) \\
&+ \left[(-\frac{Q}{C})_{II} h_1(\rho t) + (-\frac{G}{C})_{II} h_2(\rho t)\right] = 0 \tag{B-72}
\end{align*}
\]

Identifying terms of equation 65 with equation 49, a

\[
\begin{align*}
\bar{A} &= \left(\frac{R_\parallel + 1}{C}\right) t \\
\bar{B} &= \left(\frac{1}{C}\right) t \\
a_{11} &= \left(E_\parallel \text{t}\right)_I \\
a_{12} &= \left(E_y \text{t}\right)_I \\
a_{21} &= \left(E_\parallel \text{t}\right)_{II} \\
a_{22} &= \left(E_y \text{t}\right)_{II} \\
V_1 &= \left(\frac{Q}{C}\right)_I h_1(\rho t) + \left(\frac{G}{C}\right)_I h_2(\rho t) \\
V_2 &= \left(\frac{Q}{C}\right)_{II} h_1(\rho t) + \left(\frac{G}{C}\right)_{II} h_2(\rho t)
\end{align*}
\]  

\(\text{(B-67)}\)
Then from equation (50,a)

\[
\left(\frac{\frac{1}{R} + 1}{c} \right)_t = \frac{V_1 \left(\frac{E_x}{t}\right)_I - V_2 \left(\frac{E_y}{t}\right)_I}{\left(\frac{E_x}{t}\right)_I \left(\frac{E_y}{t}\right)_II \left(\frac{E_x}{t}\right)_II \left(\frac{E_y}{t}\right)_II}
\]

(B-73)

and from equation (50,b)

\[
\left(\frac{\frac{1}{R} \cdot c}{c} \right)_t = \frac{V_2 \left(\frac{E_x}{t}\right)_I - V_1 \left(\frac{E_x}{t}\right)_II}{\left(\frac{E_x}{t}\right)_I \left(\frac{E_y}{t}\right)_II \left(\frac{E_x}{t}\right)_II \left(\frac{E_y}{t}\right)_II}
\]

(B-74)
APPENDIX C
MODIFIED GROUND REFLECTION COEFFICIENTS

Given: The Fresnel Reflection Coefficients at the ground (i.e., z = 0)

\[
R_\| = \frac{C N_g^2 - W}{C N_g^2 + W} \quad (C-1)
\]

\[
R_\perp = \frac{C - W}{C + W} \quad (C-2)
\]

where

\[
W = \sqrt{N_g^2 - S^2} \quad (C-3)
\]

with

\[ S = \sin \theta \]
\[ N_g = \text{index at refraction of ground} \]

Define modified reflection coefficients as:

\[
A = \left( \frac{1}{R_\|} + 1 \right)/C \\
B = \left( \frac{-1}{R_\perp} + 1 \right)/C \quad (C-4)
\]

where \( C = \cos \theta \)

Substituting equations 1 and 2 into equations 4 and 5 gives:

\[
A = \left[ \frac{1}{\left( \frac{C N_g^2 - W}{C N_g^2 + W} \right)} + 1 \right] 1/C \quad (C-6)
\]

\[
B = \left[ \frac{1}{\frac{C - W}{C + W}} + 1 \right] 1/C \quad (C-7)
\]
then with further manipulation:

\[ A = \frac{\left[ \frac{CN_g^2 + W}{CN_g^2 - W} + 1 \right]}{C} = \frac{\left[ \frac{CN_g^2 + W + CN_g^2 - W}{CN_g^2 - W} \right]}{C} \]

\[ = \frac{\left[ \frac{2CN_g^2}{(CN_g^2 - W)} \right]}{C} = \frac{1}{1/2[C - W/N_g^2]} \quad (C-8) \]

and

\[ B = \frac{\left[ \frac{C + W + 1}{C - W} \right]}{C} = \frac{\left[ \frac{C + W + C - W}{C - W} \right]}{C} \]

\[ = \frac{\left[ \frac{2C}{C - W} \right]}{1/C} = \frac{2 \cdot 1/W}{(C - W) \cdot 1/W} = \frac{1}{\frac{1}{2} \left[ \frac{C}{W} - 1 \right]} \quad (C-9) \]

then define:

\[ A = \frac{n_\|}{d_\|} \quad \text{so that:} \quad n_\| = 1 \]

\[ d_\| = \frac{1}{2[C - W/N_g^2]} \quad (C-10) \]

and

\[ B = \frac{n_\perp}{d_\perp} \quad \text{So that:} \quad n_\perp = \frac{1}{W} \]

\[ d_\perp = \frac{1}{2} \left[ \frac{C}{W} - 1 \right] \quad (C-11) \]

The final forms of A and B as presented above are well behaved functions as \( N_g \to \infty \) and \( \theta \to 90^\circ \).
APPENDIX D

MESH SQUARE ANALYSIS

Consider an individual mesh square. See figure D-I. Any point in the complex $\mathbf{\theta}$ plane which lies within (or on an edge of) the square is described by the complex function $F(\theta_r, \theta_i)$. That is:

$$F(\theta) = F_R(\theta_r, \theta_i) + j F_I(\theta_r, \theta_i) \quad (D-1)$$

and

$$\theta = \theta_r + j \theta_i \quad (D-2)$$

where

$F_R(\theta)$ is the real part of the complex function $F(\theta)$

$F_I(\theta)$ is the imaginary part of the complex function $F(\theta)$

$\theta_r$ is the real part of the complex number $\theta$

$\theta_i$ is the imaginary part of the complex number $\theta$.

The lower left hand corner of the square is taken as the reference position and each side of the square is one-mesh unit in length. The geometry of the mesh square is shown below where the $F(\theta_r, \theta_i)$ values of the corners of the square are identified.

Figure D-I. Mesh Square Geometry
A basic assumption put on \( F(\theta_r, \theta_i) \) is that \( F_R(\theta_r, \theta_i) \) and \( F_I(\theta_r, \theta_i) \) are to be linear along each edge of the mesh square.

The function of \( F(\theta_r, \theta_i) \) at any point within the square is given by:

\[
F(\theta_r, \theta_i) = F(0,0)(1 - \theta_r - \theta_i + \theta_r \theta_i)
\]
\[
+ F(0,1)(\theta_i - \theta_r \theta_i)
\]
\[
+ F(1,0)(\theta_r - \theta_r \theta_i)
\]
\[
+ F(1,1)(\theta_r \theta_i)
\]  
(D-3)

or expanding:

\[
F(\theta_r, \theta_i) = F(0,0) + \theta_i(F(0,1) - F(0,0)) + \theta_r(F(1,0) - F(0,0))
\]
\[
+ \theta_r \theta_i (F(0,0) + F(1,1) - F(0,1) - F(1,0))
\]  
(D-4)

which can be written as

\[
F(\theta_r, \theta_i) = a + b \theta_i + c \theta_r + d \theta_r \theta_i
\]  
(D-5)

(a) Hyperbolic and Asymptotic Lines:

Consider the imaginary part of \( F(\theta_r, \theta_i) \) and \( F_I(\theta_r, \theta_i) = 0 \). This gives (from equation D-5):

\[
a_i + b_i \theta_i + c_i \theta_r + d_i \theta_r \theta_i = 0
\]  
(D-6)

Also, in the same way the real part of \( F(\theta_r, \theta_i) \) gives:

\[
a_r + b_r \theta_i + c_r \theta_i + d_r \theta_r \theta_i = 0
\]  
(D-7)

Equation D-6 (or equation D-7) is the equation of an "equilateral" or "rectangular" hyperbola with vertical and horizontal asymptotes.

To determine the center of the asymptotes for the hyperbolic curve of equation D-6, that equation must be solved for \( \theta_r \) and also for \( \theta_i \).
That is, (from equation D-6):

\[(a_i + b_i \theta_i) + (c_i + d_i \theta_i) \theta_r = 0\]

or

\[\theta_r = -\left(\frac{a_i + b_i \theta_i}{c_i + d_i \theta_i}\right)\]  \hspace{1cm} (D-8)

Also from equation D-6:

\[(a_i + c_i \theta_r) + (b_i + d_i \theta_r) \theta_i = 0\]

or

\[\theta_i = -\left(\frac{a_i + c_i \theta_r}{b_i + d_i \theta_r}\right)\]  \hspace{1cm} (D-9)

Now let \(\theta_r \to \infty\) in equation (D-8) so that:

\[c_i + d_i \theta_i = 0\]

or

\[\theta_i = -\frac{c_i}{d_i}\]  \hspace{1cm} (D-10)

The line described by equation (D-10) is the horizontal asymptote to the equilateral hyperbola.

From equation D-9 let \(\theta_i \to \infty\) so that:

\[b_i + d_i \theta_r = 0\]

or

\[\theta_r = -\frac{b_i}{d_i}\]  \hspace{1cm} (D-11)
The line described by equation (D-11) is the vertical asymptote to the equilateral hyperbola. These asymptotic lines in relation to the mesh square are illustrated in the following figure:

![Figure D-II. Vertical and Horizontal Asymptotes for an equilateral hyperbola.](image)

(b) **Intersection of Hyperbolic Lines with the Edges of the Mesh Square:**

Identify the edges of the mesh square, as shown in figure D-I, as edges 1, 2, 3 and 4. Also, consider only the imaginary part of $F(\theta_r, \theta_i)$ of equation D-4 and equate $F_I(\theta_r, \theta_i)$ to 0. Equation D-4 becomes:

$$
F_I(0,0) + \theta_i(F_I(0,1) - F_I(0,0)) + \theta_r(F_I(1,0) - F_I(0,0)) + \theta_r\theta_i(F_I(0,0) + F_I(1,1) - F_I(0,1)) - F_I(1,0) = 0 \quad (D-12)
$$

To find the intersection of a branch of the equilateral hyperbola, described by equation D-12, with edge No. 1 of the square, set $\theta_r = 0$ and
note that \( \theta_i \) varies between 0 and 1 along edge No. 1. Equation D-12 becomes:

\[
F_I(0,0) + \theta_i(F_I(0,1) - F_I(0,0)) = 0
\]

or

\[
\theta_i = \frac{-F_I(0,0)}{F_I(0,1) - F_I(0,0)}
\]

Therefore the point of intersection for edge No. 1 is:

\( \theta_r = 0, \theta_i \) is as given by equation D-13.

The point of intersection between the hyperbola and the remaining three sides may be computed in a similar manner. The results are as follows:

The point of intersection for edge No. 2 is:

\[
\theta_r = \frac{F_I(0,0)}{F_I(1,0) - F_I(0,0)}, \ \theta_i = 0
\]  \hspace{0.5cm} (D-14)

The point of intersection edge No. 3 is:

\[
\theta_r = 1, \ \theta_i = \frac{F_I(1,0)}{F_I(1,1) - F_I(1,0)}
\]  \hspace{0.5cm} (D-15)

The point of intersection edge No. 4 is:

\[
\theta_r = \frac{F(0,1)}{F_I(1,1) - F_I(0,1)}, \ \theta_i = 1
\]  \hspace{0.5cm} (D-16)
APPENDIX E

SOLUTION OF THE QUADRATIC EQUATION

FOR THE GENERAL ROOT-FINDER.

Given the Quadratic equation as used in the Root-Finder procedure:

\[ aX^2 + 2bX + C = 0 \]  \hspace{1cm} (E-1)

\[ X = -\frac{2b \pm \sqrt{4b^2 - 4ac}}{2a} = \frac{b}{a} \left[ -1 \pm \sqrt{1 - \frac{ac}{b^2}} \right] \]

let \( \epsilon = -\frac{ac}{b^2} \)

and then \((1+\epsilon)^{1/2} = 1 + \frac{1}{2} \epsilon + \frac{1}{2} \left( \frac{1}{2} - 1 \right) \epsilon^2 + \frac{1}{2} \left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right) \epsilon^3 + \ldots \)

Now \( X = \frac{b}{a} \left[ -1 \pm \sqrt{1 + \epsilon} \right] \)

so that

\[ X_1 = \frac{b}{a} \left\{ \frac{1}{2} \left( \frac{ac}{b^2} \right) + \frac{1}{2} \left( \frac{1}{2} - 1 \right) \left( \frac{ac}{b^2} \right)^2 + \frac{1}{2} \left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right) \left( \frac{ac}{b^2} \right)^3 + \ldots \right\} \]

\[ X_1 = -\frac{1}{2} \frac{c}{b} \left\{ 1 + \frac{1}{2} \left( \frac{1}{2} - 1 \right) \left[ -\frac{ac}{b^2} \right] + \frac{1}{2} \left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right) \left[ \frac{ac}{b^2} \right]^2 + \ldots \right\} \] \hspace{1cm} (E-2)

let \( t_0 = 1 \)

\[ t_1 = \left( \frac{1}{2} - 1 \right) \left[ -\frac{ac}{b^2} \right] \]

\[ t_2 = \frac{1}{2} \left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right) \left[ \frac{ac}{b^2} \right]^2 \]
or \[ t_i = t_{i-1} \left( \frac{1}{2} - \frac{1}{i+1} \right) \frac{ac}{b^2} \] 

or \[ t_i = t_{i-1} \left( \frac{1}{2} - \frac{1}{i+1} \right) \frac{ac}{b^2} \] 

and \[ x_1 = -\frac{1}{2} \frac{c}{b} \left\{ t_0 + t_1 + t_2 + \cdots + t_N \right\} \] \hspace{1cm} (E-3) 

Now \[ x_1 = \frac{b}{a} \left[ -1 + \sqrt{1 + \varepsilon} \right] \] \hspace{1cm} (E-4) 

also \[ x_2 = \frac{b}{a} \left[ -1 - \sqrt{1 + \varepsilon} \right] \] 

Then: \[ -x_1 = \frac{b}{a} \left[ 1 - \sqrt{1 + \varepsilon} \right] \] 

and \[ -\frac{2b}{a} - x_1 = \frac{b}{a} - \frac{b}{a} \sqrt{1 + \varepsilon} = \frac{-2b}{a} \] 

or \[ -\frac{2b}{a} - x_1 = -\frac{b}{a} - \frac{b}{a} \sqrt{1 + \varepsilon} \] 

or \[ -\frac{2b}{a} - x_1 = \frac{b}{a} \left[ -1 - \sqrt{1 + \varepsilon} \right] \] 

such that: \[ -\frac{2b}{a} - x_1 = x_2 \] \hspace{1cm} (E-5) 

Equations E-3 and E-5 give the solutions of the Quadratic equation (i.e., Equation E-1) as solved in the Root-Finder routine.
APPENDIX F

FORTRAN LISTING OF

THE MODESrch COMPUTER

PROGRAM
PROGRAM "MODESRCH"

THIS PROGRAM FINDS THE WAVEGUIDE MODES FOR THE EARTH-IONOSPHERE
WAVEGUIDE FOR ELF-VLF-LF

MAIN
FOR NS

PROGRAM FOR COMPUTING EARTH-IONOSPHERE WAVEGUIDE EIGEN-ANGLES AND
EXCITATION FACTORS WHICH ARE OUTPUT ON PUNCHED CARDS
FOR USE IN A PROGRAM WHICH COMPUTES FIELD STRENGTHS.

THIS ROUTINE CONTROLS THE SEQUENCE OF INPUT AND EXECUTION AND
PROVIDES FOR NAMELIST INPUT. A DESCRIPTION OF THE
INPUT PARAMETERS IS GIVEN IN A DNA REPORT.

COMMON/IC COM/IDENT(20)
COMMON/PTALT/ TALT, RALT
COMMON/PR1T/ BPRINT
COMMON/FP/HUFLk
COMMON/RTOL C/RTOL
COMMON/IEXACT C/IEXACT
COMMON/FLD COM/AZIM,CODIP,MAG FLD
COMMON/FREQ COM/FREQ
COMMON/GRD COM/EPERS,SIGMA
COMMON/HPP COM/BETA,HPRIME,SCLHTS,ENMIN
COMMON/WR COM/OMEGA,TMPOB
COMMON/TPEN C/TPEN
COMMON/EXPNU C/COEFNU(3),EXPNU(3)
COMMON/INH COM/IONS,HRATI(3)
COMMON/LIMS C/RANGER(2),RANGE(2)
COMMON/GMAX C/GMAX
COMMON/PARAM C/TMESH
COMMON/SEP COM/SEP
COMMON/TCL COM/LUB
COMMON/NEIGN C/NEIGEN
COMMON/NPUNCH C/NPUNCH
COMMON/CDF PXT/CAROPT
COMMON/CRD PNC/CARDPN
COMMON/CLT OF ICUT,CUTOFF
COMMON/RPO COM/RHO,REFLHT
COMMON/H COM/H
COMMON/MFRNT/MPIRT
COMMON/XFRNT/XPRINT
COMMON/FXTRA9/1BOFFR(50)

NAMELIST/DATUM/AZIM,CODIP,MAG FLD,
S FREQ,TALT,RALT,
S LPSP,SIGMA,
S BETA,HPRIME,SCLHTS,ENMIN,
S TPEN,TMPOB,
S COEFNU,EXPNU,NUFLAG,
S HRATI,TMESH,LUB,
S RANGER,RANGE(2)
GMAX, SEPI, RTOL, IEXACT,
NEIGEN,
NPUNCH, RHO, REFLHT,
H,
ICUT, CUTOFF,
BPRINT, NPRINT,
XPRINT, XPINT, CARDPT, CARDPN

INTEGER PROF, GULF, ADAT, QUIT
INTEGER CARDPT, CARDPN
INTEGER XPRINT
REAL MAGFLD, NRATIO, LUB
DIMENSION IBCD(20)
DATA PROF/'PROF'/'GULF'/'GULF'/'ADAT'/'EAD'/,'IEND'/'GEN'/,'QUIT'/'QUIT'/
DATA IDBLANK/'''/
DATA ITOP=10/3''/

HOFWR=0.6
UMGAR=2.5E05

INITIAL VALUES OF NAMELIST PARAMETERS.

IEXACT=0
ICUT=1
CUTOFF=0.0401
TGPEN=1.0E3
SCLHTS=3.0
ERNIN=1.0E-1
TEMPOR=2.0
HPPRF=69.0
FREQ=0.0
RHO=0.0
REFLHT=50.0
H=50.0
TALT=0.0
RALT=0.0

GMAX=5.0
PTCL=0.01
SEP=0.1
TMESH=1.0
LUB=-1.0

NUFLAG=0
COEF1U(1)=1.816E11
COEF1U(2)=4.540E9
COEF1U(3)=4.540E9
EXP1U(1)=-0.15

143
EXPNU(2) = -0.12
EXPNU(3) = -0.15

C
MRATIO(1) = 1.0
MRATIO(2) = 58000.0
MRATIO(3) = 58000.0

C
ILNS = 0
NEGFN = 0
KPUNCH = 1

C
BPRINT = 0
MPRINT = 0
XPRINT = 0
CAKOPT = 0
CAKOPT = 1

C
DO 7 I = 1, 20
7 ICOUNT(I) = IBLANK

C
DEFAULT VALUES OF GROUND PARAMETERS ARE TAKEN AS SEA WATER
SIGMA = 4.64
EPSR = 81.

C
CONTROL OF SEQUENCE OF INPUT AND EXECUTION.

10 READ(5, * , Cl , END = 9999) IBCD
   PRINT 902, IBCD
   IF (IBCD(1) .EQ. PROF) GO TO 11
   IF (IBCD(1) .EQ. PROF) GO TO 12
   IF (IBCD(1) .EQ. PROF) GO TO 13
   IF (IBCD(1) .EQ. PROF) GO TO 20
   IF (IBCD(1) .EQ. PROF) GO TO 9999
   PRINT 904
900 FORMAT ('**ERROR IN CONTROL CARD**')
   STOP

C
9999 FORMAT( 'STOP')
9999 FORMAT( '***THE CALCULATIONS ARE COMPLETE***')
   STOP

C
11 IONS = 0.
   IF (IBCD(3) .EQ. ITHREE) IONS = 1
   CALL PRFL IN
   HPRIME = -99.0
   GO TO 10
12 CALL COLF IN
   HPRIME = -99.0
   GO TO 10
13 READ 901, IDENT
   PRINT 902, IDENT
   GO TO 10
NAMELIST INPUT

20 IFHFFX(1) = IBCD(1)
    IFHFFX(2) = IBCD(2)
    J = 3
    IF 21 L = 1,25
        IF (P* 901, IBCD)
        PRINT 902, IBCD
        IF (I-CD(1) .EQ. IEND) GO TO 22
    21  CONTINUE
    IFHFFX(J) = IBCD(1)
    J = J+1

21 CONTINUE

22 IFHFFX(J) = IBCD(1)
    J = J+1
    IFHFFX(J) = IBCD(2)
    CALL CONE (I-CD, 4, J)
    READ(1, CATUMP)
    HCFWH = C.C

IF (TMFLEH .LT. J) TMFLEH = SQRT(3.75/FREQ)
IF (LUB .LT. 0) LUB = SQRT(15.0/FREW)*0.01
DO 205 U = 2,5, TMFLEH, LUB

205 FORMAT (*C, 'TMFLEH = ', F6.3, 4X, 'LUB = ', F6.4)
IF (FREQ .LE. 1.0) IFLEH = 1

IF (NFLAG .EQ. J) GO TO 40
CCEFNU(1) = 4.303E11
CCEFNU(2) = 1.376E10
CCEFNU(3) = 1.076E10
EXPNU(1) = -0.122
EXPNU(2) = -0.122
EXPNU(3) = -0.122

&ISCELANEOUS INPUT OPERATIONS AND CALL TO CONTROL ROUTINE.

40 IF (HPRFIE .GT. 0) CALL SET EN
    IF (HPRFIE .LT. 0) AND FREQ .GT. 1.0) CALL CUT TUP
    IF (RANGEX(1) .LT. RANGEK(2)) GO TO 41
    TEMP = RANGEK(1)
    RANGEX(1) = RANGEK(2)
    RANGEK(2) = TEMP

41 IF (RANGEK(1) .GT. RANGEX(2)) GO TO 42
    TEMP = RANGEK(1)
    RANGEK(1) = RANGEX(2)
    RANGEX(2) = TEMP

42 CONTINUE

INCR=ITIME(ICK)
    CALL WVGCLI
    INCR=ITIME(ICK)-INCK
    TIME=FLAT(INCK)/100.
    PRINT 615, TIME

145
615 FORMAT('0',/ 'TIME REQUIRED FOR THIS PROFILE = ',F10.2, ' SECONDS')
PRINT 980
980 FORMAT('I')
IF (HPRIME .LT. 0 ) CALL FLPEAT
GO TO 10
C
901 FORMAT(2CA4)
932 FORMAT(' ',2OA4)
C
END
SUBROUTINE WVGUID
C FOR MS
C CONTROL WAVE.
C
COMMON/ICL COM/TOL
COMMON/IEXACT C/IEXACT
COMMON/MPFINT MPFINT
COMMON/MPKCH MPKCH
COMMON/NEIGN C/NEIGN
COMMON/LIMS C/RANGEP(2), RANGEI(2)
COMMON/PARAM C/THESH
COMMON/GMAX C/GMAX
COMMON/RF OF AS/NR A
COMMON/CJMK HTS(101)
COMMON/SIDES C/TLEFT, TRIGHT, TBOT, TTOP
COMMON/THETA C/THETA
COMMON/FIG COM/EIGFN(100), NR E
COMMON/KXACT C/ KEXACT
COMMON/L CCM/0
COMPLEX THETA, EIGEN,
+ ELIST(100),
+ TEMP
C
C IF (NPUNCH .EQ. 0 .AND. NEIGN .EQ. 0) GO TO 90
C
C INITIALIZE FULL-WAVE SOLUTION.
CALL SET RK
CALL INIT 1
C
C SET UP SMALL RECTANGLES.
NK EL = 0
NK BOX = (RANGE(2) - RANGE(1)) - TOL / GMAX + 1.0
IF (IEXACT .NE. 0) NK BOX = 1
RLSPAN = (RANGE(2) - RANGE(1)) / NK BOX
IHOX = 1
TTOP = RANGE(1)
TBOT = RANGE(2)
TLEFT = TLEFT
C
20 TRIGHT = TLEFT
TLEF = TRIGHT - RLSPAN
C
C SET REFERENCE HEIGHT AND INITIALIZE LAGRANGE INTERPOLATION.
D = 0.0
IF (IEXACT .NE. 0) CALL SET RK
C
C INITIALIZE RBAR SOLUTION AND FREE-SPACE INTEGRATION.
CALL INIT RB
CALL INIT FSHTS(NR A)
C
C FIND ZEROS OF F FUNCTION - LAGRANGE INTERPOLATION
USED IF NOT ELF.
KEXACT = 0

147
CALL FZEKOS (TLEFT, TRIGHT, TBOT, TTOP, TMESH, TOL, NPRINT, $ EIGEN, NR E)
        IF (NR F .EQ. 0) GO TO 77

C ORDER EIGEN ANGLES ACCORDING TO VALUE OF REAL PART.
    IF (NR L .EQ. 1) GO TO 64
    NREML = NR E-1
    J1 = 1, NREML
    JF1 = J1+1
    DO 62 J = JF1, NR F
    IF (REAL(EIGEN(J)) .LT. REAL(EIGEN(JE))) GO TO 62
    TEMP = EIGEN(J)
    EIGEN(J) = EIGEN(JE)
    EIGEN(JF) = TEMP
62 CONTINUE
63 CONTINUE
64 CONTINUE

C FOLLOW EIGEN ANGLES AS FUNCTION IS CHANGED FROM THAT USING
C LAGRANGE INTERPOLATION TO THAT USING EXACT
C FULL-WAVE SOLUTION VALUES OF REFLECTION COEFFICIENTS.
    IF (EXACT .EQ. 0) CALL FINAL
    IF (NR E .EQ. 0) GO TO 77

C DISCARD EIGEN-ANGLES WHICH ARE OUTSIDE CURRENT SMALL RECTANGLE.
    J = 0
    DO 71 JE = 1, NR F
    IF (REAL(EIGEN(JE)) .LT. TLEFT-0.0*TOL .OR.
     4 REAL(EIGEN(JE)) .GT. TRIGHT+2.0*TOL) GO TO 71
    J = J+1
    EIGEN(J) = EIGEN(JE)
71 CONTINUE
    NR F = J

C STORAGE OF EIGEN ANGLES.
    IF (LR .EQ. 0) GO TO 77
    DO 76 JF = 1, NR E
    NR EL = NR EL+1
70 ELIST(NR EL) = EIGEN(JE)
77 IBOX = IBOX+1
    IF (IBOX .LE. NR BOXS) GO TO 20
    NR E = NR EL
    IF (NR E .EQ. 0) GO TO 85
    DO 78 JE = 1, NR E
70 EIGEN(JE) = ELIST(JE)

C COMPUTE AND PUNCH NPUNCH OUTPUT AND PRINT TABLE.
    IF (NPUNCH .NE. 0) CALL NP OUT
C
C PRINT AND PUNCH EIGEN ANGLES IF NEIGEN SET TO 1.
    IF (NEIGEN .NE. 0) CALL NE OUT
RETURN
985 FORMAT ('0', 'N') MODES)
RETURN
C
990 FORMAT ('0', 'SET NPUNCH UK NEIGEN')
STOP
C
END
SUBROUTINE SET RH
C
C ROUTINE FOR SELECTING GIVEN POINTS IN THE LAGRANGE INTERPOLATION
C AND FOR SELECTING THE REFERENCE HEIGHT.
C
COMMON/MPRINT/MPRINT
COMMON/PAFAM C/TMESH
COMMON/HTS COM/HTS(101)
COMMON/NR OF AS/NK A
COMMON/SIDES C/TLEFT,TRIGHT,TBOT,TTOP
COMMON/THETA C/THETA
COMMON/IDERV C/IDERV
COMMON/INTEGR/R(4),DROC(4)
COMMON/DM COM/DM
DIMENSION PART(2)
COMPLEX THETA,R,DROC,

$ PARTS,
$ THETAG(20),
$ PHGT(4,20),R GIVEN(4,20),R TEMP(4,20),R SAVE(4,20)
$ EQUIVALENCE (PART(1),PARTS)
DATA HT INC/1.0/
DATA HMRGN/20.0/

C
C SELECT VALUES OF THETA AT WHICH FULL-WAVE SOLUTIONS ARE TO BE
C CARRIED OUT FOR USE IN LAGRANGE INTERPOLATION.
C
JLT = TLEFT/TMESH=1.0
JKT = TRIGHT/TMESH+2.0
JBOT = TBOT/TMESH-2.0
JTOP = TTOP/TMESH
JTUP = JTOP+1

C
PART(1) = (JRT+1)*TMESH
PART(2) = (JTOP+1)*TMESH
THETAG(1) = PARTS
PART(1) = (JLT-1)*TMESH
PART(2) = (JTOP+1)*TMESH
THETAG(2) = PARTS
PART(1) = (JRT+1)*TMESH
PART(2) = (JBOT-1)*TMESH
THETAG(3) = PARTS
PART(1) = (JLT-1)*TMESH
PART(2) = (JBOT-1)*TMESH
THETAG(4) = PARTS
NG = 4
CALL INIT LG (THETAG,NG)
NG0 = NG*8

C
C SELECT THE HEIGHT (TO NEAREST WHOLE KILOMETER) AT WHICH
C THE MINIMUM OCCURS IN
C THE SQUARE OF THE MAGNITUDE OF THE DERIVATIVE OF
C (R+1.0)/C ART C=CUH(SIN(THETA)), SUMMED OVER ALL FOUR
C ELEMENTS OF R=REFLECTION MATRIX AND SUMMED OVER ALL

150
POINTS AT WHICH FULL-WAVE SOLUTIONS ARE CARRIED OUT
FOR USE IN LAGRANGE INTERPOLATION. DERIVATIVES OF THE
FULL-WAVE SOLUTIONS ARE NOT COMPUTED. RATHER,
DERIVATIVES OF THE LAGRANGE INTERPOLATION FORMULA
ARE USED. NOTE THAT THE FULL-WAVE SOLUTIONS ARE
CARRIED OUT TO THE BOTTOM OF THE PROFILE AFTER WHICH
THE FULL-WAVE SOLUTION IS CARRIED OUT IN AN UPWARD
DIRECTION THROUGH FREE SPACE OVER A CURVED EARTH TO
HEIGHTS AT ONE KILOMETER INTERVALS. THE HEIGHT AT
WHICH THE MINIMUM OCCURS IS PRINTED OUT AS THE
"REFLECTION HT" AND IS THEREAFTER USED AS THE
REFERENCE HEIGHT AT WHICH THE F FUNCTION IS COMPUTED.

IF (MPrint,ne,0) PRINT 911
911 FORMAT ("1","INTEGRATION TO DETERMINE REFLECTION HT--")
I DERIV = 0
K HTTEMP = 9.9E9
K SAVE = 9.9E9
LIMIT = NG
GB = 0.
DO 12 JG=1,NG
THETA = THETAG(JG)
CALL INTEG
12 CALL XFFK(R,R GIVEN(1,JG),8)

CALL XFFK(R GIVEN,R80T,NGB)
CALL XFFP(R GIVEN,R TEMP,NGB)
CALL XFFR(R GIVEN,R SAVE,NGB)
CALL SFT LAG (NG,R GIVEN)
DO 13 JG=1,LIMIT
THETA = THETAG(JG)
CALL LAG DER
DO 13 I=1,4
13 GB = GB+CABS(DRDC(I))**2
GR MIN = GB
IF (MPrint,ne,0) PRINT 912,HTS(NF A1),GB
HTFRST = HTS(NR A)
14 HTLAST = HTFRST+HT INC
CALL INIT FS (HTS(NR A),HTLAST)
GR = 0.0
DO 15 JG=1,NG
THETA = THETAG(JG)
CALL XFER (R80T(1,JG),R,8)
CALL FS INTG
15 CALL XFER (R,R GIVEN(1,JG),8)
CALL SET LAG (NG,R GIVEN)
DO 16 JG=1,LIMIT
THETA = THETAG(JG)
CALL LAG DER
DO 16 I=1,4
16 GB = GB+CABS(DRDC(I))**2
IF (MPrint,ne,0) PRINT 912,HTLAST,GB
912 FORMAT ("1","AT HT =",F7.2,1X,"GB =",1PE9.2)
IF (GB .GT. GB MIN) 30 TO 17
RHTEMP = HILAST
CB MIN = GB
CALL XFER (R GIVEN, R TEMP, NG8)
GO TO 16
17 KSAVE = RHTEMP
CALL XFER (R TEMP, R SAVE, NG8)
10 HTFRST = HTLAST
IF (HTFPST .LT. (HTS(1) + MARGIN)) GO TO 14
IF (RHSAVE .GT. 9.0F9) GO TO 90
D = KSAVE
CALL XFER (R SAVE, R GIVEN, NG8)
CALL SET LAG (NG, R GIVEN)
C
PRINT 913, C, TLEFT, TRIGHT
913 FORMAT ('0',*REFLECTION HT SET TO', F7.2, 1X, 'KM', $5X,'THETA RANGE = ', F8.3,' TO ', F8.3)
C
IF (HPRINT .EQ. 0) GO TO 39
PRINT 931
931 FORMAT ('0',*REFLECTION COEFFICIENTS AT THE GIVEN POINTS--')
DO 32 JG=1,NG
32 PRINT 932, THETAG(JG), (K GIVEN(I, JG), I=1, 4)
932 FORMAT ('0', 'AT THETA = ', 2F9.3, 5X, 'R = ', 4(2X, 2F9.5))
39 CONTINUE
RETURN
C
90 PRINT 900, TLEFT, TRIGHT
900 FORMAT ('C',*REFLECTION HT CANNOT BE SET FOR THETA RANGE = ', $F8.3,' TO ', F8.3)
STOP
C
END
SUBROUTINE PRFL IN
C FOR MS
C THIS ROUTINE PROVIDES FOR INPUT OF ELECTRON DENSITY AND COLLISION
C FREQUENCY PROFILE CARDS AND FOR LIMITING THE EXTENT
C OF THE TOP AND BOTTOM OF THE ELECTRON DENSITY PROFILE.
C INPUT OF ION PROFILES IS ALSO PROVIDED FOR. FURTHER
C DESCRIPTION OF THE INPUT IS GIVEN IN A DNA REPORT.

COMMON/HTLCON/NU,HTS NUI(50),ALCNU(50,3)
COMMON/FRQ/ CPRINT
COMMON/CLT OF/ ICUT,CUTOFF
COMMON/F0D CUM/ALIM,CODIP,MAG FLD
COMMON/FRQ CUM/ FREQ
COMMON/AR OF AS/NU A
COMMON/HTS CUM/HTS(101)
COMMON/ALOGN C/ALOGEN(101,3)
COMMON/HP CUM/BETA,HPIKE,SCLHTS,ENMIN
COMMON/FXNU C/COEFNU(3),EXPNU(3)
COMMON/CUM/IONS,MRATIO(3)
COMMON/E0 COLL/HT,EN(3),CNU(3)
COMMON/HR CUM/OMEGAR,TMP99
COMMON/HF/HUFR
COMMON/TCPEN C/TOPEN
COMMON/JAY CUM/JAY
REAL MAGFLD,MRATIO
DIMENSION IBCD(20),ENS(3),
*ITITLE(20),SAVHT(101),SAVEN(101,3)

C INTEGER CPRINT
C******************************************************************************
C INPUT OF AN ELECTRON DENSITY PROFILE, IONS OPTIONAL.
C******************************************************************************

READ 901,IBCD
PRINT 902,IBCD
DO 113 K =1,20
113 ITITLE(K)=IBCD(K)
J = 1
11 READ 904,HT,ENS(1),ENS(2)
904 FORMAT (F7.2,5X,E9.2,E10.2)
IF (HT .LT. 0.0) GO TO 15
HTS(J) = HT
IF (J .NE. 1 , ANC . HTS(J) .GE. HTS(J-1)) STOP
ENS(3) = ENS(2)-ENS(1)
HTS(J) = ENS(3)
SAVHT(J)=HT
SAVEN(J,1)=ENS(1)
SAVEN(J,2)=ENS(2)
SAVEN(J,3)=ENS(3)
PRINT 916,HT,ENS(1),ENS(2)
916 FORMAT (',1X,F7.2,5X,E9.2,E10.2')
DO 13 K=1,3
IF (ENS(K) .LT. 1.0E-10) ENS(K) = 1.0E-10
13 ALUGEN(J,K) = ALOG(ENS(K))
   A = J
   J = J+1
   GOTO 11
13 PRINT 900,HT
   RETURN
C*********************************************************************C
C FNTRY REPEAT
C
C THIS SECTION IS USED WHEN THE ORIGINAL UNMODIFIED PROFILE IS TO BE USED IN THE CALCULATIONS
C
NR A=NTCT
DO 134 K = 1,3
   DO 134 KJ=1,NTCT
   HTS(KJ) = SAVHT(KJ)
   IF(SAVEN(KJ,K) .LT. 1.0E-10) SAVEN(KJ,K) = 1.0E-10
134 ALUGEN(KJ,K)=ALOG(SAVEN(KJ,K))
   RETURN
C*********************************************************************C
C ENTRY GCLF IN
C
C INPUT OF COLLISION FREQUENCY PROFILE CARDS
C
J = 1
21 READ 901,(IBCD(L),L=1,20)
   PRINT 902,(IBCD(L),L=1,20)
   CALL COPF(IBCD,80)
   READ (65,905) HT,CNU
905 FORMAT (F7.2,4X,3(1X,E9.2) )
   IF (HT .LT. 0.0) GO TO 29
   HTS NU(J) = HT
   DO 22 K=1,3
      IF(CNU(K) .LT. 1.0E-10) CNU(K) = 1.0E-10
22 ALCNU(J,K) = ALOG(CNU(K))
   J = J+1
   GO TO 21
23 NT NU=J-1
   RETURN
C*********************************************************************C
C ENTRY SET EN
C
C USE OF VARIOUS NAMELIST PARAMETERS TO SET UP A DESCRIPTION OF AN-
C EXPONENTIAL ELECTRON DENSITY PROFILE IN TERMS OF
C VALUES AT THE TOP AND BOTTOM OF THE PROFILE.
C

HIS(1) = AINT(HPRIME+SCLHTS/BETA+0.5)
FNA = 7.65535E-5*COEFNU(1)*EXP((EXPNU(1)+BETA)*HTS(1)-BETA*HPRIME)
ALOGN(1,1) = ALOG(FNA)
HTS(2) = (ALOG(ENMIN/(7.65535E-5*COEFNU(1)))*BETA*HPRIME)/
      (EXPNU(1)+BETA)
IF (HTS(2) .GT. 0.0) GO TO 501
HTS(2) = 0.0
      IF = 7.65535E-5*COEFNU(1)*EXP(-BETA*HPRIME)
      ALOGN(2,1) = ALOG(ENO)
      GO TO 502

501 ALOGN(2,1) = ALOG(ENMIN)
502 NKA = 2
951 FORMAT ('** ',F7.2,5X,1PE9.2)

C

-----COMPUTED PROFILE PARAMETERS-----
C

PW = 0.28318307N03*FREQ
PJD = -1.75879601L/PW*MAGFLD
PRINT 213

213 FORMAT('** COMPUTED PROFILE PARAMETERS:

T9, H4, T15, N4, T27, N4, T39, N4, T51, Y4, T63, Z4, T75, W4)
DO 25 L=1,2
PONU=COEFNU(1)*EXP(EXPNU(1)*HTS(L))
PUNE=7.65535E-5*PONU* EXP(BETA*(HTS(L)-HPRIME))
POW = 3.1823709*PUNE/PONU
POX = 3.1823709*PUNE/(PW*PW)
POZ = PONU/PW
PRINT 214, HTS(L), PUNE, PONU, PCX, POY, POZ, PJWR
214 FORMAT(F7.1,1P6E12.3)

C

DETERMINATION OF HEIGHT WHERE OMEGA-R EQUALS 2.5E05
HOFWR = HPRIME
PRINT 952, HOFWR
952 FORMAT('** OMEGA-R = 2.5E05 AT TOP HEIGHT = ',F7.2)
C

RETURN
C

******************************************************************************
C

ENTRY CUT TOP
C

USE OF TEMPC-B AND TOPEN TO LIMIT THE EXTENT OF THE TOP OF AN
ELECTRON DENSITY PROFILE WHICH HAS BEEN INPUT FROM
PROFILE CARDS. APPLICABLE ALSO TO THE ION PROFILES
IF IONS ARE AN INPUT.

IN THIS SECTION OF THE PROGRAM THE TOP OF THE PROFILE IS DETERMINE
THE NUTATION USED IS THAT WAITS ('OMEGA-R') IS DENOTED BY 'WR'.
THE VALUE OF 'B' AS DESCRIBED IN G.E. TEMPO REPORT 75-TMP-9, V. 7,
PAGES 11 AND 12 IS DENOTED AS 'BCUT'
AND IS INPUT IN TO THIS PROGRAM AS THE NAMELISI VARIABLE 'TEMPOB'
TO DETERMINE THE VALUE OF BCUT TO BE USED
AND TO CUT OFF THE TOP OF THE PROFILE

\[ \Omega = 6.26 \times 10^{-3} \times FRQ \]
PLASCN = 3.102357EY
\[ NU = 560 \times JRFV = 1, \text{NR A} \]
J = NR A - JRFV + 1
HT = HTS(J)
JAY = J

\[ \text{IF}(J \times \text{NC} \times \text{NR A}) \text{ JAY = NR A - 1} \]
CALL EN NU
OMOVNU = ICQ0 \times \Omega / CNU(1)
BCUT = 2.0 \times OMOMVNU
\text{IF}(BCUT \leq \text{TEMPOB}) \text{ BCUT = TEMPOB}
\text{L} = (PLASCN*EN(1)/CNU(1))/\Omega/\Omega**2+CNU(1)**2)
\text{IF}(\text{IUNS} \times \text{NE} \times 0) \text{ B = B+}
\text{IF}(\text{PLASCN*EN(1)/CNU(1)}) \Omega/\Omega**2+CNU(1)**2
\text{MRATIO}(2))
\text{IF}(\text{PLASCN*EN(3)/CNU(3)}) \Omega/\Omega**2+CNU(3)**2
\text{MRATIO}(3))
\text{IF}(\text{Jays} \times \text{BCUT} \times \text{AND} \times \text{FN}(1) \times \text{GT} \times \text{TUPEN}) \text{ GO TO 561}

580 HROT = B
\text{HR} = \text{PLASCN*EN}(1)/CNU(1)
\text{IF}(\text{IUNS} \times \text{NE} \times 0) \text{ WR = WR+}
\text{PLASCN*EN}(2)/CNU(2)*MRATIO(2))
\text{PLASCN*EN}(3)/CNU(3)*MRATIO(3))
PRINT 60b

60b FORMAT(\"C\", \"\text{WARNING} -- THE TOP OF THE PROFILE CAN NOT BE CUT OFF.\",\"
\text{THE TOP OF THE PROFILE MAY BE TOO LOW.}\")
PRINT 61c,b,w,r

61b FORMAT(\"C\", \"\text{TEMPO-B AT TOP = \,1PE11.3,} \text{5X} \text{WAITS OMEGA-R AT \,1PE11.3}\")
GO TO 94c

HERE THE VALUE OF --B-- AT THIS HEIGHT IS GREATER THAN TEMPO-581 IF(J \times \text{NC} \times \text{NR A}) \text{ GO TO 583}

C
C IF THE HEIGHT IS THE BOTTOM OF THE PROFILE
PRINT 60c,b,TEMPO

60c FORMAT(\"C\", \"\text{THE TOP OF THE PROFILE CAN NOT BE SET. \,1PEX,} \text{2X,}"
\text{AT THE BOTTOM OF THE PROFILE, B = \,1PE13.3, WHICH\",}"
\text{IS GREATER THAN THE VALUE OF TEMPU-B = \,1PE13.3.}\")
\text{CHECK THE INPUT VALUES OF --TEMPO-- AND --TUPEN-- \,1X,}"
\text{ALSO CHECK THE VALUES OF THE ELECTRON AND ION DENSITIES \,1X}"
\text{AT THE BOTTOM OF THE PROFILE.}"
\text{30X, ***PROGRAM EXECUTION STOPS***}

C
STOP 1111
C

***INTERPOLATION***

503 BTOP = B
HTTOP = HTS(J)
HTROT=HTS(J+1)
54 HT=(HTTOP+HTBOT)/2.0
CALL EN NU
h=IPLASCN*EN(1)*CNU(1)/(OMEGA*(OMEGA**2+CNU(1)**2))
1(IGNS .EQ. 0) H2=0.0
1(IGNS .NE. 0) H2=
I(PLASCN*EN(2)*CNU(2))/(OMEGA*(OMEGA**2+CNU(2)**2)*MRATIO(2))
$*(PLASCN*EN(3)*CNU(3))/(OMEGA*(OMEGA**2+CNU(3)**2)*MRATIO(3))
H=I*H2
I:HTTOP-HTROT &LT. 0.10) GO TO 566
IF(R .GT. FCUT .AND. EN(1) .GT. TUPEN) GO TO 565
94HT = R
HTROT = HT
GO TO 64
565 HTTOP=HT
HTTOP=HT
GO TO 64
56b WR=PLASCN*EN(1)/CNU(1)
1(IGNS .NE. 0) WR=WR+
$PLASCN*EN(2)/(CNU(2)*MRATIO(2))
$+PLASCN*EN(3)/(CNU(3)*MRATIO(3))
PRINT 600
600 FORMAT ('106,' ,THE TOP OF THE PROFILE IS SET TO BL---1)
1(IGNS .EQ. 0) PRINT 906,HT,EN(1)
936 FORMAT ('106,' ,P7.2,5x,1PE9.2,1PE9.2,1PE9.2)
1(IGNS .NE. 0) PRINT 906,HT,EN(1),EN(2),EN(3)
PRINT 614,BOTR,WR
614 FORMAT ('106,' ,AT THE TOP IF THE PROFILE R = ' ,1PE11.3, 
$' ,OMEGA-R = ' ,1PE11.3)
HTS(1) = HT
ALGEN(1,1) = ALOG(EN(1))
1(IGNS .EQ. 0) GO TO 67
ALGEN(1,2) = ALOG(EN(2))
ALGEN(1,3) = ALOG(EN(3))
67 CONTINUE
1(IG .EQ. 1) GO TO 900
JN=1
JP1 = J+1
DO 69 JJ = JP1, NR A
JN=JN+1
HTS(JN) = HTS(JJ)
DO 66 K = 1,3
68 ALGFN(JN,K) = ALGEN(JJ,K)
NK A = JN
900 CONTINUE

C
C
C******************************************************************************
C
**BOTTOM CUT OFF**
C
C
THIS SECTION CUTS OFF THE BOTTOM OF THE PROFILE.
II (N**2 = 1-A-JB), THAN THOSE PROFILE DENSITIES WHICH COMPUTE
TO "R" LFFS THEN "CUTOFF" ARE NEGLECTED
II THE PROFILE BOTTOM IS TO BE CUTOFF THEN "ICUT= ",
IF NOT THEN ICUT=0
IF (ICUT .LT. 1) GO TO 702
C
C(0 990 JRFV = 1, NR A
J=NR A-JRFV+1
HT=HTS(IJ)
JAY=J
IF(JAY .EQ. NR A) JAY =NR A-1
CALL EN NU
B1=(PLASN*EN(1)*CNU(1))/(OMEGA*(OMEGA**2+CNU(1)**2))
IF(IONS .LT. 0) B2=0.0
IF(IONS .LT. 0) B2 =
$ (PLASN*EN(2)*CNU(2))/(OMEGA*(OMEGA**2+CNU(2)**2)*MRATIO(2))
2:n1+B2
IF(P .GE. CUTOFF) GO TO 6300
B3=0.0
CONTINUE
C
STOP 2222
C
6300 IF(IJ .NE. NR A) GO TO 6301
PRINT 704,CUTOFF,HTS(NK A),B
704 FORMAT('C', 'THE BOTTOM OF THE PROFILE CAN NOT BE CUT OFF.',
' USING CUTOFF = ',E13.3,' AT HEIGHT = ',F10.3,
' BECAUSE THE VALUE OF B AT THE BOTTOM OF THE PROFILE IS 
',F13.3,' IS LARGER THAN CUTOFF')
IF(IONS .LT. 0) GO TO 4111
PRINT 4112,B1,B2,B
4112 FORMAT('THE VALUES ARE B-ELECTRONS = ',1PF11.4,' B-IONS = ',
1PF11.4,' B-TOTAL = ',1PF11.4)
4111 CONTINUE
GO TO 702
C
***INTERPOLATION***
C
631 B1=0.3
HTOP=HTS(I)
HTBT=HTS(J+1)
304 HT=(HTOP+HTBT)/2.0
CALL EN NU
B1=(PLASN*EN(1)*CNU(1))/(OMEGA*(OMEGA**2+CNU(1)**2))
IF(IONS .LT. 0) B2=0.0
IF(IONS .LT. 0) B2 =
$ (PI * ASCA * EN(2) * CNU(2)) / (OMEGA * (OMEGA ** 2 + CNU(2) ** 2) * MRATIO(2))$
$ + (PI * ASCA * EN(3) * CNU(3)) / (OMEGA * (OMEGA ** 2 + CNU(3) ** 2) * MRATIO(3))$
$F = R + H$
$IF (111. T11 - HBL.T .LT. J. + 0. S3 T10 350$
$IF (JU. LT. CUTOFF) GC TU 305$
$B(I1) = B$
$HT9.HT = HT$
$GU 12 364$
$305 BTUP = HT$
$GO TO 364$
$364 CONTINUE$
$A1 = EN(1)$
$IF (IONS .EQ. 0) GO TO 691$
$A2 = FN(2)$
$A3 = EN(3)$
$GU TO 662$
$662 A2 = 0.0$
$A3 = 0.0$
$662 CONTINUE$
$PRINT 703, HT, CUTOFF$
$703 FORMAT(*', *THE BOTTOM PROFILE HEIGHT IS = ', F10.3,$$ AT B(CUTOFF) = ', F12.3, ')$
$IF (PRINT .EQ. 0) GO TO 411$
$PRINT 4114, A1, A2, A3, B2, 3$
$411+ FORMAT(*', ',$$ THE ELECTRON, POSITIVE ION AND NEGATIVE ION DENSITIES$, $$ ARE EQUAL TO ', '1X, 1PE10.3, 3X, 1PE10.3, 3X, 1PE10.3, //$$ 1X, 1PE10.3, //$$ 1PE10.3, ', '1PE10.3, ', '1PE10.3, $'$$ B-TOTAL = ', '1PE10.3, ')$
$4113 CONTINUE$
$NR A = J + 1$
$HTS(NR A) = HT$
$ALOGEN(NR A, 1) = ALCG(EN(1))$
$IF (IONS .EQ. 0) GU TO 677$
$ALOGEN(NR A, 2) = ALCG(EN(2))$
$ALOGEN(NR A, 3) = ALCG(EN(3))$
$677 CONTINUE$
$C$702 CONTINUE$
$C$
$C$PRINT 913$
$913 FORMAT(*', ')$
$C$C$C$C**THIS SECTION IS USED TO DETERMINE THE PROFILE HEIGHT WHERE$$ THE VALUE OF OMEGA-R EQUALS 2.5E05$$ THE VARIABLE 'WR' IS USED AS WAIT'S OMEGA-P$$ C$DO 750 J REV=1, NR A$
$J=NR A-J REV+1

159
HT=HTS(J)
JAY=J
IF(J .NE. NR A) JAY = NR A-1
CALL EN Nu
WR=PLASCN*EN(1)/CNU(1)
IF(1UNS .NE. 0) WR=WR+
$ PLASCN*EN(2)/(CNU(2)*MRATIO(2))
$ +PLASCN*EN(3)/(CNU(3)*MRATIO(3))
IF(WR .GT. OMEGAK) GO TO 701
DRT=R
790 CONTINUE
HCF#P=HTS(1)
PRINT 3608,HOFWR
3608 FORMAT(C*,5X,'WARNING THE TOP OF THE PROFILE IS TO LOW',
*, 'FOR OMEGA-R TO BE EQUAL TO 2.5E05 */5X,'THE VALUE OF',
*, 'TOP-HEIGHT IS THEREFORE TAKEN AS THE HEIGHT AT THE TOP OF',
*, 'THE PROFILE',//10X,'TOP HEIGHT = ',F7.2)
GO TO 9060
C
701 IF(J .NE. NR A) GO TO 703
HCFWR=HTS(NR A)
PRINT 3603,HOFWR
3603 FORMAT(C*,5X,'THE VALUE OF THE COMPUTED ---OMEGA-R--- AT THE',
$, 'BOTTOM OF THE PROFILE ---HOFWR = ',F7.2,' --- */5X,'IS ',
$, 'GREATER THAN THE INPUT LIMIT OF ---OMEGA-R---)
GO TO 9060
C
C
***INTERPOLATION***
C
713 BTOP=B
HTTOP=HTS(J)
HTBOT=HTS(J+1)
764 HT=(HTTOP+HTBOT)/2.0
CALL EN Nu
WR=PLASCN*EN(1)/CNU(1)
IF(1UNS .NE. 0) WR=WR+
$ PLASCN*EN(2)/(CNU(2)*MRATIO(2))
$ +PLASCN*EN(3)/(CNU(3)*MRATIO(3))
IF(HTTOP-HTBOT .LT. 0.0) GO TO 766
IF(WR .GT. OMEGAR) GO TO 765
BBOT=B
HTBOT=HT
GO TO 764
765 BTOP=B
HTTOP=HT
GO TO 764
766 HOFWR=HT
PRINT 3600,HOFWR
3630 FORMAT(C*,5X,'OMEGA-R EQUALS 2.5E05 AT TOP HEIGHT = ',F7.2)
9300 CONTINUE
RETURN
C
9J1 FORMAT (2014)
902 FORMAT (10,20A4)
SUBROUTINE FINAL

C FOR MS

C ROUTINE FOR FOLLOWING THE EIGFRA-ANGLES AS THE F FUNCTION IS CHANGED
FROM THAT USING LAGRANGE INTERPOLATION TO THAT USING
EXACT FULL-WAVE SOLUTION VALUES OF REFLECTION

C COEFFICIENTS.

C COMMON/XACT C/ XACT
COMMON/THETA C/THETA
COMMON/SEP COM/SEP
COMMON/TOL COM/TOL
COMMON/EIG COM/EIGEN(JO),X E
COMMON/MPRINT/MPRINT
COMMON/DERIV C/DERIV
DIMENSION FRACTS(32)
CCMPI EX

$ THETA,
$ SAVE T,
$ EIGEN,
$ F,DART,DELT,
$ DFIDT
DATA MAX NK/6/

C
C IF (MPRINT NE. 0) PRINT 900
900 FORMAT ('1',ITERATIONS FOR EXACT SOLUTION')
IDERV = 1

C DC 19 JE=1,NK E
10 FRACTS(1) = 0.0
FPAClS(2) = 1.0
NK STPS = 2
NS = 2
IF (MPRINT NE. 0) PRINT 901,JE
901 FORMAT ('+',18)
THETA = EIGEN(JE)
11 SAVE T = THETA
NCOUNT = 0
FE FRCT = FRACTS(NS)
FI FRCT = 1.0-FE FRCT
12 IF (MPRINT NE. 0) PRINT 902,THETA
902 FORMAT ('+',10X,'THETA =',2E9.4)
KXACT=1
CALL FDFT (THETA,F,DFDT)
IF (MPRINT NE. 0) PRINT 104,F,DFDT
104 FORMAT ('+',60X,'EXACT F =',2E11.3,3X,2E11.3)
IF (NS NE. NR STPS) GO TO 14

C
13 KXACT =0
CALL FDFT (THETA,FI,DFIDT)
IF (MPRINT NE. 0) PRINT 105,FI,DFIDT
105 FORMAT ('+',60X,'APPRX F =',2E11.3,3X,2E11.3)
F = F1 FRCT*FI+FE FRCT*FR
DFDT = F1 FRCT*DFIOT+FE FRCT*DFDT
IF (MPRINT .NE. 0) PRINT 106,F,DFDT
106 FORMAT (' ',AOX,'COMBA F = ',2E11.3,3X,2E11.3)
14 D F T = -F/DFDT
THETA = THETA+DELT
NCOUNT = NCOUNT+1
11 (CABS(DELT) .GT. SEP .AND. NCOUNT .GE. 2) GO TO 20
11 (CABS(DELT) .LT. .3*TOL) GO TO 16
IF (NS .NE. NR STPS .AND. CABS(DELT) .LT. SEP) GO TO 16
GO TO 12
C
20 IF (MPRINT .NE. 0) PRINT 906
906 FORMAT ('O','RESET')
IF (NR STPS .GE. MAX NR) GO TO 17
THETA = SAVE T
DO 22 NREV=NS,NR STPS
N = NR STPS-NREV+NS
22 FRACTS(N+1) = FRACTS(N)
FRACTS(NS) = (FRACTS(NS-1)+FRACTS(NS+1))/2.0
NR STPS = NR STPS+1
GO TO 11
C
16 IF (MPRINT .NE. 0) PRINT 907
907 FORMAT ('O')
NS = NS+1
IF (NS .LE. NR STPS) GO TO 11
EIGEN(JE) = THETA
IF (MPRINT .NE. 0) PRINT 902,THETA
GO TO 19
17 PRINT 109,THETA
109 FORMAT (O,'INVALID MODE AT',F9.4,'TRY SMALLER RECTANGLE AROUND'
$ THIS AREA WITH APPROPRIATE RANGER AND RANGE VALUES AND '
$ 'RE-EXECUTE')
NR E = NR E+1
IF(NR E .EQ. 0) GO TO 33
DO 18 JJ=JE,NR E
18 EIGEN(JJ) = EIGEN(JJ+1)
GO TO 10
19 CONTINUE
C
IF (NR E .LE. 1) GO TO 33
!STOP = 0
NREM1 = NR E-1
DO 32 J=1,NREM1
JPL = J+1
DO 32 JJ=JPL,NR E
IF (CABS(EIGEN(J)-EIGEN(JJ)) .GT. TOL) GO TO 32
PRINT 905,J,JJ
32 PRINT 905,J,JJ
905 FORMAT ('O','EIGENS NR*,I3,1X,'AND*,I3,1X,'ARE TO BE RESOLVED'',/
$ 'TRY SMALLER RECTANGLE AROUND'
$ THIS AREA WITH APPROPRIATE RANGER AND RANGE VALUES AND RE-EXECUTE
$')
PRINT 302, EIGEN(J), EIGEN(JJ)
302 FORMAT ('O', 'EIGEN ANGLES AT', '2F9.4, 1X, 'AND', '2F9.4, 1X, 'ARE TO BE R
$ESOLVED')
ISTOP = 1
32 CONTINUE
IF (.NOT. ISTOP) STOP
33 RETURN
C
90 PRINT 990
990 FORMAT ('O', 'PROBLEMS IN SUBROUTINE FINAL')
STOP
C
END
SUBROUTINE NEOUT
C
C ROUTINE PROVIDING FOR PUNCHED OUTPUT OF EIGEN ANGLES
C
COMMON/EIG,COM/EIGEN(100),NR E
COMMON/HTS,COM/HTS(101)
COMMON/NR OF AS/NR A
COMMON/CCM/H
COMMON/FREQ,COM/FREQ
COMPLEX EIGEN,
S$THETA,$
S$S,S0
DATA PI/3.141592/
DATA RTO/57.29578/
DATA VEL LT/2.997928E5/
DATA RE/6369.427/
C
WAVE NR = 2.0*PI*FREQ*L%UO.0/VEL LT
ACOEF = -8.6859*WAVE NR*1000.0
CAP K = 1.0/((1.0-H/RE)
C
PUNCH 971,HTS(NR A)
971 FORMAT (1X,'D = ',F7.2,')
PRINT 972,HTS(NR A)
972 FORMAT ('O',1X,'D = ',F7.2,')
PUNCH 975
975 FORMAT (1X,'EIGEN = ')
PRINT 976
976 FORMAT (' ',1X,'EIGEN = ',2X,'ATTEN WVERC')
DO 77 JE=1,NR E
THETA = EIGEN(JE)
PUNCH 977,THETA
977 FORMAT (1X,2(F8.3,'))
S = 2*SIN(THETA/RTO)
S0 = S*CAP K
ATTEN = ACOEF*AIMAG(S0)
WVERC = 1.0/REAL(S0)
PRINT 978,THETA,ATTEN,WVERC
77 CONTINUE
978 FORMAT (' ',1X,2(F8.3,')',F15.3,F9.5)
THETA = 0.0
PUNCH 977,THETA
PRINT 978,THETA
RETURN
C
END
SUBROUTINE NP OUT
C FOR MS
C ROUTINE FOR COMPUTING AND PRINTING TABLE AND PUNCHED CARD OUTPUT
ASSOCIATED WITH NPUNCH OPTION. FURTHER DESCRIPTION OF OUTPUT IS GIVEN IN A DNA REPORT.

COMMON/IC COM/IDENT(20)
COMMON/MPRINT/PRINT
COMMON/NPUNCH C/NPUNCH
COMMON/CRD PST/CARDPT
COMMON/CRD PNC/CARDPN
COMMON/RHO COM/RHO, WALT FH
COMMON/FLD COM/ALIM, C0DIP, MAGFLD
COMMON/GAD CCM/EPSPR, SIGMA
COMMON/FREQ COM/FREQ
COMMON/H CCM/H
COMMON/WAV/WAVE NR
COMMON/HOF/HOFWR
COMMON/N/THETA C/THETA
COMMON/FIG COM/EIGEN(1U), NR E
COMMON/INTEGR/X11, X21, X12, X22
COMMON/F8 CCM/NUM11, NUM22, DEN11, DEN22
COMMON/F2 C/F1, F2, HG, F1ZT, F1ZR, F2ZT, F2ZR, GF1ZT
COMPLEX
  $ THETA, EIGEN,
  X11, X21, X12, X22,
  NUM11, NUM22, DEN11, DEN22,
  F1, F2, HG, F1ZT, F1ZR, F2ZT, F2ZR, GF1ZT,
  C,
  R11, R21, R12, R22,
  RBAR11, RBAR22, D11, D22,
  THETAP, TERM1, TERM2, TERM3, TERM4,
  RB11P1, RB22P1,
  FM, DFMOT,
  UFDT, FACTOR,
  S, CP, CASIN,
  W1, Q2, J3, FOVRD, FOVXO,
  KXZ, KEZ, KBZ,
  SP, STURF1, ECUMP, EXLOG
REAL MAGFLD
INTEGER CARDPT, CARDPN
DATA RE/6369.427/
DATA RTD/57.29578/
DATA ALN TEN/2.302585/
DATA PI/3.141593/
DATA RTD/57.29578/

C
C PRINT/PUNCH 'ID' INFORMATION
IF (MPRINT NE. 0) PRINT 901, IDENT
901 FORMAT(20A4)
IF(CARDPN .GT. 0) PUNCH 902, IDENT
902 FORMAT(20A4)
C PUNCH WHO CARD, PRINT H VALUE AND TABLE HEADINGS.
   IF (PRINT, NE, 0) PRINT 971, KHO, FREQ, AZIM, CODIP, MAGFDI, SIGMA, EPSR
971 FORMAT ('10', F8.3, F1,F6.4, A$, F8.3, C$, F8.3, M', 1PE10.3, 
     S', E10.3, E', 0PF5.1)
   IF(CARDNP .GT. 0)
PUNCH 972, KHO, FREQ, AZIM, CODIP, MAGFDI, SIGMA, EPSR
972 FORMAT ('10', F8.3, F1, F6.4, A$, F8.3, C$, F8.3, M', 1PE10.3, 
     S', E10.3, E', 0PF5.1)
   PRINT 974, H
974 FORMAT ('10', F8.3, H = ', F8.2)
   PRINT 975
975 FORMAT ('10', 1X, 'MODE', 1X, 'THETA', 6X, 'ATTEN', 3X, 'VUVERC', 3X, 
     'WAIT MAG', 2X, 'WAIT ANG', 8X, 'THETAP', 13X, 'POL')
   IF(NPUNCH .EQ. 2 .AND. CARDNP .EQ. 1) PUNCH 93Y, FREQ
939 FORMAT(F8.3)
C
C FORMAT OF THE USUAL REFLECTION MATRIX, R, FROM (R+1.0)/C AND
C FORMATION OF THE USUAL RBAR MATRIX VALUES FROM
C FORM USED TO FIND EIGEN ANGLES. ALSO, COMPUTATION OF
C DERIVATIVES FOR USE IN COMPUTING DFDTHETA.
C
C CALL DFDT (THETA, FM, DFMDT)
C C = COS(THETA/RTD)
R11 = C*X11-1.0
R21 = C*X21
R12 = C*X12
R22 = C*X22+1.0
D11 = C*NUM11-DEN11
D22 = C*NUM22-DEN22
RBAR11 = DEN11/D11
RBAR22 = DEN22/D22
KB11P1 = C*NUM11/D11
RB22P1 = C*NUM22/D22
C
DFDT = DFMDT*C**2/(D11*D22)*RTD
S = CSIN(THETA/RTD)
FACTOR = CSQRT(S)/DFDT
C
ITERM=1
IF(CABS(1.0-RBAR11) .GE. CABS(2.0-RBAR12)) ITERM=2
C VLF-LF EXCITATION TERMS (E.G. THE **T** TERMS TO BE USED FOR
C NPUNCH=1 OR NPUNCH = 9).
C TERM1 = RR11P1**2*(1.0-RBAR22*R22)*FACTOR/(RBAR11*F1/F1)
TERM2 = R822P1**2*(1.0-RBAR11*R11)*FACTOR/(RBAR22*F2/F2)
TERM3 = R811P1*R822P1*FACTOR*R21/(F1*F2)
TEAM4

TERM4 = F12/R21

C

ELF EXCITATION TERMS (E.G. THE TERMS TO BE USED FOR NPUNCH = 7).

XVZ = TERM1*S*F12*F12K
XZL = -TERM1*S*GF*F12T*F12K
XGZ = -TERM3*TERM4*S*F2Z*T*F12D

C

IN TERMS OF MAGNITUDE AND PHASE THESE TERMS ARE GIVEN BY

XVLM = CABX(VXZ)
XVZA = CANG1(XVZ)
XZLM = CABX(XZL)
XBLM = CANG1(XGZ)
XBLA = CANG1(XBL)

C

EIGEN ANGLE REFERRED TO GROUND LEVEL.

RATIO = F/RE
CP = COSRT(C**2 + (C**2 - 1.0) * (2.0*RATIU**3.0*RATIU**2))
THETAP = 90.0 - CASIN(CP) * KTO

C

ATTENUATION AND PHASE VFLUCIT Y REFERRED TO GROUND LEVEL.

SP = S/(1.0 - H/RE)
ATTEN = AC0EF*AIMAG(SP)
VOVERC = 1.0/REAL(SP)

C

WAIT'S EXCITATION.

STORFL = RB11P1**2*(1.0-RBAR22*R22)

*FACTOR/RBAR11

ECOMP = H**2*STORE1**2

EXLOG = CLCIEG( ECOMP*0.5*WAVE Nk*WAIT RH)

WM = 20.0*REAL(EXLOG)/ALN TEN

WA = AIMAG(EXLOG) - PI/2.0

IF (WA < IT< PI/2.0) WA = WA+2.0*PI

C

MODE CONVERSION TERMS TO BE USED FOR NPUNCH = 8 AND NPUNCH = 9.

XM = CABX(ECOMP)
XA = CANG1(ECOMP)
Q1 = 1.0 - R11*RBAR11
Q2 = 1.0 - R22*RBAR22
Q3 = R82*PI/RBAR11

IF (CABS(Q1) < CABS(Q2)) FOVRD = Q3*Q1/(R12*RBAR22)

IF (CABS(Q2) < CABS(Q1)) FOVRD = Q3*(R21*RBAR11)/Q2

FUVRS = (FOVRD*F1)/F2

C

POLARIZATION.

Q1 = CABX(R21*RBAR22)
Q2 = CABX(1.0-RBAR22*R22)

IF (Q1 < Q2) POL = CABX(1.0-RBAR22*R22)/Q1

IF (Q2 < Q1) POL = CABX(K12*RBAR11)/Q2

C

PRINT VALUES FOR TABLE.

C

MODE = MCDE+1

PRINT 932,MODE,THETA,ATTEN,VOVERC,WM,WA,THETAP,POL
NPUNCH * OPTION FOR OBTAINING OUTPUT CARDS

NPUNCH=1*** GIVES OUTPUT CARDS FOR HORIZONTALLY HOMOGENOUS
MODE-SUM OR WKB-SUM IN TERMS OF (T'S)

NPUNCH=2*** GIVES OUTPUT CARDS IN TERMS OF SNVLF OUTPUT

NPUNCH=7*** GIVES OUTPUT CARDS FOR INPUT INTO ELF-WKB MODE-SUMS
IN TERMS OF (FOVR)

NPUNCH=9*** GIVES OUTPUT CARDS FOR INPUT INTO MODE CONVERSION
IN TERMS OF BOTH (T'S) AND (FOVR)

CARD PN=0*** GIVES USUAL MODE SUMMARY PRINT OUT BUT NO CARDS
CARD PN=1*** GIVES USUAL MODE SUMMARY PRINT AND CARDS
CARD PT=0*** CAUSES THE OUTPUT CARDS TO NOT BE LISTED
CARD PT=1*** CAUSES THE OUTPUT CARDS TO BE LISTED

PUNCH *NPUNCH *** OPTION CARDS----'MPRINT' USED ONLY FOR DEBUGGING

IF (MPRINT .NE. 0) PRINT 933, THETAP, TERM1, TERM2,
   $ ABORT, TERM3, TERM4
933 FORMAT ('O', 'I', '0P2F9.5, 0P1E15.8', '/', '2', '0P2F9.5, 0P1E15.8')
IF (MPRINT .NE. 0) PRINT 930
930 FORMAT ('')

IF (NPUNCH .EQ. 1 .AND. CAROPT .EQ. 1) PRINT 930
IF (NPUNCH .EQ. 1 .AND. CAROPT .EQ. 1)
   * PRINT 931, THETAP, TERM1, TERM2, THETAP, TERM1, TERM2, TERM3, TERM4
931 FORMAT ('**', 'I', '0P2F9.5, 1I, 0P1E15.8', '/', '2', '0P2F9.5, 1I, 0P1E15.8')
IF (NPUNCH .EQ. 1 .AND. CAROPT .EQ. 1)
   * PUNCH 934, THETAP, TERM1, TERM2, THETAP, TERM1, TERM2, TERM3, TERM4
934 FORMAT (**', 'I', '0P2F9.5, 1I, 0P1E15.8', '/', '2', '0P2F9.5, 1I, 0P1E15.8')
IF (NPUNCH .EQ. 1) GO TO 900
IF (NPUNCH .EQ. 1 .AND. CAROPT .EQ. 1)
   * PRINT 937, ATTN, VORC, WM, WA
937 FORMAT ('O', '4F10.5')
IF (NPUNCH .EQ. 2 .AND. CAROPT .EQ. 1)
   * PUNCH 938, ATTN, VORC, WM, WA
938 FORMAT ('O', '4F10.5')
IF (NPUNCH .EQ. 2) GO TO 600
IF (NPUNCH .EQ. 7 .AND. CAROPT .EQ. 1) PRINT 930
IF (NPUNCH .EQ. 7 .AND. CAROPT .EQ. 1)
   * PRINT 626, THETAP, XMZ, XVZ, XZM, XZM, XZM, XZM, XZM
626 FORMAT ('2(F10.5, 3(IPE13.4, 9.*), 0PF8.3, 9.*),')
IF (NPUNCH .EQ. 7 .AND. CAROPT .EQ. 1)
   * PUNCH 627, THETAP, XMZ, XVZ, XZM, XZM, XZM, XZM, XZM
627 FORMAT ('5X, 2(F10.5, 3(IPE13.4, 9.*), 0PF8.3, 9.*),')
IF (NPUNCH .EQ. 7) GO TO 600
IF (NPUNCH .EQ. 8 .AND. CAROPT .EQ. 1) PRINT 930
IF (NPUNCH .EQ. 8 .AND. CAROPT .EQ. 1)
   * PRINT 935, THETAP, XM, XA, VORC, WM, WA

169
935 FORMAT(9F2F9.5,1PE14.5,OPF9.5,1PE16.8,OPF7.2)
   IF(NPUNCH .EQ. 8 .AND. CARDPN .EQ. 1)
   *PUNCH 930, THETAP, XM, XA, FUVR0, HOFWR
936 FORMAT(9F2F9.5,1PE14.5,OPF9.5,1PE16.8,OPF7.2)
   IF(NPUNCH .EQ. 9 .AND. CARDPN .EQ. 1)
   IF(NPUNCH .EQ. 9 .AND. CARDPT .EQ. 1)
   PRINT 931, THETAP, ITERM, TERM1, TERM2, THETAP, ITERM, TERM3, TERM4
   IF(NPUNCH .EQ. 9 .AND. CARDPT .EQ. 1)
   PRINT 935, THETAP, XM, XA, FUVR0, HOFWR
   IF(NPUNCH .EQ. 9 .AND. CARDPT .EQ. 1)
   *PUNCH 934, THETAP, ITER4, TERM1, TERM2, THETAP, ITETM, TERM3, TERM4
   IF(NPUNCH .EQ. 9 .AND. CARDPN .EQ. 1)
   *PUNCH 936, THETAP, XM, XA, FUVR0, HOFWR
600 CONTINUE
79 CONTINUE
C
   IF(CARD PN .GT. 0) PUNCH 760
700 FORMAT('**')
   RETURN
C
   END
FUNCTION CASIN(S)
C ROUTINE FOR COMPUTING THE COMPLEX SINE FUNCTION OF A COMPLEX
ARGUMENT.
C FOR MS

COMPLEX CASIN,S,SSQ,SPOWER,SUM,TERM,PARTS,C,CPLX 1
DATA CPLX 1/(0.0,1.0)/
DIMENSION PART(2)
EQUIVALENCE (PART(1), PARTS)

IF (CABS(S) .GT. 0.7) GO TO 20
FACTOR = 1.0
SPOWER = S
SSQ = S**2
SUM = S
IF (CABS(SUM) .LT. 1.0E-7) GO TO 12
DO 11 K=3,99,2
FACTOR = FACTOR*(K-2.0)/(K-1.0)
SPOWER = SPOWER*SSQ
TERM = FACTOR*SPOWER/K
SUM = SUM+TERM
PARTS = TERM/SUM
IF (ABS(PART(1))+ABS(PART(2)) .LT. 1.0E-10) GO TO 12
11 CONTINUE
12 CASIN = SUM
RETURN

20 C = CSQRT(1.0-S**2)
CASIN = -CPLX I*CLOG(C+CPLX I*S)
RETURN

END
FUNCTION CANS(ARG)
IMPLICIT REAL (A-H,O-Z)
COMPLEX ARG, MINUSI/(0., -1.)
ARGR=ARG
ARGI=MINUSI*ARG
CANG=ATAN2(ARG1, ARG3)
IF(AARGI .LT. 0.) CANG= CANG + 6.283165
RETURN
END
SUBROUTINE INTEG
C FOR MS
C THIS SUBROUTINE PERFORMS AN INTEGRATION OF THE DIFFERENTIAL
C EQUATIONS FOR THE IONOSPHERE REFLECTION MATRIX
C USING RUNG-KUTTA INTEGRATION FORMULAS. THE
C INTEGRATION VARIABLES ARE THE ELEMENTS OF THE MATRIX
C \( (R+1.0)/C \) WHERE \( R \) IS THE REFLECTION MATRIX DESCRIBED
C BY R/JCDEN. IF IDERIV IS SET NON-ZERO, THE DERIVATIVES
C OF \( (R+1.0)/C \) ELEMENTS WRT \( C = \cos(\theta) \) ARE ALSO
C INTEGRATION VARIABLES. THE SET OF HEIGHTS AT WHICH
C PAIRS OF INTEGRATION STEPS BEGIN AND END IS STORED
C IN THE ARRAY 'RKHTS'. SIZES OF THE STEPS ARE
C DETERMINED BY COMPARING THE VALUES OF THE ELEMENTS
C OF \( (R+1.0)/C \) AFTER A PAIR OF STEPS AND VALUES OF THE
C SAME VARIABLES AFTER A SINGLE DOUBLE-SIZE COMPARISON
C STEP. THE \( (R+1.0)/C \) VARIABLES USED IN THE COMPARISON
C STEP ARE DENOTED AS 'X'. IF THE DIFFERENCE IS TOO
C LARGE, THE STEP SIZE PAIRS ARE CUT IN HALF BY ADDING
C A NEW HT TO THE LIST IN 'RKHTS'. IN ORDER TO
C ENHANCE STABILITY IN OTHER PARTS OF THE PROGRAM, NO
C HEIGHTS ARE DELETED FROM THE LIST UNLESS THE LIST IS
C RESET BY A CALL TO 'SET RK'.

COMMON/NR OF AS/NR A
COMMON /RTOL C/RTOL
COMMON/HTS COM/HTS(101)
COMMON/XPRINT/XPRINT
COMMON/INTEG/R(10),DRJH(10)
COMMON/X INTPR/X(8),DXDH(8)
COMMON/EN COLL/HT
COMMON/WAVE NR
COMMON/IDERIV C/IDERIV
COMMON/N C/N
COMMON/VRFLO/IDVFLO
COMMON/JAY COM/JAY
DIMENSICA
$ R(16),HDELRO(16),DELR1(16),DELR2(16),
$ XO(8),HDELXO(8),DELX1(8),DELX2(8),
$ P SAVE(16),
$ RKHTS(201)
INTEGR. XPRINT
DATA MAX NR/201/
DATA DIMIN/0.01/
C
N = 8
IF (IDERIV .NE. 0) N = 16
IDVFLO = 0
IF(XPRINT .NE. 0) CALL R COLS
THIRD = 1.0/3.0
CALL INIT S
JAY = 1
JRK = 1
HT = HTS(1)
CALL S MTRX
CALL INITL R

CALL XFER (R,X,8)
IF (XPRINT NE. 0) PRINT 901
901 FORMAT (*0*)
IF (XPRINT NE. 0) CALL PRINT R

IRAM1 = NR A-1
DO: 79 J=1, NRAM1
     JAY = J
30 CALL XFER(R,R SAVE,N)
     HT = RKHTS(JRK)
     CALL XFER (R,X,8)
31 DELH = (RKHTS(JRK+1)-RKHTS(JRK))*0.5
     IF (ABS(CFLH) .LT. DHMIN) GO TO 90
     DH = DEL*WAVE NR/2.0
     WTH = 0.5*DH
     TCH = 2.0*DH

C BEGIN FIRST RUNGE-KUTTA STEP OF THE PAIR AND THE COMPARISON STEP.
CALL R DERIV
IF (IOVFLC NE. 0) GO TO 60

C
DO 32 I=1,N
     PO(I) = R(I)
     HDELR0(I) = DROH(I)*DH
32 R(I) = R(I)+HDELR0(I)

C
DO 33 I=1,N
     XU(I) = X(I)
     HDELEX(I) = HDELP(I)*2.0
33 X(I) = X(I)+HDELEX(I)

C
     HT = HT+C.5*DELP
     CALL S MTRX
     CALL R DERIV
     IF (IOVFLC NE. 0) GO TO 60

C
DO 34 I=1,N
     DELR1(I) = DROH(I)*DH
34 R(I) = RO(I)+0.5*DELR1(I)

C
CALL R DERIV
     IF (IOVFLC NE. 0) GO TO 60

C
DO 35 I=1,N
     DELR2(I) = DROH(I)*DH
35 R(I) = RO(I)+DELR2(I)

C
     HT = HT+C.D*DELP
     CALL S MTRX
CALL R DERIV
CALL X DERIV
IF(I0VFLC .NE. 0) GO TO 60

C
C DO 30 I=1,N
HDELP3 = DROH(I)*HDH
DELR4 = (HDELRO(I)+DELR1(I)+DELR2(I)+HDELP3)*THIRD
30 R(I) = RC(I)+DELR4

C BEGIN SECOND RUNGE-KUTTA STEP OF THE PAIR.
C DO 37 I=1,8
DEIX1(I) = UXDH(I)*TDH
37 X(I) = X0(I)+0.5*DELA1(I)

C CALL R DERIV
CALL X DERIV
IF(I0VFLC .NE. 0) GO TO 60

C
C DO 46 I=1,N
HDELR0(I) = DRDHI(I)*HDH
46 R(I) = RO(I)+HDELR0(I)

C DO 47 I=1,8
DEIX2(I) = UXDH(I)*TDH
47 X(I) = X(I)+DEIX2(I)

C HT = HT+C.5*DELH
CALL S MTRX
CALL R DERIV
IF(I0VFLC .NE. 0) GO TO 60

C
C DO 48 I=1,N
DELR1(I) = DRDHI(I)*DH
48 R(I) = RO(I)+0.5*DELR1(I)

C CALL R DERIV
IF(I0VFLC .NE. 0) GO TO 60

C
C DO 49 I=1,N
DELR2(I) = DRDHI(I)*DH
49 R(I) = RO(I)+DELR2(I)

C HT = HT+C.5*DELH
CALL S MTRX
CALL R DERIV
CALL X DERIV
IF(I0VFLC .NE. 0) GO TO 60

C
C DO 50 I=1,N
HDELP3 = DROH(I)*HDH
DELR4 = (HDELRO(I)+DELR1(I)+DELR2(I)+HDELP3)*THIRD
50 R(I) = RO(I)+DELR4
C
DO 47 I=1,8
HDELX3 = DXDH(I)*DH
DELY4 = (HDELX3(I)+DELY1(I)+DELY2(I)+HDELX3)*THIRD
47 X(I) = XO(I)+DELY4
C
ESTIMATE TRUNCATION ERROR AND CUT STEP SIZE IF APPROPRIATE.
IF(XPRINT .NE. 0) CALL PRINT R
DO 51 I=1,7,2
ERRSC = (R(I)-X(I))*2+(R(I+1)-X(I+1))*2
RSQ = R(I)**2+R(I+1)**2
IF (PSQ.LT. 1.0) EPPS = ERRSQ/RSQ
IF (ERRSC.GT. RTOL**2) GO TO 61
51 CONTINUE
GO TO 70
C
60 IF (XPRINT .NE. 0) CALL PRINT R
61 IF (XPRINT .NE. 0) PRINT 960
960 FORMAT (' ,OVKFLD!')
IF (NR RK .GE. MAX NR) GO TO 95
IOVFLO = 0
CALL XFER(R SAVE,R,N)
CALL XFER (R SAVE,X,8)
HT = RKHTS(JRK)
CALL S MTRX
C
NRKMK1 = NR RK-1
DO 62 JCOUNT=1,RNKMK1
62 RKHTS(JJ+2) = RKHTS(JJ+1)
RKHTS(JRK+1) = (RKHTS(JRK)+RKHTS(JRK+2))*0.5
NR RK = NR RK+1
GO TO 31
C
70 JRK = JRK+1
IF (ABS(RKHTS(JRK)-HTS(J+1)).GT. 0.001) GO TO 30
79 CONTINUE
RETURN
C
C RESET THE INTEGRATION STEP PAIRS TO BE THE INPUT PROFILE SEGMENTS.
ENTRY SET RK
DO 81 J=1,NR A
81 RKHTS(J) = HTS(J)
NR PK = NR A
RETURN
C
90 PRINT 990
990 FORMAT ('O','STEP TOO SMALL IN INTEGR')
STOP
C
95 PRINT 995
995 FORMAT ('O','TOO MANY STEPS IN INTEGR')
C
STOP
END
SUBROUTINE EN NU
C FOR MS
C COMPUTATION OF ELECTRON DENSITY AND COLLISION FREQUENCY (AND ION
DENSITY IF APPROPRIATE) AS A FUNCTION OF HEIGHT.
C IF PROFILE WAS INPUT FROM FORMATTED CARDS, LOGARITHMIC
INTERPOLATION IS USED. SEE ALSO NOTES IN SUBROUTINE
PRFLIN.
C
COMMON/NR CF AS/NR Δ
COMMON/ICN COMMON/IONS
COMMON/JAY COMMON/J
COMMON/HTS COMMON/HTS(101)
COMMON/HTLOGN/NK NU,HTS NU(50),ALCNU(50,3)
COMMON/ALOGN C/ALOGN(101,3)
COMMON/EXPNU C/CEFNU(3),EXPNU(3)
COMMON/EN COLL/HT,EN(3),CNU(3)
C
DATA JNU/1/
C
FACTR = 1.0/(HTS(J)-HTS(J+1))
NK SPEC = 1
IF (IONS .NE. 0) NR SPEC = 3
F1 = (HT-HTS(J+1))*FACTR
F2 = (HTS(J)-HT)*FACTR
DO 11 K=1,NR SPEC
11 EN(K) = EXP(ALLOGN(J,K)*F1+ALLOGN(J,K+1)*F2)
IF(CEFNU(1) .LT. 0.0) GO TO 15
DC 12 K=1,NR SPEC
12 CNU(K) = CEFNU(K)*EXP(EXPNU(K)*HT)
RETURN
C
15 IF(JNU .GT. NF NU-1) JNU=1
16 IF (HT .LE. HTS NU(JNU) .OR. JNU .EQ. 1) GO TO 17
JNU = JNU-1
GO TO 16
17 IF (HT .GE. HTS NU(JNU+1) .OR. JNU .EQ. NR NU-1) GO TO 18
JNU = JNU+1
GO TO 17
18 DO 19 K = 1, NR SPEC
ALN = ALCNU(JNU,K)+(ALCNU(JNU+1,K)-ALCNU(JNU,K))
* (HT-HTS NU(JNU))/HTS NU(JNU+1)-HTS NU(JNU))
19 CNU(K) = EXP(ALN)
RETURN
C
END
SUBROUTINE R OUT
C FOR MS
C OUTPUT OF RUNGE-KUTTA INTEGRATION VARIABLES, USED ONLY FOR DEBUGGING.
C
COMMON/I NTEG R/R(8)
COMMON/X INTEG/X(8)
COMMON/ENV COL L/HT, EC OMIT(2)
C
ENTRY R COLS
PRINT 900
900 FORMAT('C8X,'HT,10X,'11R11',16X,'11R1',16X,'1H11',17X,'1R1')
RETURN
C
ENTRY PRINT R
PRINT 901,HT,(X(I),I=1,8)
PRINT 901,HT,(R(I),I=1,8)
901 FORMAT(' ',F10.2,4(2X,F4.5,F9.5))
PRINT 902
902 FORMAT('01')
RETURN
C
END
SUBROUTINE XFER (A, B, N)
C FOR MS
C ROUTINE FOR TRANSFERRING ONE ARRAY INTO ANOTHER.
C
DIMENSION A(1), B(1)
C
DO 11 J = 1, N
  11 B(J) = A(J)
  RETURN
C
END
SUBROUTINE INITLR
C FOR MS
C COMPUTATION OF (R+1.0)/C FOR REFLECTION FROM A SHARPLY BOUNDED
C ANISOTROPIC ICSUSPHERE OF SEMI-INFINITE EXTENT
C WHERE R IS THE REFLECTION COEFFICIENT MATRIX AS DEFINED
C BY BUJVEN. SOLUTION IS USED AS INITIAL CONDITION
C FOR RUNGE-KUTTA INTEGRATION. THE SOLUTION IS BASED
C ON RADIO SCIENCE PAPER VOL. 3, AUG. 1968, P. 792.
C DERIVATIVES WRT C=CS3(THETA) ARE ALSO COMPUTED IF
C DERIV IS SET NON-ZERO.
C
C COMMON/THETA C/THETA(2)
C COMMON/XPRINT/XPRINT
C COMMON/M M11,M12,M13,M21,M22,M23,M31,M32,M33
C COMMON/INTEGR/R11,R12,R13,R21,R22,R23,R31,R32,R33
C COMMON/C CCM/C
C COMMON/S CCM/S
C COMMON/IDERV C/IDERV
C COMPLEX
C C, CSQ, S,
C > M11, M21, M31, M12, M22, M32, M13, M23, M33,
C R11, R21, R12, R22, R13, R23, R31, R32, R33,
C QTEMP(4), EPI104(Q21),
C B4, B3, B2, B1, B0,
C D11, D12, D13, D21, D22, D23,
C P(21), T(2),
C DEN, FACTOR,
C D3DC, DQDC(2), DDENDC,
C DB3DC, DB2DC, DB1DC, DB0DC,
C DD11DC, DD12DC, DD13DC, DD22DC,
C UPDC(2), DTDC(2)
C INTEGER XPRINT
C DATA EPI104/(U.707107, U.707107)/
C
C CSQ = C**2
C
C BOOKER QUARTIC COEFFICIENTS AND ITS FOUR ROOTS.
C B4 = 1.0*M33
C B3 = S*(M13*M31)
C B1 = S*(M12*M23+M21*M32-(CSQ*M22)*(M13+M31))
C B0 = (1.0*M11)*(CSQ*M22)*(CSQ*M33)+M12*M23*M31+M13*M21*M32
C
C CALL QUARTC(B4,B3,B2,B1,B0,QTEMP)
C
C SELECTION OF THE TWO ROOTS CORRESPONDING TO UPWARD TRAVELLING
C WAVES AS BEING THOSE OF ANGLE LESS THAN 45 BUT
C GREATER THAN -135 DEGREES IN THE COMPLEX PLANE.
C K = 0
C DD 21 KT=1.4
C IF (REAL(EPI104*QTEMP(KT)) .LT. 0.0) GO TO 21
K = K + 1
IF (K .GE. 3) GO TO 90
Q(K) = CTEMP(K)
21 CONTINUE
IF (K .NE. 2) GO TO 90
C
IF (XPRINT .NE. 0) PRINT 901, THETA, (Q(J), J=1,2)
901 FORMAT ('C', 'THETA = ', 2F9.3, 5X, 'Q = ', 2(E15.4, E15.4), ',
C
C REMAINDER OF THE SOLUTION AND ITS DERIVATIVES.
DO 31 J=1,2
D11 = 1.0 + M11 - Q(J)**2
D12 = M12
D13 = M13 + Q(J)*S
D31 = M31 + Q(J)*S
D32 = M32
D33 = CSS + M33
DEN = D11*D33 - D13*D31
P(J) = (-CL*Q(J)+D13*D31)/DEN
31 T(J) = Q(J)*P(J) - S*(-D11*D32 + D12*D31)/DEN
C
IF (IDERV .EQ. 0) GO TO 50
DSDC = -C/S
DB3DC = CSS*C(M13 + M31)
DB2DC = -2.0*C*(1.0 + M11 + 1.0 + M33)
DB1DC = (DSDC/S)*B1 - S*2.0*C*(M13 + M31)
DB0DC = 2.0*C*(1.0 + M11)*(CSS + M22 + CSS + M33) - M13*M31 - M12*M21
C
DO 41 K=1,2
DQDC(K) = -((DB3DC*Q(K) + DB2DC)*Q(K) + DB1DC)*Q(K) + DB0DC)
41 T(K) = Q(K)*P(K) - S*(-D11*D32 + D12*D31)/DEN
C
DEN = (T(1)*C + P(1))*C + Q(2)) - (T(2)*C + P(2))*C + Q(1))
FACTOR = 2.0/DEN
R11 = (T(1)*C + Q(2)) - (T(2)*C + Q(1)) * FACTOR
R22 = ((T(1)*C + Q(1)) - (T(2)*C + P(2)) * FACTOR
R12 = -((T(1)*P(2) - T(2)*P(1)) * FACTOR
R21 = -(C(1) - Q(2)) * FACTOR
C
IF (IDERV .EQ. 0) RETURN
DDENC = (T(1)*C + P(1))**2.0 + DQDC(2)
$ = (T(1)*DQDC(1)*C + UQDC(1))**2.0 + DQDC(2)
$ = -(T(2)*C + P(2))**2.0 + DQDC(1)
$ = -(T(2)*DQDC(2)*C + UQDC(2))**2.0 + DQDC(1)
C

90 PRINT 965
960 FORMAT('9C', 'PROBLEM IN SORTING Q VALUES')
PRINT 995, THETA, QTEMP
995 FORMAT('9G', 'THETA = ', 'Q = ', '1PE13.3, 1PE11.3')
C
STOP
C
END
SUBROUTINE QUARTC(B4,FOUF B3,SIX B2,FOUR B1,ONE B0,Q)
C FOR MS
C SOLUTION FOR THE ROOTS OF A FOURTH-ORDER POLYNOMIAL (QUARTIC
C EQUATION) TAKEN FROM BURNSIDE AND PANTON (1904) THE
C THEORY OF EQUATIONS. A SUMMARY OF PERTINENT EQUATIONS
C IS GIVEN IN RADIO SCIENCE VOL. 3, AUG. 1958, P. 792-795.

COMPLEX FJLR P3,SIX B2,FOUR B1,ONE B0,Q,B4,
$ H**2, G**2, PRIME, G'PRIME,
$ SQ ROOT,P POS,P,LCG P,
$ CUB RT0,CUB RT1,CUB RT2,OMEGA1,OMEGA2,
$ ROOT P,ROOT Q,ROOT R,
$ TWLV B3,TWLV B2,F,DFQ,DEL Q,FB4

REAL MAG POS,MAG NEG
DIMENSION Q(4),P RI(2)
DATA OMEGA1/(-0.5,0.860254038)/,OMEGA2/(-0.5,-0.860254038)/
EQUIVALENCE(P,P RI(1))

B3 = FOUF B3/(4.0*B4)
B2 = SIX B2/(6.0*B4)
B1 = FOUR B1/(4.0*B4)
B0 = ONE BG/B4

B3 SQ = B3**2
H = B2-B3 SQ
I = B0-4.0*B3*B1+3.0*B2**2
G = R1+B3*(-3.0*B2+2.0*B3 SQ)
H PRIME = -1/12.0
G PRIME = -G**2/4.0-H*(H**2+3.0*H PRIME)

SQ ROOT = CSQRT(G PRIME**2+4.0*H PRIME**3)
P = (-G PPRIIME+SQ ROOT)*0.5
MAG POS = ABS(P RI(1))+ABS(P RI(2))
P POS = P
P = (-G PRIME-SQ ROOT)*0.5
MAG NEG = ABS(P RI(1))+ABS(P RI(2))
IF(MAG POS,GLT,MAG NEG) P = P POS
LOG P = CLG(P)
CUB RT0 = CEXP(LOG P/3.0)
CUB RT1 = OMEGA1*CUB RT0
CUB RT2 = OMEGA2*CUB RT0

ROOT P = CSQRT(CUB RT0-H PRIME/CUB RT0-H)
ROOT Q = CSQRT(CUB RT1-H PRIME/CUB RT1-H)
ROOT R = CSQRT(CUB RT2-H PRIME/CUB RT2-H)
IF(CABS(G) LT. 1.0E-50) GO TO 21
SIGN = -ROOT P*ROOT Q*ROOT R*2.0/G
IF(SIGN LT. 0.0) ROOT R = -ROOT R

Q(1) = +ROOT P*ROOT Q*ROOT R-B3
Q(2) = +ROOT P*ROOT Q*ROOT R-B3
Q(3) = -ROOT P*ROOT Q*ROOT R-B3
SUBROUTINE QUARTC(B4,FOUR B3,SIX B2,FOUR B1,ONE B0,Q)
  C FOR MS
  C SOLUTION FOR THE ROOTS OF A FOURTH-ORDER POLYNOMIAL (QUARTIC
  EQUATION) TAKEN FROM BURNSIDE AND PANTON (1934) THE
  THEORY OF EQUATIONS. A SUMMARY OF PERTINENT EQUATIONS
  IS GIVEN IN KADIS SCIENCE VOL. 3, AUG. 1966, PP. 792-795.

  COMPLEX FULK P3,SIX B2,FOUR B1,ONE B0,Q,B4,
  $ M1,G,H PRIME,G PRIME,
  $ SQ ROOT,P POS,P,LLG P,
  $ CUB RT0,CUB RT1,CUB RT2,OMEGA1,OMEGA2,
  $ ROOT P,ROOT Q,ROOT R,
  $ TWLV B3,TWLV E2,F,DOQ,DEL Q,B4

  REAL MAG POS,MAG NEG
  DIMENSION Q(4),P RI(2)
  DATA OMEGA1/(-0.5,0.860254058),OMEGA2/(-0.5,-0.860254058)/

  B3 = FOUR B3/(4.0*B4)
  B2 = SIX B2/(6.0*B4)
  B1 = FOUR B1/(4.0*B4)
  B0 = ONE B0/B4

  B3 SQ = R3**2
  H = B2-B3 SQ
  I = B0-4.0*B3*B1+3.0*B2**2
  G = P1+R3*(-3.0*B2+2.0*B3 SQ)
  H PRIME = -1/12.0
  G PRIME = -G**2/4.0-H*(H**2+3.0*H PRIME)

  SQ ROOT = CSQRT(G PRIME**2+4.0*H PRIME**3)
  P = (-G PRIME+SQ ROOT)*0.5
  MAG POS = ABS(P RI(1))+ABS(P RI(2))
  P POS = P
  P = (-G PRIME-SQ ROOT)*0.5
  MAG NEG = ABS(P RI(1))+ABS(P RI(2))
  IF(MAG PCS .GT. MAG NEG) P = P POS
  LOG P = CLOG(P)
  CUB RT0 = CEXP(LOG P/3.0)
  CUB RT1 = OMEGA1*CUB RTO
  CUB RT2 = OMEGA2*CUB RTO

  ROOT P = CSQRT(CUB RT0-H PRIME/CUB RT0-H)
  ROOT Q = CSQRT(CUB RT1-H PRIME/CUB RT1-H)
  ROOT R = CSQRT(CUB RT2-H PRIME/CUB RT2-H)
  IF(CABS(G) .LT. 1.0E-50) GO TO 21
  SIGN = -ROOT P*ROOT Q*ROOT R*2.0/G
  IF(SIGN .LT. 0.0) ROOT P = -ROOT P

  Q(1) = +ROOT P+ROOT Q+ROOT R-B3
  Q(2) = +ROOT P-ROOT Q-ROOT R-B3
  Q(3) = -ROOT P+ROOT Q-ROOT R-B3
SUBROUTINE TIPRX
C FOR MS
C COMPUTATION OF M MATRIX ELEMENTS AS DEFINED BY BUDDEN IN PROC.
C RAY. SOC. (LONDON), A227, PP. 516-537 (1955). THOSE
C COMBINATIONS OF M MATRIX ELEMENTS USED IN THE
C T MATRIX WHICH DO NOT INCLUDE USE OF THETA ARE THE
C FINAL OUTPUT OF THIS ROUTINE. THE DUMMY INDEXING
C ON L IS USED IN VIEW OF POSSIBLE STORAGE AS FUNCTIONS
C OF HEIGHT) AND RE-USE OF THESE COMBINATIONS FOR
C DIFFERENT VALUES OF THETA.
C
COMMON/FFQ COMMON/FREQ
COMMON/FLD COMMON/CI2M,CM2I,CMP,FLMAG FLD
COMMON/H COMMON H
COMMON/ICNS COMMON/ICNS,MRATIO(3)
COMMON/WR COMMON WR
COMMON/EA COMMON/EA,COLL/HT,EN(1),CNU(1)
COMMON/M COMMON/M,MTX/M11,M21,M31,M12,M22,M32,M13,M23,M33
COMMON/I COMMON/I,STORE X/X(1001),X(1001),X(1001),X(1001),
3 X(1001),X(1001),X(1001),X(1001),
COMPLEX
$ U,USD,JOI, RD,
$ USQD,TA,TR,
$ U TEMP,CPLX 1,UPTEMP,
$ M13D,M31D,M23D,M32D,
$ M232D,M132D,M231D,M321D,
$ M11,M21,M31,M12,M22,M32,M13,M23,M33,
$ X11,X44,X44,X44,X44,X44,X44,X44
REAL MAG FLD,LY,MY,NO,IU,PRT,MRATIO,
$ LSQSQ,MSQSQ,NSQSQ,LSQSQ,LSQSQ,MSQSQ,LSQSQ,MSQSQ
DATA CPLX 1/(1.0,0.1,0.1)/
DATA PI/3.141592653/
DATA STR/0.01745329252/
DATA COEFF X/3.182357693/,COEFF Y/1.759796911/
DATA VEL LT/2.997928905/
DATA RE/6.39.427/
DIMENSION U PARTS(2),USQ PRT(2),U PRTS(2),IUD PRT(2)
EQUIVALENCE (U,U PARTS(1) ),(USQ,USQ PRT(1) ),(D,D PARTS(1) ),
$ (IUD,IUD PRT(1) )

CALL EN
C
IF(EN. NE. 0) GO TO 20
X = (UEF EN*EN(1)
Z = CNU(1)*QV OPMA
U PARTS(1) = 1.0
U PARTS(2) = -Z
USQ PRT(1) = 1.0-Z*Z
USQ PRT(2) = -Z-Z
D = -Y*U*(USQ-YSQ))
IUD PRT(1) = Z*D PARTS(1)-D PARTS(2)
IUD PRT(2) = D PARTS(1)+Z*D PARTS(2)
USQR = USQR*D
C1. TL: 30

22 C =......
USQR = 0.0
IUD = 0.0
DG 22 K = 1,3
Z = CNR(K1)/MEGA
U = 1.0 - CPLX 1+i2
USQR = U**2
UTEMP = -X/(U*(U-SQ-YSQ*MRATI(O(K))*2))
L = U + UTEMP/MRATIO(K)**2
UTEMP = UTEMP/2**2
IF(K EQ. 2) UDTMP = -UDTEMP
IUD = IUD + CPLX 1*UDTMP
22 USQR = LSQR + USQR*UTEMP

C3 C/VTRM = 2.0*(HT-H)/RE

M11 = JSCD - LSQYSQ*D + C/VTRM
M22 = JSCD - MSCYSQ*D + C/VTRM
M23 = USQR - NSCYSQ*D + C/VTRM
TA = NY*IUD
TB = LMYSC*D
M1 = +TA-TB
M12 = -TA-TB
TA = NY*IUD
TB = LNYSQ*D
M13 = +TA-TB
M13 = -TA-TB
TA = LMY*IUD
TB = LNYSQ*D
M32 = +TA-TB
M23 = -TA-TB

C4 = 1.0/(1.0*M33)
M13D = M13*D
M31D = M31*D
M23D = M23*D
M32D = M32*D
M232D = M32*M23D
M131D = M31*M13D
M231D = M31*M23D
M321D = M32*M13D

C

L = 1
X11(L) = -M11D
X14(L) = -M11D
X12(L) = M22D
X34(L) = M22D
X14(L) = D
X31(L) = M2331D-M21
X42(L) = M3213D-M12
X32(L) = M22-M2332D
X41(L) = 1.0+M11-M1331D
RETURN

C
C COMPUTATION OF QUANTITIES WHICH ARE NOT FUNCTIONS OF HEIGHT
C OR OF THE ETA.
C
ENTRY INIT T
OMEGA = 2.0*PI*FREQ*1J*0.0
WAVE NR = OMEGA/VEL LT
COEFF EN = COEFF X*1.0F06/OMEGA**2
UV OMEGA = 1.0/OMEGA
SIN CIP = SIN(COIP*DTR)
DIR CS L = SIN DIP*COS(AZIM*DTR)
DIR CS M = SIN DIP*SIN(AZIM*DTR)
DIR CS N = -COS(C) DIP*DTR
Y = -COEFF Y*MAG FLC/OMEGA
YSQ = Y**2
LY = DIR CS L*Y
MY = DIP CS M*Y
NY = DIR CS N*Y
LSQYSQ = DIR CS L**2*YSQ
MSQYSQ = DIR CS M**2*YSQ
NSQYSQ = DIR CS N**2*YSQ
LMYSQ = DIR CS L*DIR CS M*YSQ
LMYSQ = DIR CS L*DIR CS N*YSQ
MNYSQ = DIR CS M*DIR CS N*YSQ
RETURN

END
SUBROUTINE S_MTRX
C FOR MS
C COMPUTATION OF COEFFICIENTS USED IN THE DIFFERENTIAL EQUATIONS
C FOR (3+1)/C. THESE ARE STORED IN COMMON AREA 'S_MTX'
C AND ARE ANALOGOUS TO S MATRIX ELEMENTS GIVEN BY
C BUDDEN (1955). DERIVATIVES OF THESE COEFFICIENTS
C WRT (C=COS(THETA)) ARE ALSO COMPUTED IF I_JERIV
C IS SET NON-ZERO.
C
COMMON/THETA C/THETA
COMMON/C CCM/C
COMMON/S CM/S
COMMON/STORE X/X11(001),X44(001),X12(001),X34(001),X14(001),
$X31(001),X42(001),X32(001),X41(001)
COMMON/S MTRX/A11,A22,B11,B12,B22,C11,C21,C22,D11,D12,D22,
$C110DC,DI22DC,DC22DC,DD11DC,DD22DC,UD12DC
COMMON/IERIV C/I_JERIV
COMPLEX
S
$THETA,
$C,C,S,C5W,C5,
$CSDC,DCSDC,
$X11,X44,X12,X34,X14,X31,X42,X32,X41,
$TEMP,TGWC,
$A11,A22,B11,B12,B22,C11,C21,C22,D11,D12,D22,
$CD11DC,DI22DC,DC22DC,DD11DC,DD22DC,UD12DC
C
DATA RTD/57.29570/
C
CALL T_MTRX
C
L = 1
TEMP = X11(L)+X41(L)
A11 = TEMP+TEMP
A22 = 4.*C
TEMP = S*X44(L)-C*X41(L)
B11 = TEMP+TEMP
B12 = -X42(L)-X42(L)
TGWC = C+C
B22 = -TGWC
TEMP = -S*X11(L)-C*X41(L)
C11 = TEMP+TEMP
C21 = X31(L)+X31(L)
C22 = -TGWC
D11 = CS*(X11(L)-X44(L)-(1.0-X14(L)))+CSW*(X41(L)-X14(L))
D12 = -C*X31(L)+S*X34(L)
D21 = S*X12(L)+C*X42(L)
D22 = -X32(L)
C
IF (I_JERIV .EQ. 0) RETURN
TEMP = CSDC*X44(L)-X41(L)
D11DC = TEMP+TEMP
D22DC = -2.*C
TEMP = -CSDC*X11(L)-X41(L)
C
OC11CC = TEMP+TEMP
DC22DC = -2.0
DD11CC = DSDC*(X11(L)-X44(L))
  +2.0*(X41(L)-X44(L))
DD21DC = -X31(L)+DSDC*X34(L)
DD12CC = DSDC*X12(L)+X42(L)
RETURN

C

C INITIALIZATION FOR GIVEN VALUE OF THETA.
ENTRY INIT S
C = CCOS(THETA/RTO)
CSQ = C**2
S = CSIN(THETA/RTO)
CS = C*S
C
IF (IDERV .EQ. 0) RETURN
DSDC = -C/S
DCSDDC = S-CSQ/S
RETURN
C
END
SUBROUTINE DIFF EQ
C FOR MS
C DIFFERENTIAL EQUATIONS FOR (R+1.0)/C WHERE R IS THE REFLECTION
C MATRIX AS DEFINED BY BUDGEN. THE EQUATIONS ARE
C DERIVED FROM THE DIFFERENTIAL EQUATIONS FOR R
C GIVEN BY BUDGEN IN OC. RCY. SOC. (LONDON), A27;
C PP. 515-537 (1955). STORAGE INTO THE ARRAY 'R MTRX'
C IS MADE SO THAT THE EQUATIONS MAY BE USED WITH
C EITHER THE MAIN SET OF INTEGRATION VARIABLES, OR
C WITH THE COMPARISON SET, X, DIFFERENTIAL EQUATIONS
C FOR THE DERIVATIVES OF (R+1.0)/C WITH C=COS(THETA) ARE
C ALSO USED IF DERIV IS SET NON-ZERO. NOTE THAT THE
C VARIABLE CALLED 'R' IN THIS ROUTINE IS ACTUALLY
C (R+1.0)/C. MULTIPLICATION BY -1 IS EFFECTED BY
C APPROPRIATE TRANSFER OF VALUES FROM 'DERIV' TO
C COMMON VARIABLES.
C
COMMON/ INTG /X(16),DR0H(16)
COMMON/ X INTH/X(8),NINH(8)
COMMON /S MTRX/ALL,A11,A22,ALL,A12,B22,C11,C22,D11,D21,D12,D22,
$ EB1DC,EB2DC,EC1DC,EC2DC,DD1DC,DD2DC,DD1DC
COMMON /EXTRA/ R11,R21,R12,R22,DR11DC,DR21DC,DR12DC,DR22DC,
$ ER11CH,DR11CH,DR12CH,DR22CH
COMMON /ITERV /C/DERIV
COMMON / CCW/N
COMMON /CVFLO/ICVFLO
COMPLEX
$ P11,P12,R12,R22,DR11DC,DR21DC,DR12DC,DR22DC,
$ DR11DC,DR21DC,DR12DC,DR22DC,
$ ER11CH,DR11CH,DR12CH,DR22CH,
$ P11,A22,B11,B22,C11,C22,D11,D21,D22,
$ EB1DC,EB2DC,EC1DC,EC2DC,DD1DC,DD2DC,DD1DC
COMMON /DIMEN S/ R MTRX(10),DERIV(16)
EQUIVALENCE (R11,R MTRX(1) ),(DR11DC,DERIV(1) )
C ENTRY P CE RIV
CALL XHFF (P,R MTRX,N)
IFLAG = 0
20 DO 21 I=1,N
IF (ABS(R MTRX(I)) .GT. 1.0E4) GO TO 20
21 CONTINUE
C
D11R11 = D11*R11
D12R21 = D12*R21
D21R12 = D21*R12
D22R22 = D22*R22
R11R22 = R11*R22
P12R21 = R12*P21

C

DR11DH = A11+(B11+C11+D11R11+D12R21+D21R12)*R11
$ +R12*P11+C21+R12*D22*R12R21

DR12DH = (B22+C11+D11R11+D12R21+D22R22)*R21
$ +C21*P22+D21*R11R22

DR12DH = (B11+C22+D11R11+D21R12+D22R22)*R12
$ +R12*P22+D12*R11R22

DR22DH = A22+(B22+C22+D12R12+D21R12+D22R22)*R22
$ +D11*R12R21

IF (IXFLAG .NE. 0) GO TO 40

C

IF (IDERIV .EQ. 0) GO TO 22

G11R11 = LD11DC*R11
G12R21 = DU12DC*R21
G21R12 = DU21DC*R12

C

D11R11 = D21*R11
D12R11 = D12*R11
D11R21 = D11*R21
D22R21 = D22*R21
D11R12 = D11*R12
D22R12 = D22*R12
D11R22 = D11*R22
D22R22 = D12*R22

C

DR11CH = (C11DC+DC11DC+G11R11+G12R21+G21R12)*R11
$ +D11R11+C21+D21R12)*R11DC
$ +(D12R11+C21+D12R21)*DR11DC

DR12CH = (C12DC+DC12DC+G11R11+G12R21+G21R12)*R21+R22DC*R12R22
$ +D12R12+D12R22)*DR12DC
$ +(D12R12+D12R22)*DR22DC

DR12DH = (C11DC+DC12DC+G11R11+G12R21+G21R12)*R21+R22DC*R12R22
$ +D12R12+D12R22)*DR12DC
$ +(D12R12+D12R22)*DR22DC

DR22DH = (C12DC+DC22DC+G12R21+G21R12)*R22+R22DC*R12R22
$ +D12R12+D12R22)*DR22DC
$ +(D12R12+D12R22)*DR12DC

C

22  NO 24  I=1,N,2

C

ENTRY X DERIV
CALL XFEF (X,R,MTX*,8)
IXFLAG = 1
GO TO 20

C
SUBROUTINE ROARS

C FOR MS
C THIS SUBROUTINE COMPUTES VALUES OF VARIABLES WHICH MAY BE USED TO
C FORM THE ELEMENTS OF THE RBAR MATRIX, WHERE RBAR
C REPRESENTS REFLECTION OF AN ELM WAVE FROM THE EARTH'S
C SURFACE. NAMELY, RBAR11=DEN11/(C*NUM11-DEN11),
C OR (1.0/REBAR11+1.0)/C=NUM11/DEN11,
C AND RBAR22=DEN22/(C*NUM22-DEN22),
C OR (1.0/REBAR22+1.0)/C=NUM22/DEN22. DERIVATIVES
C WRt C=COS(THETA) ARE ALSO COMPUTED IF IDERV
C IS SET NON-ZERO. NOTE THAT THE
C EQUATIONS ARE FORMULATED IN SUCH A WAY THAT A SMOOTH
C TRANSITION IS MADE FROM THE 'CURVED EARTH' FORM TO
C THE 'FLAT EARTH' FORM. SEE ALSO NOTES IN SUBROUTINE
C MUHNLKl REGARDING DEFINITIONS OF HANKEL FUNCTION
C PARAMETERS. THE VALUE OF EARTH'S RADIUS IS SUCH AS
C TO MAKE 2.0/RE=3.14E-4. COMPUTATION AT THETA=90
C IS NOT EXCLUDED.

COMMON/RTALT/ TALT, RALT
COMMON/THETA C/THETA
COMMON/C CM/C
COMMON/NB COM/NUM11, NUM22, DEN11, DEN22, DNM1DC, DNM2DC, DDN1DC, DDN2DC
COMMON/FLF2 C/FL, F2, HG, F1LT, F1ZR, F2ZT, F2ZR, GF1ZT
COMMON/FRQ COM/FREQ
COMMON/H CLM/H
COMMON/C CCM/D
COMMON/GND COM/EPSPREL, SIGMA
COMMON/ICERV C/IDERV

COMPLEX
$ EZ, F2,
$ .EZ, C,
$ NUM11, NUM22, DEN11, DEN22,
$ DNM1DC, DNM2DC, DDN1DC, DDN2DC,
$ FL, F2, HG, HG, FCTR, FLZT, F1ZR, F2ZT, F2ZR, UF1LT, GF1ZT,
$ CSQ, SQS,
$ NGSO, I,
$ SQR,
$ L, UZDC,
$ H1, HIP, H1P,
$ H2, H2P, H2PP,
$ E, EO, EXPON,
$ KK, KK,
$ A1, A2, A3, A4, B1, B2, B3, B4,
$ DAIUC, DAZ2DC, DA3UC, DA4DC, DB1DC, DB2DC, DB3DC, DB4DC
REAL KVRANT, KVRAIT, NOSQ, NDSQ
DATA RTC/57.29578/
DATA I/(0,1,0,1)/
DATA EPSLNO/8.85434E-12/
DATA VEL LT/2.997928E5/
DATA PI3.141593/
DATA RE/6364.427/
C
C C = CCF*STHETA/KTD)
C SQ = C**2
C SQ = 1.0
C
C HEIGHT GAINS FOR Z=0
C SQ = C*CR(NGSC-SQ)
C NUM11 = 1.0
C DEN11 = (C-SQR/NOSQ)/2.0
C NUM22 = 1.0/SQR
C DEN22 = (C/SQR-1.0)/2.0
C IF (19F4.0 EQ. 0) GO TO 19
C DNMLC = 0.0
C ENMLDC = (1.0-C/(SQP*NOSQ))/2.0
C ENMLDC = -(C/SQR)**3
C DZDC = -(C**2/SQR**3+1.0/SQR)/2.0
C
C 19 = KFRAAT*1(CSQ*EMT)
C CALL MDHKL (Z+1+H2+1+H1+HP+HP,FJ)
C A1 = C*H1+(KK*H1+K2*H1)/NOSQ
C P2 = C*H2+(KK*H2+K2*H2)/NOSQ
C A3 = C*H1+KK*H2P
C B4 = C*K2+KK*H2P
C A1 = NUM11*81-2.0*DEN11*H1
C A2 = NUM11*82-2.0*DEN11*H2
C A3 = NUM22*83-2.0*DEN22*H1
C A4 = NUM22*84-2.0*DEN22*H2
C IF (19F4.0 EQ. 0) GO TO 20
C DZDC = KFRAAT*2.0*C
C H1PP = Z*H1
C H2PP = Z*H2
C DB1DC = F1+(C*H1P+(KK*H1PP+K2*H1PP)/NOSQ)*DZDC
C DB2DC = F2+(C*H2P+(KK*H2PP+K2*H2PP)/NOSQ)*DZDC
C DB3DC = F1+(C*H1P+KK*H1PP)*DZDC
C DB4DC = F2+(C*H2P+KK*H2PP)*DZDC
C DA1DC = NUM11*DB1DC+DNM11*DEN11*H1-2.0*(DEN11*H1P*DZDC+DNM11*H1)
C DA2DC = NUM11*DB2DC+DNM11*DEN11*H2-2.0*(DEN11*H1P*DZDC+DNM11*H1)
C DA3DC = NUM22*DB3DC+DNM22*DEN22*H1-2.0*(DEN22*H1P*DZDC+DNM11*H1)
C DA4DC = NUM22*DB4DC+DNM22*DEN22*H2-2.0*(DEN22*H1P*DZDC+DNM22*H2)
C HG FCTR = (A2-H1-H2)/(1-KK)
C
C C HEIGHT GAINS FOR ELEVATED TRANSMITTER AND ELFINATED RECEIVER
C Z = KFRAA*T(CSQ+EMT)
C CALL MDHKL (Z,H1,H2,H1P,H2P,EZ)
C EXPON = CEXP(EZ-E0)
C H1 = H1/EXPON
C H2 = H2/EXPON
C H1P = H1P/EXPON
C H2P = H2P/EXPON
C F1ZT = (A2-H1-H2)/(1-KK)*EXP((TALT-D)/RE)
C F2ZT = (A4-H1-H2)/(1-SQR/KK)
C C PART OF THE EQUATION OF THE DERIVATIVE OF HEIGHT GAIN FOR
C FPERAL AT HEIGHT Z
PF1Z1 = \( (A2*H1P-A1*H2P)/(-KK)*\exp((TALT-D)/RE) \)

C
The DERIVATIVE OF THE HEIGHT GAIN FOR FPERL AT HEIGHT Z
C
EXCLUSIVE OF THE D11 TERM
C
F1ZT = -1*(AV*KLT)*F1ZT + AV*F1LT/2.*Z)
C
C
Z = KVRAT*(CSQ+FMZR)
CALL MDHNLH (Z,H1,H2,H1P,H2P,EZ)
EXPON = CFXPIE-EO)
H1 = H1/EXPON
H2 = H2+EXPON
F1ZP = (A2+H1-A1*H2)/(-KK)*\exp((ZALT-D)/RE)
F2ZR = (A4+H1-A3*H2)*(-SQR/KK)
C
C
HEIGHT GAINS AT Z=D
C
CALL MDHNLH (Z,H1,H2,H1P,H2P,EZ)
EXPON = CFXPIE-EO)
H1 = H1/EXPON
H2 = H2+EXPON
H1P = H1F/EXPON
H2P = H2F+EXPON
C
B1 = C*H1+(KK*H1P+KK*H1P)/NDSQ
B2 = C*H2+(KK*H2P+KK*H2P)/NDSQ
B3 = C*H1+KK*H1P
B4 = C*H2+KK*H2P
DNUM1 = A2*B1-A1*B2
DNUM2 = A4+B3-A3*B4
NUM11 = 2.0*(A2+H1-A1*H2)
NUM22 = 2.0*(A4+H1-A3*H2)
IF (IDEFINED *EQ. 0) RETURN
H1PP = -Z*H1
H2PP = -Z*H2
DB1DC = F1+(C*H1P+(KK*H1P+KK*H1P)/NDSQ)*DZDC
DB2DC = F2+(C*H2P+(KK*H2P+KK*H2P)/NDSQ)*DZDC
DB3DC = F1+(C*H1P+KK*H1P)*DZDC
DB4DC = F2+(C*H2P+KK*H2P)*DZDC
DNUM1DC = A2*B1DC+DA2DC*H1-A1*DB2DC-DA1DC*B2
DNUM2DC = A4+B3DC+BA4DC*B3-A3*DB4DC-BA3DC*B4
DNM1DC = 2.0*(A2+H1P*DZDC+DA2DC*H1-A1*H2P*DZDC-DA1DC*H2)
DNM2DC = 2.0*(A4+H1P*DZDC+DA4DC*H1-A3*H2P*DZDC-DA3DC*H2)
C
C
FOLLOWING FCUP STATEMENTS NEEDED ONLY FOR COMPUTING EXCITATION
C
FACTORS.
C
HEIGHT GAIN FOR FPERL AT Z=D
F1 = (A2+H1-A1*H2)/(-KK)
C
HEIGHT GAIN FOR FPERP AT Z=D
F2 = (A4+H1-A3*H2)*(-SQR/KK)
C
HEIGHT GAIN FOR FPERL AT Z=0 DIVIDED BY
C
HEIGHT GAIN FOR FPERL AT Z=D
HG = \exp(-D/RE)*HG FCTR/F1
RETURN
C
COMPUTATION WHICH IS INDEPENDENT OF THETA.

ENTRY INIT

OMEQA = 2.0*PI*EXP10*1.0000

WAVE NF = OMEGA/VEL LT
NGSQ = EPSNEL-1*SIGMA/OMEGA/EPSLNO

ALK = 2.0/(WAVE NR*KE)
AVFKOT = EXP(ALG((ARG)/3.0))
KVKALT = 1.0/AVFKOT
KVRAT = 1.0/AVFKUT**2
KK = I*AVKLOT
KX = 1/(WAVE NR*RE)

EMZK = 2.0*(KALT-H)/RE
EMZT = 2.0*(TALT-H)/RE

EMT = 2.0*(L-H)/RE
NSQ = 1.0+EMD
EMU = -2.0*H/RE
NOSQ = 1.0-2.0*H/RE
RETURN

END
SUBROUTINE MDHNLK (Z,HL,H2,H1P,H2P,E)
C FOR MS
C THIS SUBROUTINE COMPUTES VALUES OF VARIABLES WHICH MAY BE COMBINED
C TO FORM MODIFIED HANKEL FUNCTIONS AS DESCRIBED
C IN TABLES OF THE MODIFIED HANKEL FUNCTIONS OF ORDER
C ONE THIRD AND OF THEIR DERIVATIVES', BY THE STAFF OF
C THE COMPUTATION LABORATORY, HARVARD UNIVERSITY
C PRESS, 1959. THE NOTATION IS SUCH THAT THE
C ACTUAL H1=H1*EXP(-E), THE ACTUAL H2=H2*EXP(E),
C ACTUAL DERIVATIVES WRT ARGUMENT, Z, ARE H1P*EXP(-E)
C AND H2P*EXP(E).
C
COMPLEX Z,HL,H2,H1P,H2P,E,
$ RE,
$ IM,IPARTS,
$ ZP,POWER,TERM,
$ SUMA,SUMB,SUMC,SUMD,
$ MZ3,
$ FG,FPS,GP,T,P,
$ IPI16T,I2THRD,IP1512,
$ SUM1,SUM2,SUM1P,SUM2P,
$ RTZ,OVKZ,OKM4Z,33H,33HUVZ,3RTZ,3MZ3H,3MIUZ3,
$ CSUM,ESUM,OSUMP,ESUMP,
$ EDIFF

DIMENSION A(LZ),B(LZ),CAP(LZ),PART(LZ)
EQUIVALENCE (PART(LZ),IPARTS)
DATA AO/C,930436/1693/
DATA 80/C,6782957254/
DATA CAP/ $ 1.04166666666667D-01,8.35503472222222D-02,1.2823657455327D-01,
$ 2.68949292454140D-01,6.8162726743750D-01,3.3214828186227D 00,
$ 1.49597629866250D 01,1.79230130115870D 01,4.7441538866600D 02,
$ 3.20749030100000D 03,2.40865496000000D 04,1.99923120000000D 05,
$ 1.79150200000000D 06,1.74843770000000D 07/ 
DATA 1/(C.1,1.0)/
DATA RCCT3/1.73205083075668/
DATA ALPHA/0.85356721838951/
DATA PI/3.14159265358979/
DATA I1ST/1/
DATA IFIRST/1/
C
Z=Mag=Calc(Z)
IF (ZMag .GT. 4.2) GO TO 50
C
MACLAURIN'S SERIES FORM FOR MAGNITUDE(Z) NOT GREATER THAN 4.2
IF (I1ST .EQ. 0) GO TO 12
A(1) = 1.0/6.0
B(1) = 2.0/24.0
DL 11 M=2,Z2
A(M) = A(M-1)*3.0*M-2.0)/(1.0*M)*(1.0*M-1.0)*(1.0*M-2.0)
B(M) = B(M-1)*(3.0*M-1.0)/(1.0*M+1.0)*(3.0*M)*(1.0*M-1.0)
11 CONTINUE
15  CONTINUE

16  F = .0*SUMA
    G = Z*.0*SUMA
    FP = 3.*C*AO*SUMC/Z
    GP = (3.*C*SUMC*SUMM)*R0
    T = 1.*(G+F-F)/RDOT3
    TP = 1.*(GP-FP-FP/R1J)T3
    H1 = G+T
    H1P = GP+TP
    H2 = G-T
    H2P = GP-TP
    E = 0.*0
    RETURN

C ASYMP&TIC SERIES FOR MAGNITUDE(Z) GREATER THAN 4.*2

50  IF (IFIRST .EQ. 0) GO TO 51

   IPI16T = 16.*0*PI*1/12.0
   I2THFD = 1*2.*0/3.0
   IPI512 = 5.*0*PI*1/12.0
   ALPHSW = ALPHA**2
   IFIRST = 0

51  RTZ = CSCR(TZ)
    OVR2 = 1.0/Z
    CVRMZ = -0.25*CVRZ
    L3H = Z*RTZ
    M30VZ = -1.5*0VZ
    H1T = I*RTZ
    M1Z3H = -I*Z3H
    M1OZ3 = -0.25*Z3H
    F = -12THRD*Z3H+IPI512
    R = CSWFT(ALPHSW/RTZ)

C

OSUM = C.*0
OSUMF = C.*0
F  
ESUMP = 0.0
ESUMP = 0.0
ZPOWER = 1.0
DIM 5, M=2,14,2
ZPOWER = ZPOWER*M10VZ3
TFRM = CAP(M-1)*ZPOWER
USUM = CSUMP+TERM
USUMP = CSUMP+(M-1)*TERM
TFRM = CAP(M)*ZPOWER
LSUM = ESUMP+TERM
ESUMP = ESUMP+TERM
PARTS = TFRM
IF (ABS(PART(1)) + ABS(PART(2)) < 1.0E-8) GO TO 53
52 CONTINUE
53 CONTINUE
USUM = M1Z3H*OSUM
USUMP = M1Z3H*M3HOVZ*OSUMP
FSUMP = M3HOVZ*ESUMP
SUM1 = 1.0 - USUM + EESUM
SUM2 = 1.0 + USUM + FSUMP
SUM1P = - OSUMP + ESUMP
SUM2P = CSUMP + ESUMP
C
H1 = SUM1
H2 = SUM2
H1P = (CVRM4Z+IKTL)*SUM1+SUM1P
H2P = (CVRM4Z-IKTL)*SUM2+SUM2P
PARTS = 2
IF (PART(1) .LT. 0.0) GO TO 70
C
IF (PART(2) .LT. 0.0) GO TO 63
EDIFF = CFXP(E+E-IP116T)
H1 = H1+EDIFF*H2
H1P = H1P+EDIFF*H2P
GO TO 70
63 EDIFF = CEXP(-E-E+IP116T)
H2 = H2+EDIFF*H1
H2P = H2P+EDIFF*H1P
C
70 H1 = F*H1
H2 = R*H2
H1P = R*H1P
H2P = P*H2P
RETURN
C
END
SUGGESTED FS INTG
C FOR MS
C THIS SUBROUTINE PERFORMS AN INTEGRATION OF THE DIFFERENTIAL
C EQUATIONS FOR THE IONOSPHERE REFLECTION MATRIX
C THROUGH A FREQUENCY REGION OVER A CURVED EARTH.
C THE INTEGRATION MAY BE PERFORMED IN EITHER A POSITIVE
C OR NEGATIVE HEIGHT DIRECTION, BUT IN THIS PROGRAM THE
C INTEGRATION IS ALWAYS UPWARD. THE INTEGRATION VARIABLES
C ARE THE ELEMENTS OF THE MATRIX (R+1)/C WHERE K IS THE
C REFLECTION MATRIX DESCRIBED BY PUDDEEN. THE SOLUTION
C IS BASED ON BUDDEN, RADIO WAVES IN THE IONOSPHERE,
C IN PARTICULAR THE MATERIAL ON PP. 118, 327-329,
C 336-358, AND 343-345. IF IDERV
C IS SET NON-ZERO, THE DERIVATIVES OF (R+1)/C ELEMENTS
C WRT C = COS(THETA) ARE ALSO INTEGRATION VARIABLES.
C COMPUTATION AT THETA=90 IS NOT EXCLUDED. ALSO,
C THE EQUATIONS ARE FORMULATED IN SUCH A WAY THAT A
C SMOOTH TRANSITION IS MADE FROM THE 'CURVED EARTH' FORM
C TO THE 'FLAT EARTH' FORM FOR APPROPRIATE VALUES OF
C THETA. THE INITIAL AND FINAL VALUES OF
C THE INTEGRATION VARIABLES ARE STORED IN THE COMMON
C AREA 'INTEGR'.
C
COMMON/XPRINT/XPRINT
COMMON/INTEGR/R11,R21,R12,R22,DR11DC,DR21DC,DR12DC,DR22DC
COMMON/THETA C/THETA
COMMON/C CCM/C
COMMON/H CCM/H
COMMON/NO/WAVE NO
COMMON/IDERV C/IDERV
COMPLEX THETA,C,
$ R11,R21,R12,R22,
$ DR11DC,DR21DC,DR12DC,DR22DC,
$ I,K,KX,C SQ,
$ EXPUN,
$ Z,H1,H2,H1P,H2P,E0,E,
$ H1P,H2P,DZDC,
$ A1,A2,A12,A22,
$ DA11DC,DA12DC,DA21DC,DA22DC,
$ F1,G1,A1,B1,F2,G2,A2,B2,
$ DF1DC,DA1DC,DA2DC,DB1DC,DF2DC,DG2DC,DA2DC,DD2DC,
$ EP1,EP2,EY1,EY2,
$ DEP1DC,DEP2DC,DEY1DC,DEY2DC,
$ DEN,VI,V2,
$ DDENDC,DV1DC,DV2DC
REAL KVRATT,NOSQ,NZSQ
INTEGER XPRINT
DATA I/(C,C,1.0)/
DATA RE/636.427/
DATA RT/57.2957/
C
IF (XPRINT .NE. 0) PRINT 100,R11,R21,R12,R22
### COMPUTATION OF HEIGHT-GAIN COEFFICIENTS F, G, A, B FOR TWO CONDITIONS

On the upgoing wave at height = 20, namely, $E_1 = 1$, $E_2 = 0$. Each of the coefficients $F, G, A, B$ is divided by $C = \cos(\Theta)$.

$$Z = KVATT \times (CSQ + EMO)$$

CALL MDHKKL ($Z, H_1, H_2, H_{1P}, H_{2P}, E_1$)

$A_{11} = H_1$

$A_{12} = H_2$

$A_{21} = C \times T + (KK \times H_1P + KK \times H_1) / NOSQ$

$A_{22} = C \times T + (KK \times H_2P + KK \times H_2) / NOSQ$

$DEN = A_{11} \times A_{22} - A_{21} \times A_{12}$

$F_1 = (R_{11} \times A_{22} - 2.0 \times A_{12}) / DEN$

$G_1 = (Z \times C \times A_{11} - R_{11} \times A_{12}) / DEN$

$F_2 = R_{12} \times A_{22} / DEN$

$G_2 = -R_{12} \times A_{21} / DEN$

### IF (IDERV = .EQ. 0) GO TO 25

$D1DC = KVATT \times 2.0 \times C$

$H_{1P} = -Z \times H_1$

$H_{2P} = -Z \times H_2$

$DA_{11DC} = H_{1P} \times D_{ZDC}$

$DA_{12DC} = H_{2P} \times D_{ZDC}$

$DA_{21DC} = H_{1} + (C \times H_{1P} + (KK \times H_{1P} + KK \times H_{1P}) / NOSQ) \times D_{ZDC}$

$DA_{22DC} = H_{2} + (C \times H_{2P} + (KK \times H_{2P} + KK \times H_{2P}) / NOSQ) \times J_{ZDC}$

$DEN1DC = A_{11} \times A_{22} - A_{21} \times A_{12}$

$DF1DC = (R_{11} \times A_{22} \times D_{R11DC} \times A_{22} - 2.0 \times A_{12}) / DEN$ + $D_{DENDC} / DEN$

$DG1DC = (2.0 \times A_{11DC} \times R_{11} \times A_{21DC} \times D_{R11DC} \times A_{21}) / DEN$ + $D_{G} \times D_{DENDC} / DEN$

$DF2DC = (R_{12} \times A_{22} \times D_{R12DC} \times A_{22}) / DEN$ + $D_{F2} \times D_{DENDC} / DEN$

$DG2DC = (-R_{12} \times A_{21DC} \times D_{R12DC} \times A_{21}) / DEN$ + $G_2 \times D_{DENDC} / DEN$

### 25 A11 = H1

$A_{12} = H_2$

$A_{21} = C \times T + KK \times H_{1P}$

$A_{22} = C \times T + KK \times H_{2P}$

$DEN = A_{11} \times A_{22} - A_{21} \times A_{12}$

$A_1 = R_{21} \times A_{22} / DEN$

$R_1 = -R_{21} \times A_{21} / DEN$

$A_2 = (R_{22} \times A_{22} - 2.0 \times A_{12}) / DEN$

$A_2 = (2.0 \times A_{11} - R_{22} \times A_{21}) / DEN$

### IF (IDERV = .EQ. 0) GO TO 30

$DA_{11DC} = H_{1P} \times D_{ZDC}$

$DA_{12DC} = H_{2P} \times D_{ZDC}$

$DA_{21DC} = H_{1} + (C \times H_{1P} + KK \times H_{1P}) \times D_{ZDC}$

$DA_{22DC} = H_{2} + (C \times H_{2P} + KK \times H_{2P}) \times D_{ZDC}$

---

202
COMPUTATION OF UPWARD FIELDS $E_{11}$ AND $E_Y$ AT HEIGHT $= 2Z$ FOR THE TWO CONDITIONS DESCRIBED ABOVE.

30 $Z = \text{VRA}^{2.5}(C \times Q + EMZ)$

CALL MLHAKL ($Z, H_1, H_2, H_1P, H_2P, E$)

$FXPON = \text{CEXP}(E - 10)$

$H_1 = H_1 / FXPON$

$H_2 = H_2 / FXPON$

$H_1P = H_1P / FXPON$

$H_2P = H_2P / FXPON$

$A_{11} = C * H_1 + (K_K * H_1P + K_2 * H_1) / NZSQ$

$A_{12} = C * H_2 + (K_K * H_2P + K_2 * H_2) / NZSQ$

$EP_1 = (F_1 * A_{11} + G_1 * A_{12}) / 2.0$

$EP_2 = (F_2 * A_{11} + G_2 * A_{12}) / 2.0$

IF (1DERIV .EQ. 0) GO TO 35

$HIPP = -Z * H_1$

$HP2P = -Z * H_2$

$DZ1DC = H_1 * (C * H_1P + K_K * H_1P) / NZSQ * DZDC$

$DZ2DC = H_2 * (C * H_2P + K_K * H_2P) / NZSQ * DZDC$

$DEP1DC = (F_1 * DA1DC + F1DC * A_{11} + G_1 * DA2DC + G1DC * A_{12}) / 2.0$

$DEP2DC = (F_2 * DA1DC + F1DC * A_{11} + G_2 * DA2DC + G1DC * A_{12}) / 2.0$

35 $A_{11} = C * H_1 + K_K * H_1P$

$A_{12} = C * H_2 + K_K * H_2P$

$EY1 = (A_1 * A_{11} + B_1 * A_{12}) / 2.0$

$EY2 = (A_2 * A_{11} + B_2 * A_{12}) / 2.0$

IF (1DERIV .EQ. 0) GO TO 40

$DZ1DC = H_1 * (C * H_1P + K_K * H_1P) * DZDC$

$DZ2DC = H_2 * (C * H_2P + K_K * H_2P) * DZDC$

$DEY1DC = (A_1 * DA1DC + DA1DC * A_{11} + B_1 * DA2DC + B1DC * A_{12}) / 2.0$

$DEY2DC = (A_2 * DA1DC + DA1DC * A_{11} + B_2 * DA2DC + B1DC * A_{12}) / 2.0$

COMPUTATION OF REFLECTION COEFFICIENTS AT HEIGHT $= 2Z$. 

40 $V_1 = F_1 * H_1 + G_1 * H_2$

$V_2 = F_2 * H_1 + G_2 * H_2$

DEN = $EP_1 * EY_2 - EP_2 * EY_1$

$R_{11} = (V_1 * EY_2 - V_2 * EY_1) / DEN$

$H_1 = (V_2 * EP_1 - V_1 * EP_2) / DEN$

IF (1DERIV .EQ. 0) GO TO 45

$DV1DC = (F_1 * H_1P + U_1 * H_2P) * DZDC + DF1DC * H_1 + DG1DC * H_2$

$DV2DC = (F_2 * H_1P + U_2 * H_2P) * DZDC + DF2DC * H_1 + DG2DC * H_2$

$DZ1DC = EPI * DEY2DC + DEP1DC * EY_2 - EP_2 * DEY1DC - DEP2DC * EY_1$

$DZ1DC = (V_1 * DEY2DC - DV1DC * EY_2 - V_2 * DEY1DC - UV2DC + EY_1) / DEN$

$DEN = -R_{11} * DZ1DC / DEN$
C

\[ V_1 = A_1 + B_1 \times H_2 \]
\[ V_2 = A_2 + B_2 \times H_2 \]
\[ R_{z1} = (V_1 \times E\_C - V_2 \times E\_F) / T \]
\[ R_{z2} = (V_2 \times E\_P - V_1 \times E\_P) / T \]

C

\[ \text{IF (R2*E) > 0.1) \text{ CC IC 50} \]
\[ \text{IF (R2*E) < 0.1) \text{ CC IC 50} \]
\[ \text{IF (R2*E) = 0.1) \text{ CC IC 50} \]

C

\[ \text{COMPUTATION WHICH IS INDEPENDENT OF THETA, THE INTEGRATION WILL} \]
\[ \text{PROCEED FROM HEIGHT 20 TO HEIGHT 22 WHEN FSINTG IS} \]
\[ \text{CALLED.} \]

C

\[ F\text{NTRY INIT FS(20,22)} \]
\[ \text{APG} = 2.0 / \text{WAVE NR*RE) } \]
\[ \text{AVRKUT = EXP ALOG(ARG) / 3.0} \]
\[ \text{AVRATT = 1.0 / AVRKUT**2} \]
\[ \text{KY = 1 / AVRKUT} \]
\[ \text{KX = 1 / (WAVE NK*RE)} \]

C

\[ \text{EMO} = 2.0 * (ZO-H) / \text{RE} \]
\[ \text{NUSO} = 1.0 + \text{EMO} \]
\[ \text{EMZ} = 2.0 * (ZL-H) / \text{RE} \]
\[ \text{Z2SG} = 1.0 + \text{EMZ} \]

C

\[ \text{FEND} \]
SUBROUTINE FCT VAL (THETAZ, F)

FOR MS.

ROUTINE: FOR COMPUTING F FUNCTION VALUES AT MESH POINTS, CALLED ON
PRINCIPALLY BY SUBROUTINES FZEPOS AND NO MESH. THIS
IS THE MODIFIED F FUNCTION WHICH HAS NO POLES AND WHICH
HAS NO ZEROS AT THETA=90.

COMMON/IDERV C/IDERIV
COMMON/IXACT C/IEXACT
COMMON/THETA C/THETA
COMMON/INTGR/R11,R21,R12,R22
COMMON/RB C/RB NUM11,NUM22,DEN11,DEN22
COMPLEX THETAZ,F,
$   $  THETA, 
$   $  R11,R21,R12,R22,
$   $  NUM11,NUM22,DEN11,DEN22

THETA = THETAZ
IDERIV=0
IF (IEXACT .EQ. 0) GO TO 11
CALL INTFG
CALL FS INTG
GO TO 12
11 CALL LACRNG
12 CALL PBA,
F = (NUM11-R11*DEN11)*(NUM22-R22*DEN22)
$   $  -R12*R21*DEN11*DEN22
RETURN

END
SUBROUTINE F DFDT (THETA,F,DFDT)
C FOR FS
C ROUTINE FOR COMPUTING F FUNCTION VALUES AND THEIR DERIVATIVES
C AT THETA AT ARBITRARY VALUES OF THETA. CALLED ON
C PRINCIPALLY BY SUBROUTINES NO MESH AND FINAL. THIS
C IS THE MODIFIED F FUNCTION WHICH HAS NO POLES AND WHICH
C HAS NO ZEROS AT THETA=90.
C
COMMON/ICERV C/IDERIV
COMMON/IAACT C/IEXACT
COMMON/KXACT C/ KEXACT
COMMON/THETA C/THETA
COMMON/INTEGR/R11,R12,R,R22
COMMON/RB COM/NUM11,NUM22,DEN11,DEN22,
COMMON/DM DC DN MDC DN1DC, DN2DC
COMPLEX
$ THETA,F,DFDT,
$ THETA,
$ DFDC,S,
$ R11,R21,R12,R22,
$ LR11DC,DR21DC,DR12DC,DR22DC,
$ NUM11,NUM22,DEN11,DEN22,
$ DN1DC, DN2DC, DN1DC, DN2DC,
$ FACTRI,FACTR2,R12,R21,UN1DN2,
$ DFDR11,DFDR21,DFDR12,DFDR22,
$ DFUNM1,DFUNM2,DFUN1,DFUN2
DATA RTC/57.295 /
C
C THETA = THETA
IDERIV=1
IF(IEXACT .EQ. 0 .AND. IEXACT .EQ. 0) GO TO 11
CALL INTEG
CALL FS INTG
GO TO 12
11 CALL LAG DER
12 CALL RBARS
C
FACTR1 = NUM11-R11*DEN11
FACTR2 = NUM22-R22*DEN22
R12R21 = R12*R21
DN1DN2 = DEN11*DEN22
F = FACTR1*FACTR2-R12R21*UN1DN2
C
DFUNM1 = FACTR2
DFUNM2 = FACTR1
DFUN1 = -R11*FACTR2-R12R21*DEN22
DFUN2 = -R22*FACTR1-R12R21*DEN11
DFOR11 = -DEN11*FACTR2
DFOR21 = -R12*DN1DN2
DFOR12 = -R21*DN1DN2
DFOR22 = -DEN22*FACTR1

206
DFQC = (DFOR1L*OR1UC + DFOR2L*OR2UC + DFDR1L*OR1UC + DFDR2L*OR2UC + DFON1L*ON1UC + DFON2L*ON2UC)

S = L*Sin(THETA/RTD)

RETURN

END
SUBROUTINE LAGRANG
C FOR MS
C ROUTINE FOR PERFORMING LAGRANGE INTERPOLATION OF REFLECTION
C COEFFICIENTS IN THE COS(THETA) PLANE.
C
COMMON/THETA C/THETA
COMMON/INTEGR/R(4),DRC(4)
COMPLEX THETAG(1),R GIVEN(4,1),
$ 1*THETA,R,DRC,
$ C,PROD,SUM(4),COEF(4,100),C=100),DEN(JG)
DATA RTE/57,29578/
C
C LAGRANGE INTERPOLATION.
C = C*COS(THETA/RTD)
21 DO 22 I=1,4
22 SUM(I) = 0,0
DC 24 JG=1,NG
PROD = 1.0
DC 23 J=1,NG
IF (J .EQ. JG) GO TO 23
PROD = PROD*(C-CG(J))
23 CONTINUE
DO 24 I=1,4
24 SUM(I) = SUM(I)+COEF(I,JG)*PROD
DC 25 I=1,4
25 R(I) = SUM(I)
RETURN
C
C INITIALIZATION OF LAGRANGE INTERPOLATION USING INFORMATION ON
C LOCATION OF GIVEN VALUES IN COS(THETA) PLANE.
ENTRY INIT LS (THETAG,NG)
DO 31 JG=1,NG
31 CG(JG) = CCOS(THETAG(JG)/RTD)
C
CC 34 JG=1,NG
PROD = 1.0
DC 32 J=1,NG
IF (J .EQ. JG) GO TO 33
PROD = PROD*(CG(JG)-CG(J))
33 CONTINUE
34 DEN(JG) = PROD
RETURN
C
C FURTHER INITIALIZATION OF LAGRANGE INTERPOLATION USING INFORMATION
C ON GIVEN VALUES.
ENTRY SET LAG (NG,R GIVEN)
DC 41 JC=1,NG
DC 41 I=1,4
41 COEF(I,JG) = R GIVEN(I,JG)/DEN(JG)
RETURN
ENTRY LAGRANGE

COMPUTATION OF DERIVATIVES OF INTERPOLATED VALUES WRT COS(THETA).

NO EXACT VALUES OF DKOC ARE USED.

THE LAGRANGE INTERPOLATION FORMULA ITSELF IS DIFFERENTIATED WRT COS(THETA).

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C

C
SUBROUTINE FZEROS (TLEFT, TRIGHT, TBOT, TTOP, TMESH, TUL, MPRINT, 
  ZEROS, NR Z)
FOR ROOT-FINDER
FZEROS IS A ROUTINE FOR FINDING THE ZEROS OF A COMPLEX FUNCTION, F,
which lie within a specified rectangular region of the complex (theta) plane, provided the function has no
poles in the vicinity of the rectangle.

EXPLANATION OF PARAMETERS--
TLEFT - VALUE OF REAL PART OF THETA AT LEFT EDGE OF rectange.
TRIGHT - VALUE OF REAL PART OF THETA AT RT EDGE OF RECTANGLE.
TBOT - VALUE OF IMAG PART OF THETA AT BOTTOM EDGE OF RECTANGLE;
TTOP - VALUE OF IMAG PART OF THETA AT TOP EDGE OF RECTANGLE.
TMESH - SET EQUAL TO ABOUT HALF THE AVERAGE SPACING BETWEEN
ZEROS WITHIN THE RECTANGLE. A SMALLER VALUE MAY BE USED
AS A SAFETY MEASURE, BUT TOO SMALL A VALUE WILL RESULT
IN EXCESSIVE RUN TIME.
TOL - TOLERANCE TO WHICH ZEROS ARE TO BE FOUND. IF TWO
ZEROS ARE CLOSER THAN 'TOL', THE ROOT-FINDER WILL STOP
WITH AN ERROR MESSAGE.
MPRINT - NORMALLY SET TO ZERO. A NON-ZERO VALUE LEADS TO
PRINT-OUT FOR DEBUGGING.
ZEROS - OUTPUT LIST OF (COMPLEX) VALUES OF THETA AT WHICH
ZEROS ARE FOUND.
NR Z - THE NUMBER OF ZEROS FOUND.

SUBROUTINES TO BE PROVIDED--
FCI VAL (THETA, F) - TO RETURN THE VALUE OF THE FUNCTION, F,
AT THE POINT IN THE COMPLEX PLANE SPECIFIED BY 'THETA'.
F DFDT (THETA, F, DFDT) - SAME AS 'FCI VAL' EXCEPT THAT THE
DERIVATIVE, DFDT, OF THE FUNCTION WRIT THETA MUST ALSO
BE RETURNED.

COMMON/ZLS COM/NR ZT, NR ZLS(100)
COMM/N/TMC COM/TMESHC
DIMENSION ZEROS(2,1)
COMPLEX F, PFV, F, FO0, FL, FQ1, FQ, FPARS
DIMENSION PAKT(2), SOL(2), THETA(2)
DIMENSION K EDGE1(50), K EDGE2(50), K EDGE3(50), K EDGE4(50)
EQUIVALENCE (PAKT1), PARIS)

IF (MPRINT .NE. 0) PRINT 906
906 FORMAT ('1')
TMESHC = TMESH
SIDE5 OF RECTANGLE IN TMESH UNITS
JLT = TLEFT/TMESH
IF (TLEFT .GE. 0.0) JLT = JLT-1
IF (TLEFT .LT. 0.0) JLT = JLT-2
JRT = TRIGHT/TMESH
IF (TRIGHT .GT. 0.0) JRT = JRT+2
IF (TRIGHT .LE. 0.0) JRT = JRT+1

210
C
JBOT = TBOT/TMESH
IF (TBOT .GE. 0.0) JBOT = JBOT - 1
IF (TBOT .LT. 0.0) JBOT = JBOT + 1

C
JTOP = ITOP/TMESH
IF (ITOP .GT. 0.0) JTOP = JTOP + 1
IF (ITOP .LE. 0.0) JTOP = JTOP - 1

C
INITIALIZATION OF PARAMETERS.
KI = JTOP
KR = JLT
KEDGE = 1
CALL FIND F (KR, KI, F)
NR E1 = 0
NR E2 = C
NR E3 = C
NR E4 = C
NR LT = C
NR ZL = C
NR Z = 0
10 IF (NR ZL .EQ. 0) GO TO 20
NR ZLS(NR Z) = NR ZL
NR ZL = 0

C
20 PREVF = F
IF (NR ZL .GT. 1) PRINT 920, NR ZL
920 FPPMAT ('O', '13', '1X', 'MODES FOUND ON SAME PHASE LINE')
NR ZL = C
GO TO (21, 26, 31, 36), KEDGE

C
SEARCH ALONG LEFT EDGE OF RECTANGLE FOR SIGN CHANGES IN IMAG(F).
21 IF (K1 .EQ. JBOT) KEDGE = 2
IF (KEDGE .EQ. 2) GO TO 26
KI = KI - 1
CALL FIND F (KR, KI, F)
IF (((AIMAG(PREV F) .GT. 0.0) .AND. AIMAG(F) .LT. 0.0))
$ .OR. ((AIMAG(PREV F) .LT. 0.0) .AND. AIMAG(F) .GT. 0.0))
$ GO TO 20
IF (NR E1 .EQ. 0) GO TO 23
DO 22 K = 1, NR E1
IF (K1 .EQ. KEDGES(K)) GO TO 20
22 CONTINUE
23 FO1 = PREVF
FOO = F
LI = KI
LP = JLT
GO TO 43

C
SEARCH ALONG BOTTOM EDGE OF RECTANGLE FOR SIGN CHANGES IN IMAG(F).
26 IF (KR .EQ. JRT) KEDGE = 3
IF (KEDGE .EQ. 3) GO TO 31
KR = KR + 1
CALL FIND F (KR,KI,F)
IF ((AIMAG(PREV F) .GT. 0.0 .AND. AIMAG(F) .GT. 0.0))
  GO TO 20
IF (KR .EQ. 0) GO TO 2d
DO 27 K=1,NK
  IF (KR .EQ. KEDGE(K)) GO TO 20
27 CONTINUE
2E FO0 = PREV F
F10 = F
LI = JB(I)
LH = KR-1
GO TO 48

C SEARCH ALONG RIGHT EDGE OF RECTANGLE FOR SIGN CHANGES IN IMAG(F).
31 IF (K1 .EQ. JTOP) KEDGE = 4
  IF (KEDGE .EQ. 4) GO TO 36
  KI = KI+1
  CALL FINF F (KR,KI,F)
IF ((AIMAG(PREV F) .GT. 0.0 .AND. AIMAG(F) .GT. 0.0))
  GO TO 20
IF (NR .EQ. 0) GO TO 33
DO 32 K=1,NR
  IF (KI .EQ. KEDGE(K)) GO TO 20
32 CONTINUE
33 F10 = PREV F
F11 = F
LI = KI-1
LH = JRT-1
GO TO 53

C SEARCH ALONG TOP EDGE OF RECTANGLE FOR SIGN CHANGES IN IMAG(F).
36 IF (KR .EQ. JLT) GO TO 80
  KR = KR-1
  CALL FIND F (KR,KI,F)
IF ((AIMAG(PREV F) .GT. 0.0 .AND. AIMAG(F) .GT. 0.0))
  GO TO 20
IF (NR .EQ. 0) GO TO 38
DO 37 K=1,NR
  IF (KR .EQ. KEDGE(K)) GO TO 20
37 CONTINUE
38 F11 = PREV F
F01 = F
LI = JTCP-1
LH = KR
GO TO 58

C ENTER MESH SQUARE FROM LEFT SIDE OR EXIT RECTANGLE AT RIGHT EDGE.
41 LH = LR+1
  IF (LP .LE. JRT-1) GO TO 42
  NR F3 = NR E3+1

212
K = EDGE3(NR E3) = LI+1

IF (MPRINT .NE. 0) PRINT 440
GO TO 10

40 F01 = F11
FC0 = F1C

43 CALL FINC F (LR+1,LI+1,F11)
CALL FINC F (LR+1,LI,F10)
LEDF = 1
ENTER R = 0.0
ENTER I = -AIMAG(F00)/AIMAG(F01-F10)
GO TO 60

C ENTER MESH SQUARE FROM LEFT SIDE OR EXIT RECTANGLE AT TOP EDGE.

50 LI = LI+1
IF (LI .GE. JTOP-1) GO TO 47
NR E4 = NR E4+1
K = EDGE4(NR E4) = LR
IF (MPRINT .NE. 0) PRINT 490
GO TO 10

47 F00 = F01
F10 = F11

48 CALL FINC F (LR,LI+1,F01)
CALL FINC F (LR+1,LI,F11)
LEDF = 2
ENTER R = -AIMAG(F00)/AIMAG(F10-F00)
ENTER I = 0.0
GO TO 60

C ENTER MESH SQUARE FROM RIGHT SIDE OR EXIT RECTANGLE AT LEFT EDGE.

51 LF = LR-1
IF (LR .GE. JLT) GO TO 52
NR E1 = NR E1+1
K = EDGE1(NR E1) = LI
IF (MPRINT .NE. 0) PRINT 490
GO TO 10

52 F11 = F01
F00 = FC0

53 CALL FINC F (LR,LI+1,F01)
CALL FINC F (LR+1,LI,F00)
LEDF = 3
ENTER R = 1.0
ENTER I = -AIMAG(F10)/AIMAG(F11-F10)
GO TO 60

C ENTER MESH SQUARE FROM TOP SIDE OR EXIT RECTANGLE AT BOTTOM EDGE.

50 LI = LI-1
IF (LI .LE. JBO1) GO TO 57
NR E2 = NR E2+1
K = EDGE2(NR E2) = LR+1
IF (MPRINT .NE. 0) PRINT 490
GO TO 10

57 F01 = FC0
F11 = F1C

213
CALL FINC F (LR,LI,FO0)
CALL FINC F (LR+1,LT,F10)

C

LEDGE = 4

F = IMAG(FO1)/AIMAG(F11-FO1)
ENTER I = 1.0
GL TD 60

C FOR DEBUGGING ONLY, PRINT CO-ORDINATE OF LOWER LEFT CORNER
OF CURRENT MESH SQUARE. RESULTING SET OF PRINTED
CO-ORDINATES GIVES TRACE OF EACH LINE ALONG WHICH
IMAG(F)=0.0 FROM ITS ENTRY ON THE EDGE OF THE
RECTANGLE TO ITS EXIT AT ANOTHER POINT ON THE
RECTANGLE.

60 IF (MPRINT .NE. 0) PRINT 960,LR,LI
960 FORMAT (I,20x,215)

C TEST FOR THERE BEING TWO (HYPERBOLIC) LINES ENTERING AND LEAVING
THE MESH SQUARE ALONG EACH OF WHICH IMAG(F)=0.0
IF SO, SET 'Ltwo' NON-ZERO.

Ltwo = 0
IF ((AIMAG(FOO) .GT. 0.0 .AND. AIMAG(F11) .GT. 0.0)
$ .AND. AIMAG(FO1) .LT. 0.0 .AND. AIMAG(F11) .LT. 0.0)
$ .OR. (AIMAG(FO0) .LT. 0.0 .AND. AIMAG(F11) .LT. 0.0)
$ .AND. AIMAG(FO1) .GT. 0.0 .AND. AIMAG(F11) .GT. 0.0))
Ltwo = 1
C TEST FOR THERE BEING AT LEAST ONE (HYPERBOLIC) LINE ENTERING AND
LEAVING THE MESH SQUARE ALONG WHICH REAL(F)=0.0
IF NOT, SET 'I90' TO ZERO.

I90 = 1
IF ((REAL(FO0) .GT. 0.0 .AND. REAL(F10) .GT. 0.0)
$ .AND. REAL(FO1) .LT. 0.0 .AND. REAL(F11) .LT. 0.0)
$ .OR. (REAL(FO0) .LT. 0.0 .AND. REAL(F10) .LT. 0.0)
$ .AND. REAL(FO1) .GT. 0.0 .AND. REAL(F11) .GT. 0.0))
I90 = 0
IF (Ltwo .EQ. 0 .AND. I90 .EQ. 0) GO TO 70

C COMPUTATION OF COEFFICIENTS TO BE USED IN DESCRIBING THE VARIATION
OF THE FUNCTION WITHIN A MESH SQUARE GIVEN THE
VALUES AT THE CORNERS OF THE SQUARE AND LINEAR
VARIATION ALONG ITS EDGES. ALSO, THE POSITION OF
CRUSSING OF THE HYPERBOLIC ASYMPTOTES (WHICH ARE
PARALLEL TO THE SIDES OF THE SQUARE) FOR THE LINES
IMAG(F)=0.0 ARE COMPUTED IF BOTH LINES (BRANCHES)
ENTER AND LEAVE THE SQUARE.

PARTS = FO0
AP = PART(1)
AI = PART(2)

PARTS = FO1-FO0
BR = PART(1)
BI = PART(2)

PARTS = F10-FJ0
CK = PART(1)
CI = PART(2)

PARTS = FO0+F11-FJ1-F10

214
C IN
Th
Cj
(190
*Ol
G)
TO
70
61
AL = AR*R1-Al*Rk
AL = AR+CI-Al+CF
AD = AR+CI-Al+DR
BC = BR+CI-BI*CF
BD = BR+CI-BI*DR
CD = CR+CI-CI*DR
C
IF (ABS(RD) .LT. ABS(CD)) GO TO 64
C
SOLUTION FOR TWO POSSIBLE POINTS AT WHICH THERE MAY BE CROSSINGS
OF THE (HYPERBOLIC) LINES REAL(F)=0.0 AND IMAG(F)=0.0.
A CROSSING POINT IS CHOSEN TO BE A ZERO IF THE
FUNCTION IF IT LIES WITHIN THE CURRENT MESH SQUARE
AND IF IT LIES ON THE BRANCH OF IMAG(F)=J*0 CURRENTLY
BEING FOLLOWED. MULTIPLE CROSSINGS ALONG THIS BRANCH
(ACTUALLY A SERIES OF CONNECTING HYPERBOLIC BRANCHES)
IN THE SAME OR IN DIFFERENT SQUARES ARE NOTED SINCE
THESE MUST LATER BE RESOLVED.
A = CD
b = (AD-HC)*0.5
C = AB
CALL QUIC (A+B,C,SOL,NR,SUL)
IF (NR SUL .EQ. 0) GO TO 70
DG 63 N=1,NR SOL
UREAL = SOL(N)
IF (UREAL .LT. 0.0) OR. UREAL .GT. 1.0) GO TO c3
D1 = BR+CF*UREAL
D2 = BI+CI*UREAL
IF (ABS(C1) .GE. ABS(D2)) UIMAG = -(AR+CR*UREAL)/D1
IF (ABS(C2) .GE. ABS(D1)) UIMAG = -(AI+CI*UREAL)/D2
IF (UIMAG .LT. 0.0) OR. UIMAG .GT. 1.0) GO TO c3
IF (LTWC .EQ. 0) GO TO o2
IF (ENTER R-CENTR R)*U REAL-CENTR R)*LT. 0.0)
S
GO TO o3
o2
THETA(1) = (LR+U REAL)*TMESH
THETA(2) = (LI+U IMAG)*TMESH
IF (MPINT .NE. 0) PRINT 962,LR,L1,THETA(1),THETA(2)
962
FORMAT (1,2F9.4,T14,3X,T14,3X,T14,3X,T14,3X,T14,3X)
NR ZT = NR ZT+1
ZEROS(1,NR ZT) = THETA(1)
ZEROS(2,NR ZT) = THETA(2)
IF (NR ZL .EQ. 0) NR Z = NR Z+1
NR ZL = NR ZL+1
o3
CONTINUE
GC TO 67
ALTERNATE SOLUTION FOR THE ABOVE. TWO FORMS ARE NEEDED SINCE
IN ANY GIVEN CASE, EITHER FORM (BUT NOT BOTH FORMS)
MAY BE INDETERMINATE.

64 A = 50
B = (AD+PC)*0.5
C = AC
CALL QUAL (A,B,C,SOL,NK SL)
IF (NR SCL *EQ. 0) GO TO 7J
D = 66 N+1,NR SCL
UIMAG = SOL(N)
IF (UIMAG .LT. 0.0 .OR. UIMAG .GT. 1.0) GO TO 66
D1 = CR+CR*UIMAG
D2 = CI+CI*UIMAG
IF (ABS(C1) .GE. ABS(D2)) UREAL = -(AR+BR*UIMAG)/D1
IF (ABS(C2) .GE. ARS(D1)) UREAL = -(AI+BI*UIMAG)/D2
IF (UREAL .LT. 0.0 .OR. UREAL .GT. 1.0) GO TO 66
IF (LTWC .EQ. 0) GO TO 65
IF ((ENTER P-CENTR R)*(U REAL-CENTR R) .LT. 0.0)
$ OR. (ENTER I-CENTR I)*(U IMAG-CENTR I) .LT. 0.0)
$ GO TO 66

65 THETA(1) = (LR+U REAL)*TMESH
THETA(2) = (LI+U IMAG)*TMESH
IF (IMPRINT .NE. 3) PRINT 965,LR,LI,THETA(1),THETA(2)
465 FORMAT (1,'14.3X,'14.4) NK ZT = NR ZT+1
ZEPOS(1,NF ZT) = THETA(1)
ZEPOS(2,NF ZT) = THETA(2)
IF (NR ZL .EQ. 0) NR Z = NR Z+1
IF ZL .EQ. 0 GO TO 66
66 CONTINUE
67 CONTINUE
70 CONTINUE
C TEST FOR EXIT FROM LEFT EDGE OF MESH SQUARE.
IF (LEDGE .EQ. 1) GO TO 72
IF ((AIMAG(FOO) .GT. 0.0 .AND. AIMAG(F01) .GT. 0.0)
$ .OR. (AIMAG(F00) .LT. 0.0 .AND. AIMAG(F01) .LT. 0.0))
$ GO TO 72
IF (LTWO .EQ. 0) GO TO 51
EXIT R = 0.0
EXIT I = -AIMAG(F00)/AIMAG(F01-F00)
IF ((ENTER R-CENTR R)*(EXIT R-CENTR R) .LT. 0.0)
$ OR. (ENTER I-CENTR I)*(EXIT I-CENTR I) .LT. 0.0)
$ GO TO 72
GO TO 51
C TEST FOR EXIT FROM BOTTOM EDGE OF MESH SQUARE.
72 IF (LEDGE .EQ. 2) GO TO 73
IF ((AIMAG(F00) .GT. 0.0 .AND. AIMAG(F10) .GT. 0.0)
$ .OR. (AIMAG(F00) .LT. 0.0 .AND. AIMAG(F10) .LT. 0.0))
$ GO TO 73
IF (LTWO .EQ. 0) GO TO 56
EXIT R = -AIMAG(F00)/AIMAG(F10-F00)
EXIT I = 0.0
IF ( (ENTER R-CENTR R)*(EXIT R-CENTR R) .LT. 0.0
OR. (ENTER I-CENTR I)*(EXIT I-CENTR I) .LT. 0.0) $ GO TO 73

73  IF (LEGGE .EQ. 3) GO TO 74
IF ( (AIMAG(F10) .GT. 0.0 .AND. AIMAG(F11) .GT. 0.0)
OR. (AIMAG(F10) .LT. 0.0 .AND. AIMAG(F11) .LT. 0.0)) $ GO TO 74
IF (LTWO .EQ. 0) GO TO 41
EXIT R = 1.0
EXIT I = -AIMAG(F10)/AIMAG(F11-F10)
IF ( (ENTER R-CENTR R)*(EXIT R-CENTR R) .LT. 0.0
OR. (ENTER I-CENTR I)*(EXIT I-CENTR I) .LT. 0.0) $ GO TO 74

GO TO 41

C TEST FOR EXIT FROM TOP EDGE OF MESH SQUARE.
74  IF (LEGGE .EQ. 4) GO TO 90
IF ( (AIMAG(F01) .GT. 0.0 .AND. AIMAG(F11) .GT. 0.0)
OR. (AIMAG(F01) .LT. 0.0 .AND. AIMAG(F11) .LT. 0.0)) $ GO TO 90
IF (LTWO .EQ. 0) GO TO 46
EXIT R = -AIMAG(F01)/AIMAG(F11-F01)
EXIT I = 1.0
IF ( (ENTER R-CENTR K)*(EXIT R-CENTR K) .LT. 0.0
OR. (ENTER I-CENTR I)*(EXIT I-CENTR I) .LT. 0.0) $ GO TO 90

GO TO 46

C 30 IF (VR.Z .NE. 0) CALL MESH (TMFIN, TOL, MPRINT, ZERUS, NP 2)
RETURN

C 90 PRINT 909, LR, LI
909 FORMAT ('F01', 'NO EXIT FROM MESH SQUARE', 216)
STOP

C 990 FORMAT('D01')

END
SUBROUTINE QUAD (A,B,C,SOL,NR_SOL)

C FOR ROOT-FINDER

C SOLUTION: FIND THE REAL ROOTS OF A QUADRATIC EQUATION OF THE
C FORM A*X**2+B*X+C=0.0, WHERE X IS CALLED 'SOL'
C IN THIS ROUTINE. THE NUMBER OF REAL ROOTS FOUND IS
C GIVEN BY 'NR SOL'. A VALUE OF 1 FOR 'NR SOL' RESULTS
C FROM THE QUADRATIC EQUATION APPROACHING LINEARITY.
C USED BY SUBROUTINES FZEROs AND NO MESH.

C DIMENSION SOL(2)

C ACOBSQ = A*C/B**2

IF (ABS(ACOBSQ) .LT. 0.5) GO TO 20

ARG = B**2-A*C
NR_SOL = 0
IF (ARG .LT. 0.0) RETURN
NR_SOL = 2
ROOT = SCRT(ARG)
SOL(1) = (-B+ROOT)/A
SOL(2) = (-B-ROOT)/A
RETURN

20 TERM = 1.0
SUM = 1.0
DO 21 K=1,50
TERM = TERM*((K-0.5)/(K+1.0))*ACOBSQ
SUM = SUM+TERM
IF (ABS(TERM) .LT. 1.0E-10) GO TO 22
21 CONTINUE
22 SOL(1) = -C/(2.0*B)*SUM
NR_SOL = 1
IF (ABS(A/(2.0*B)) .LT. 1.0E-30) RETURN
NR_SOL = 2
SOL(2) = -2.0*B/A-SOL(1)
RETURN

END
SUBROUTINE FIND F (JR, JI, F)
C FOR FOOT-FINDER
C
COMMON/INC, COS/MESH
DIMENSION PART(2)
COMPLEX F, THETA, PARTS
EQUIVALENCE (PART, PARTS)
C
PART(1) = JR*MESH
PART(2) = JI*MESH
THETA = PARTS
CALL FCT VAL (THETA, F)
RETURN
C
END
SUBROUTINE NO MESH (TMESH,TOL,MPRINT,ZEROS, NR Z)  
C FOR FOOT-FINDER  
C ROUTINE FOR FINDING EXACT (IN THE SENSE OF NO MESH APPROXIMATION)  
C LOCATIONS OF ZEROS OF THE FUNCTION, F, FOR WHICH  
C A COMPLETE, BUT APPROXIMATE, SET WAS FOUND IN  
C SUBROUTINE ZEROS.  
C TWO DEFICIENCIES REMAIN IN THIS ROUTINE. ONE IS  
C THAT ZEROS CLOSER THAN THE FINITE VALUE 'TOL' CANNOT  
C BE RESOLVED. THIS HAS CAUSED A PROBLEM A FEW TIMES  
C IN USAGE TO DATE, BUT HAS ALWAYS BEEN RESOLVED BY  
C USING A SMALLER VALUE OF 'TOL'. THE SECOND PROBLEM  
C IS THAT THE RESOLUTION OF 'MULTIPLE CROSSINGS',  
C WHICH OCCUR BECAUSE OF THE NON-ANALITICITY OF THE  
C MESH APPROXIMATION, IS SUCH THAT IN PRINCIPLE A  
C ZERO CAN BE MISSED. THERE IS NO EVIDENCE THIS HAS  
C EVER OCCURRED IN PRACTICE. IT MAY OR MAY NOT BE  
C COST EFFECTIVE TO DEVELOP THE PROGRAM FURTHER TO  
C CORRECT THESE TWO PROBLEMS.  
C  
COMMON/ZLS COM/NR ZT, NR ZLS(100)  
DIMENSION PART(2), SOL(2)  
COMPLEX ZEROS(1),  
$  \theta,  
$  $F, DFT, DLT,$  
$ PARTS,$  
$ ZEROSO(100), TMIN,$  
$ ATEMP, BTEMP, CTEMP, DTEMP,$  
$ FARRAY(4,4), FO0, FO1, F10, F11$  
EQUIVALENCE (PART(1), PARTS)  
C  
IF (MPRINT .NE. 0) PRINT 900  
900 FORMAT ('1', 'ITERATIONS TO REMOVE MESH')  
C  
NS = 1  
C  
FUNCTION=APHSON (FIRST-ORDER) ITERATION TO FIND EXACT LOCATIONS  
C OF THE ZEROS OF F.  
DO 29 JZ=1,NR ZT  
ZEROSO(JZ) = ZEROS(JZ)  
IF (MPRINT .NE. 0) PRINT 901, JZ  
901 FORMAT ('1', 18)  
21 IF (MPRINT .NE. 0) PRINT 902, ZEROS(JZ)  
902 FORMAT ('1', 10X, 2(F9.4))  
THETA = ZEROS(JZ)  
CALL F DFT (THETA, F, DFT)  
DLT = -F/DFT  
ZEROS(JZ) = ZEROS(JZ) + DLT  
IF (CABS(DLT) .GT. JZ*3*TOL) GO TO 21  
29 CONTINUE  
C  
FUNCTION=IF UNIQUE ZEROS COMPARED TO TOTAL NUMBER MINUS  
C MULTIPLETY DUE TO MULTIPLE CROSSINGS.  
C
30 IF (NR J#*(Fw. 1) RETURN
   JZ = 1
   DO 33 J=2,NR ZT
   MATCH = C
   JMJ = J = 1
   DO 32 J=1,JMJ
   IF (LRASJZEROS(J)-ZEROS(JJ)) GT TOL GO TO 3z
   IF (MPRINT .NE. J) PRINT 905, J, JJ
   905 FORMAT ('0', 'ZEROS N#', J3, 1X, 'AND', J3, 1X,
                           'ARE THE SAME')
   MATCH = 1
   32 CONTINUE
   IF (MATCH .NE. 0) GO TO 33
   JZ = JZ + 1
   ZEROS(JZ) = ZEROS(J)
   33 CONTINUE
C
   IF (JZ .EQ. NR ZT) RETURN
   IF (JZ .LT. NR ZT) GO TO 50
   PRINT 947, JZ, NR Z

   947 FORMAT ('0', 'WARNING--', J3, 1X, 'ZEROS FOUND ON', J3, 1X,
                           'PHASE LINES')
   NR Z = JZ
   RETURN
C
   THE FOLLOWING LOGIC IS NEEDED ONLY OCCASIONALLY, NAMELY, WHEN
   TEST INDICATES THAT A ZERO HAS BEEN MISSED IN THE
   INITIAL NEWTON-RAPHSON ITERATION. IT ALLOWS FOR
   STEP-WISE CHANGE IN THE FORM OF THE FUNCTION FROM
   THE MESH APPROXIMATION TO THE EXACT FORM.

   50 NS = NS*2
   IF (NS .GT. 16) GO TO 90
C
   IF 54 JZ=1, NR ZT
   IF (MPRINT .NE. 0) PRINT 901, JZ
   ZEROS(JJ) = ZEROS(JZ)
   DO 58 K=1, NS
   FK FRACT = FLOAT(K)/FLOAT(NS)
   FM FRACT = 1.0-FN FRACT

   51 THETA = ZEROS(JZ)
   IF (MPRINT .NE. 0) PRINT 902, ZEROS(JZ)
   CALL F DFDT (THETA, f, DFDT)
   IF (K .EQ. NS) GO TO 50
C
   JRO = REAL(THETA)/TMESH
   IF (REAL(THETA) .LT. 0.0) JRO = JRO-1
   JIO = AIMAG(THETA)/TMESH
   IF (AIMAG(THETA) .LT. 0.0) JIO = JIO-1
   DC 52 JRLoop=1,4
   DC 52 JILOOP=1,4
   JR = JFC+JRLoop-2
   J1 = J10+JILOOP-2

   52 CALL Finc F (JR, J1, FARRAY(JRLoop, JILOOP))
C

DMIN = 9.5E9
DG 67 JFLCP=1,3
Gc 67 JILLCP=1,3
F00 = FARRAY(JRLLOOP,JILLOOP)
F01 = FARRAY(JRLLOOP,JILGCP+1)
F10 = FARRAY(JRLLOOP+1,JILLOOP)
F11 = FARRAY(JRLLOOP+1,JILGCP+1)
JR = JRO+JFLCP-2
J1 = J10+JULLCP-2
REFR = REAL(ZEROS(JZ))/TMESH-JR
REFI = AIMAG(ZEROS(JZ))/TMESH-JI
ATEMP = F00
BTEMP = F01-F00
CTEMP = F10-F00
DTEMP = F00+F11-F01-F10
PARTS = (ATEMP+BTEMP*REFI+CTEMP*REFR+DTEMP*REFR*REFI)*FM*FRCT
+*FN*FRCT
AF = PART(1)
AI = PART(2)
PARTS = (BTEMP+DTEMP*REFR)*FM*FRCT+DFDT*(0.0,1.0)*TMESH*FN*FRCT
BS = PART(1)
BI = PART(2)
PARTS = (CTEMP+LTEMP*REFI)*FM*FRCT+DFDT*TMESH*FN*FRCT
CS = PART(1)
CI = PART(2)
PARTS = CTEMP*FM*FRCT
DS = PART(1)
DI = PART(2)

C

AB = AR*BI-AL*Bk
AC = AR*CI-AL*CR
AD = AR*CI-AL*DR
AC = AR*CI-AL*CR
BL = BR*CI-AL*DR
CD = CR*CI-AL*DR

C

IF (ABS(BR) .LT. ABS(CU)) GO TO 64
A = CD
F = (AD-BC)*0.5
C = A6
CALL QUAC (A,B,C,SOLNR, NR SOL)
IF (NR SOL .EQ. 0) GO TO 67
DO 62 N=1,NR SOL
UREAL = SOL(N)
IF (UREAL+REFR .LT. -0.030 OR UREAL+REFR .GT. 1.030) GO TO 62
D1 = BR*CK*UREAL
D2 = BI*CI*UREAL
IF (ABS(D1) .GE. ABS(D2)) UIMAG = -(AR*CR*UREAL)/D1
IF (ABS(D2) .GE. ABS(D1)) UIMAG = -(AI*CI*UREAL)/D2
IF (UIMAG+REFI .LT. -0.030 OR UIMAG+REFI .GT. 1.030) GO TO 62
PART(1) = (JR+UREAL+REFR)*TMESH
PART(2) = (JI+UIMAG+REFI)*TMESH
\begin{verbatim}
THETA = PARTS

DIST = CBABS(THETA-ZEROS(JZ))
IF (DIST .GT. DMIN) GO TO 62
DMIN = DIST
TMIN = THETA

62 CONTINUE
GO TO 67

64 A = ED
B = (AD+BC)*0.5
C = AC
CALL QUAC (A,B,C,SOL,NR SOL)
IF (NR SOL .EQ. 0) GO TO 67
DF 66 N=1,NR SOL
UIMAG = SOL(N)
IF (UIMAG+REFI .LT. -0.030 .OR. UIMAG+REFI .GT. 1.330) GO TO 66
D1 = CR+CR*UIMAG
D2 = CI+CI*UIMAG
IF (ABS(C1) .GE. ABS(D2)) UREAL = -(AR+BR*UIMAG)/D1
IF (ABS(C2) .GE. ABS(D1)) UREAL = -(AI+BI*UIMAG)/D2
IF (UREAL+REF & .LT. -0.003 .OR. UREAL+REF & .GT. 1.003) GO TO 66
PART(1) = (JR+UREAL+REF & )*TMESH
PART(2) = (JI+UIMAG+REFI)*TMESH
THETA = PARTS
DIST = CBABS(THETA-ZEROS(JZ))
IF (DIST .GT. DMIN) GO TO 66
DMIN = DIST
TMIN = THETA

66 CONTINUE
67 CONTINUE

IF (DMIN .GT. 9.0E9) GO TO 95
DEL T = TMIN-ZEROS(JZ)
ZEROS(JZ) = TMIN
IF (CA BS(DEL T) .LT. 2.03*TMESH) GO TO 57
GO TO 51

50 DEL T = -F/DFDT
ZEROS(JZ) = ZEROS(JZ)+DEL T
IF (CA BS(DEL T) .LT. 0.3*TMESH) GO TO 57
GO TO 51

57 IF (MPRINT .NE. 0) PRINT 907
907 FORMAT (*0*)
58 CONTINUE
59 CONTINUE
GO TO 30

C 90 PRINT 990,JZ,NR Z
990 FORMAT (*0*,'ONLY',13,IX,'ZEROS FOUND ON',13,IX,'PHASE LINES')
STOP

C 95 PRINT 995
995 FORMAT (*0*,'PROBLEMS IN SUBROUTINE NO MESH')
STOP
\end{verbatim}
<table>
<thead>
<tr>
<th>Role</th>
<th>Name and Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Director</td>
<td>Defense Nuclear Agency</td>
</tr>
<tr>
<td></td>
<td>DDST (Peter Haas)</td>
</tr>
<tr>
<td></td>
<td>RAAE (Carl Fitz)</td>
</tr>
<tr>
<td></td>
<td>RAAE (Dow Evelyn)</td>
</tr>
<tr>
<td></td>
<td>STTL (Tech Library)</td>
</tr>
<tr>
<td></td>
<td>RAEV</td>
</tr>
<tr>
<td></td>
<td>Washington, DC 20305</td>
</tr>
<tr>
<td>Director</td>
<td>National Security Agency</td>
</tr>
<tr>
<td></td>
<td>Attn: Oliver W. Bartlett W65</td>
</tr>
<tr>
<td></td>
<td>Fort George G. Meade, MD 20755</td>
</tr>
<tr>
<td>Director</td>
<td>Institute for Defense Analysis</td>
</tr>
<tr>
<td></td>
<td>Attn: Dr. Joel Bengston</td>
</tr>
<tr>
<td></td>
<td>400 Army-Navy Drive</td>
</tr>
<tr>
<td></td>
<td>Arlington, VA 22202</td>
</tr>
<tr>
<td>Director</td>
<td>Defense Advanced Research Projects</td>
</tr>
<tr>
<td></td>
<td>Agency</td>
</tr>
<tr>
<td></td>
<td>Dr. Donald Mann</td>
</tr>
<tr>
<td></td>
<td>James C. Goodwyn</td>
</tr>
<tr>
<td></td>
<td>1400 Wilson Blvd</td>
</tr>
<tr>
<td></td>
<td>Arlington, VA 22209</td>
</tr>
<tr>
<td>Commander</td>
<td>Naval Surface Weapons Center,</td>
</tr>
<tr>
<td></td>
<td>White Oak</td>
</tr>
<tr>
<td></td>
<td>WA-501 (Robert Jenkins)</td>
</tr>
<tr>
<td></td>
<td>WA-501 (John Downs)</td>
</tr>
<tr>
<td></td>
<td>Silver Spring, MD 20910</td>
</tr>
<tr>
<td>Director</td>
<td>Army Ballistic Research Labs.</td>
</tr>
<tr>
<td></td>
<td>Aberdeen Proving Grounds</td>
</tr>
<tr>
<td></td>
<td>Maryland 21005</td>
</tr>
<tr>
<td>Director</td>
<td>Any Electronics Command</td>
</tr>
<tr>
<td></td>
<td>A: AMSEL-NR-R-2</td>
</tr>
<tr>
<td></td>
<td>F0: Monmouth, NJ 07703</td>
</tr>
<tr>
<td>Commander</td>
<td>Commander Air Force Weapons Laboratory</td>
</tr>
<tr>
<td></td>
<td>Attn: John M. Kamm/SAS</td>
</tr>
<tr>
<td></td>
<td>Kirtland Air Force Base, NM 87117</td>
</tr>
<tr>
<td>Chief of Research and</td>
<td>HQ. USAF (AFTAC)</td>
</tr>
<tr>
<td>Development</td>
<td>Attn: TD-3</td>
</tr>
<tr>
<td>Department of the Army</td>
<td>TD-5</td>
</tr>
<tr>
<td></td>
<td>Washington, DC 20333</td>
</tr>
<tr>
<td>Chairman</td>
<td>CINCSAC</td>
</tr>
<tr>
<td>Joint Chiefs of Staff</td>
<td>Attn: Director-Communications and</td>
</tr>
<tr>
<td></td>
<td>Electronics, Chief Scientist</td>
</tr>
<tr>
<td></td>
<td>Offutt Air Force Base, NE 68113</td>
</tr>
<tr>
<td>Defense Communications</td>
<td>CINCNORAD</td>
</tr>
<tr>
<td>Agency</td>
<td>Attn: Chief Scientist</td>
</tr>
<tr>
<td>Office of MEECN Sys.</td>
<td>ESD</td>
</tr>
<tr>
<td>Engineering</td>
<td>L. G. Hanscom Field</td>
</tr>
<tr>
<td></td>
<td>Bedford, MA 01730</td>
</tr>
<tr>
<td>Director</td>
<td>AFCRL</td>
</tr>
<tr>
<td></td>
<td>Attn: CRU, S. Horowitz</td>
</tr>
<tr>
<td></td>
<td>L. G. Hanscom Field</td>
</tr>
<tr>
<td></td>
<td>Bedford, MA 01730</td>
</tr>
</tbody>
</table>

225