NOAA Technical Report ERL 212-WPL 16

U.S. DEPARTMENT OF COMMERCE
National Oceanic and Atmospheric Administration
Environmental Research Laboratories

A Program for Calculating Three-Dimensional Acoustic-Gravity Ray Paths in the Atmosphere

T. M. GEORGES

BOULDER, COLO.
AUGUST 1971

ONMENTAL RESEARCH LABORATORIES

n of the Environmental Research Laboratories is to study the oceans, inland ver and upper atmosphere, the space environment, and the earth, in search anding needed to provide more useful services in improving man's prospects influenced by the physical environment. Laboratories contributing to these

nces Laboratories (ESL): Geomagnetism, seismology, geodesy, and related earth sciences; earthquake processes, internal structure and accurate figure of the Earth, and distribution of the Earth's mass.

Atlantic Oceanographic and Meteorological Laboratories (AOML): Oceanography, with emphasis on the geology and geophysics of ocean basins, oceanic processes, sea-air interactions, hurrican research, and weather modification (Miami, Florida).

Pacific Oceanographic Laboratories (POL): Oceanography; geology and geophysics of the Pacific Basin and margins; oceanic processes and dynamics; tsunami generation, propagation, modification, detection, and monitoring (Seattle, Washington).

Atmospheric Physics and Chemistry Laboratory (APCL): Cloud physics and precipitation; chemical composition and nucleating substances in the lower atmosphere; and laboratory and field experiments toward developing feasible methods of weather modification.

Air Resources Laboratories (ARL): Diffusion, transport, and dissipation of atmospheric contaminants; development of methods for prediction and control of atmospheric pollution (Silver Spring, Maryland).

Geophysical Fluid Dynamics Laboratory (GFDL): Dynamics and physics of geophysical fluid systems; development of a theoretical basis, through mathematical modeling and computer simulation, for the behavior and properties of the atmosphere and the oceans (Princeton, New Jersey).

Research Flight Facility (RFF): Outfits and operates aircraft specially instrumented for research; and meets needs of NOAA and other groups for environmental measurements for aircraft (Miami, Florida).

National Severe Storms Laboratory (NSSL): Tornadoes, squall lines, thunderstorms, and other severe local convective phenomena toward achieving improved methods of forecasting, detecting, and providing advance warnings (Norman, Oklahoma).

Space Environment Laboratory (SEL): Conducts research in solar-terrestrial physics, provides services and technique development in areas of environmental monitoring, forecasting, and data archiving.

Aeronomy Laboratory (AL): Theoretical, laboratory, rocket, and satellite studies of the physical and chemical processes controlling the ionosphere and exosphere of the earth and other planets.

Wave Propagation Laboratory (WPL): Development of new methods for remote sensing of the geophysical environment; special emphasis on propagation of sound waves, and electromagnetic waves at millimeter, infrared, and optical frequencies.

Marine Minerals Technology Center (MMTC): Research into aspects of undersea mining of hard minerals: development of tools and techniques to characterize and monitor the marine mine environment; prediction of the possible effects of marine mining on the environment; development of fundamental mining technology (Tiburon, California).

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION

BOULDER, COLORADO 80302

U.S. DEPARTMENT OF COMMERCE
Maurice H. Stans, Secretary

NATIONAL OCEANIC AND ATMOSPHERIC ADMINISTRATION
Robert M. White, Administrator
ENVIRONMENTAL RESEARCH LABORATORIES
Wilmot N. Hess, Director

NOAA TECHNICAL REPORT ERL 212-WPL 16

A Program for Calculating Three-Dimensional Acoustic-Gravity Ray Paths in the Atmosphere

T. M. GEORGES

BOULDER, COLORADO
August 1971

For sale by the Superintendent of Documents, U. S. Government Printing Office, Washington, D. C. 20402
Price 50 cents

TABLE OF CONTENTS

A PROGRAM FOR CALCULATING THREE-DIMENSIONAL ACOUSTIC-GRAVITY RAY PATHS IN THE ATMOSPHERE

T. M. Georges

The ITS-Jones-Haselgrove ray-tracing program is adapted to calculate acoustic-gravity ray paths in a compressible atmosphere with arbitrary three-dimensional wind and temperature variability and spherical earth. The program and its use are described, including deck listings and sample runs. Application to ocean acoustics should be possible with little modification.

1. INTRODUCTION

This report documents the current state of development of a digital computer program for calculating three-dimensional ray paths of acoustic-gravity waves in the atmosphere. The program was devised in order to extend ray-tracing capabilities for acoustic-gravity waves to model atmospheres having arbitrary three-dimensional variability of wind and temperature fields. In addition, these fields can be time-variable (within certain limitations), permitting the calculation of Doppler shifts. One program applies to the entire spectrum of acoustic-gravity waves (i.e., those waves treated by Hines, 1960), although, for the longer waves, considerable care must be exercised in the interpretation of ray-path calculations when the atmospheric refractive index is not "slowly varying" in the W.K.B. sense [Budden, 1961].*

The program is written in FORTRAN language for the Control Data 3800 computer. No attempt has yet been made to run the program on other machines, but the availability of an IBM version of the radio ray-tracing program suggests that there would be little difficulty in adapting the program to run on IBM machines. The CRT plotting routines present the major compatibility problem.

The method for calculating the ray path is a numerical integration of Hamilton's canonical equations [Landau and Lifshitz, 1959] written in a form similar to that given by W. L. Jones [1969], except that the atmosphere is allowed to be compressible, and spherical polar coordinates (earth-centered) are used. The six equations are:

$$\frac{dr}{dt} = \frac{c^2 k_r \Omega}{2\Omega^2 - \omega_a^2 - c^2 k^2} + v_r \qquad (1)$$

$$\frac{d\theta}{dt} = \frac{1}{r}\left\{\frac{c^2 k_\theta (\Omega^2 - \omega_g^2)}{\Omega(2\Omega^2 - \omega_a^2 - c^2 k^2)} + v_\theta\right\} \qquad (2)$$

---

*The acoustic-gravity wave spectrum includes, as subsets, ordinary acoustic or sound waves (in the high-frequency limit), so-called "infrasound", as well as internal atmospheric gravity waves.

$$\frac{d\phi}{dt} = \frac{1}{r \sin\theta} \left\{ \frac{c^2 k_\phi (\Omega^2 - \omega_g^2)}{\Omega(2\Omega^2 - \omega_a^2 - c^2 k^2)} + v_\phi \right\} \tag{3}$$

$$\frac{dk_r}{dt} = \frac{\Omega}{c} \frac{\partial c}{\partial r} \frac{(\omega_a^2 - c^2 k^2)}{(2\Omega^2 - \omega_a^2 - c^2 k^2)} - \underset{\sim}{K} \bullet \frac{\partial \underset{\sim}{v}}{\partial r} + k_\theta \frac{d\theta}{dt} + k_\phi \sin\theta \frac{d\phi}{dt} \tag{4}$$

$$\frac{dk_\theta}{dt} = \frac{1}{r} \left\{ \frac{\Omega}{c} \frac{\partial c}{\partial \theta} \frac{(\omega_a^2 - c^2 k^2)}{(2\Omega^2 - \omega_a^2 - c^2 k^2)} - \underset{\sim}{K} \bullet \frac{\partial \underset{\sim}{v}}{\partial \theta} - k_\theta \frac{dr}{dt} + r k_\phi \cos\theta \frac{d\phi}{dt} \right\} \tag{5}$$

$$\frac{dk_\phi}{dt} = \frac{1}{r \sin\theta} \left\{ \frac{\Omega}{c} \frac{\partial c}{\partial \phi} \frac{(\omega_a^2 - c^2 k^2)}{(2\Omega^2 - \omega_a^2 - c^2 k^2)} - \underset{\sim}{K} \bullet \frac{\partial \underset{\sim}{v}}{\partial \phi} - k_\phi \sin\theta \frac{dr}{dt} - r k_\phi \cos\theta \frac{d\theta}{dt} \right\}, \tag{6}$$

where (r, θ, ϕ) are the spherical coordinates of the ray point with respect to the earth's center, (k_r, k_θ, k_ϕ) are the components of the wave vector $\underset{\sim}{K}$, and k is its magnitude, c is the local speed of sound, ω_a and ω_g are the corresponding acoustic-cutoff and Brunt-Väisälä frequencies, and Ω is the intrinsic wave frequency with respect to the air moving at a local wind velocity $\underset{\sim}{v}$, whose components are v_r, v_θ, v_ϕ. The wave frequency measured in a frame moving with the fluid is $\Omega = \omega - \underset{\sim}{k} \bullet \underset{\sim}{v}$, where ω is the angular wave frequency measured in a frame at rest with respect to the earth. A seventh equation integrates Doppler shift along the ray path:

$$\frac{d\omega}{dt} = \underset{\sim}{K}\bullet\left(\frac{\partial \underset{\sim}{v}}{\partial t}\right) - \frac{\Omega}{c}\frac{\partial c}{\partial t}\frac{(\omega_a^2 - c^2k^2)}{(2\Omega^2 - \omega_a^2 - c^2k^2)} \quad . \tag{7}$$

This equation may be used to estimate Doppler shift, provided it is small, and if the medium varies little during ray passage. The development of these equations is straightforward and follows the procedure given by Jones [1969]; however, the conversion to spherical coordinates becomes rather involved, and there the development of Brandstatter [1959] was followed. The Hamiltonian (invariant quantity) along the ray path is taken from the acoustic-gravity-wave dispersion relation (Hines, 1960, eq. 21):

$$H \equiv \Omega^4 - \Omega^2\omega_a^2 - \Omega^2c^2k^2 + \omega_g^2c^2k_H^2 = 0 \quad , \tag{8}$$

where k_H is the horizontal component of $\underset{\sim}{K}$. The local values of ω_a and ω_g are assumed to be given by their "isothermal" values, i.e.,

$$\omega_a = \frac{\gamma g}{2c} \tag{9}$$

$$\omega_g^2 = \frac{(\gamma - 1)g^2}{c^2} \quad , \tag{10}$$

where g is the acceleration of gravity and γ is the specific heat ratio. Certain questions about the definitions of these quantities in non-isothermal atmospheres, admittedly, have been bypassed here.

The basic framework of the computer program is essentially that of the ionospheric radio ray-tracing program developed (over a period of some seven years) by Dr. R. M. Jones of the Institute for Telecommunication Sciences. That program was first documented in an ESSA Technical Report [Jones, 1966] , and, although its details have evolved considerably since then, the basic programming principles remain essentially the same. An updated report is now in preparation [Jones and Stephenson, 1971] . Potential users should study these reports as well.

2. CHANGES IN CONVERSION

The major alterations in converting the program to acoustic-gravity rays lie in the subroutines HASEL and RINDEX. Formerly, HASEL contained Hamilton's equations written in Haselgrove's [1955] formalism, which explicitly contains the wave refractive index. RINDEX then calculated this refractive index from the Appleton-Hartree formula, given an electron-density distribution.

It is easier (and faster in execution) to write Hamilton's equations for acoustic-gravity waves in a form that does not explicitly contain the refractive index (the form that has been given above). Thus HASEL is entirely rewritten and RINDEX is eliminated entirely. Also, subroutines REACH, POLCAR and CARPOL are deleted, since they were needed for "free-space" paths, which do not occur with atmospheric waves. The remaining subroutines have been modified mainly with respect to the form of the printed or plotted output, and the nature of these changes is evident on comparison of program listings. Provision for punching "rayset" cards has been omitted. Naturally, all new subroutines had to be written to describe the atmospheric wind and temperature field. These are described in Appendix II.

3. INPUT

Data are input to the program via punched cards in the data deck and are stored in a 400-element array called W. The W array is also used to transmit other variables, not normally input, between subprograms. Parameters defining the atmospheric model, wave quantities, and other physical and control variables are all input to the W array. One element of the W array is carried on each card in the following format: Col. 1-3 identify the element number, Col. 4-17 contain the value of that W parameter, Col. 18-24 are provided for logical variables that permit unit conversion, and Col. 25-80 provide space for identifying comments. The W cards may be read in any order, and only those need be specified which differ from the previous case (see "Deck Setup"). The user wishing to quickly familiarize himself with the program's operation would do well to study the list of parameters stored in W (Table 1). Some W numbers have been omitted in Table 1 because they have been used for the ionospheric program and are not relevant to the atmospheric-wave case. The 300-series of W's generally are those which have been added in the adaptation of the program to acoustic-gravity waves. (To minimize confusion between the two versions of the program, the 300-series was added to the W array rather than substituting new meanings for unused lower numbers.)

The program, in its present form, will accept specification of wave parameters in only one format: the value of total wavelength (or k) (W309) and the initial direction of phase propagation (W255-257 and W263-265). In addition, a choice must be made between the acoustic or internal-gravity wave mode (W305), since waves of either type may possess the same $\underset{\sim}{K}$, but have quite different frequencies. A new wave frequency is calculated from the wave dispersion relation for each new initial value of $\underset{\sim}{K}$ (magnitude or direction); thus the option of specifying (or stepping) frequency explicitly is not presently available. Azimuth and elevation of transmission may still be stepped, however. Other input formats could be devised, utilizing various combinations of wave frequency, the magnitude or direction of $\underset{\sim}{K}$ or any of its components; the present format, however, has been found to be the most convenient and free of the complications of determining whether a real wave actually satisfies the input parameters.

4. OUTPUT

Raypath quantities are printed out in the format shown in Table 2. User controls the intervals along the ray path at which printout is produced (W180). Plots of the raypaths (projected on either vertical or horizontal planes) are optionally available with machines having CRT output facilities. The programs listed here were written for the Control Data Model 280 CRT plotting system. Those who use the plotting programs should, of course, check for compatibility with their particular system. Parameters controlling both plotted and printed output are input through the W array. Sample plots of each type are shown in figure 1, for the test case whose printout is Table 2.

TABLE 1

W ARRAY

A. General Control Variables

W3	angular wave frequency, ω, in rad./sec.
W14	east geographic longitude of transmitter, radians
W16	north geographic latitude of transmitter, radians
W17	elevation angle of transmission, radians
W18	azimuth angle of transmission (radians, positive E of N)
W19	earth's radius in km
W20	height of transmitter above earth in km
W40	receiver height above the earth in km
W41	= 1. for Runge-Kutta integration
	= 2. for Adams-Moulton integration without error checking
	= 3. for Adams-Moulton integration with relative error checking
	= 4. for Adams-Moulton integration with absolute error check
W42	maximum allowable single-step integration error (S.S.E.)
W43	maximum S.S.E./minimum S.S.E.
W44	initial integration step size in sec.
W45	maximum step size in sec.
W46	minimum step size in sec.
W47	factor by which to increase or decrease step size
W93	maximum allowable number of steps per hop
W180	number of steps per periodic printout
W254	maximum number of hops
W255	initial elevation angle of transmission, radians
W256	final elevation angle of transmission, radians
W257	step in elevation angle of transmission, radians
W263	initial azimuth of transmission, radians
W264	final azimuth of transmission, radians
W265	step in azimuth of transmission, radians
W272	= 0. if no plot desired
	= 1. to plot projection of ray path on vertical plane
	= 2. to plot projection of ray path on ground

TABLE 1 (Con't)

W299	number of equations to be integrated
	= 7. if Doppler shift to be calculated
	= 8. if absorption to be calculated
	= 6. otherwise
W301	ground range of ray point, km
W302	maximum value of ground range, km
W305	= 1. for acoustic mode
	= -1. for internal gravity mode
W309	total wave number of acoustic-gravity wave, km^{-1}
W303	maximum value of height above ground, km

B. Variables used to control plotting

Projection on a vertical plane (W 272 = 1.)

W274	latitude of left edge of plot, radians
W275	longitude of left edge, radians
W276	latitude of right edge, radians
W277	longitude of right edge, radians
W287	height above ground of plot bottom, km
W286	distance between scale divisions, km

Projection on the ground (W272 = 2.)

W274	latitude of left edge of plot, radians
W275	longitude of left edge, radians
W276	latitude of right edge, radians
W277	longitude of right edge, radians
W271	expansion factor for lateral deviation scale
W286	distance between range-scale divisions, km

C. Variables defining atmospheric models

W320	atmospheric molecular weight

TLNEAR:

W340	ground temperature in ^{o}K
W341	vertical temperature gradient in $^{o}K/km$

WCONST:

W370	constant radial wind in m/s
W371	constant southward wind in m/s
W372	constant eastward wind in m/s
W376	height gradient of eastward wind m/s/km

TABLE 1 (Con't)

WGAUSS:

W381	maximum of Gaussian eastward wind m/s
W382	height width (l/e) of Gaussian eastward wind, km
W383	latitude width of Gaussian eastward wind, radians
W384	longitude width of Gaussian eastward wind, radians
W385	height where wind maximizes, km
W386	colatitude where wind maximizes, radians
W387	longitude where wind maximizes, radians

WTIDE:

W390	amplitude of phi component, m/s
W391	amplitude of theta component, m/s
W392	vertical wavelength, km
W393	time in wave periods
W394	wave period in sec.

ULOGZ:

W372	eastward wind at a height of 1 km, m/s

VVORTEX:

W331	radius of vortex to V_{max}, km
W332	colatitude of vortex center, radians
W333	longitude of vortex center, radians
W335	V_{max}, m/s

D. Unit-Conversion Characters

A 1 in Col. 18 converts degrees to radians
A 1 in Col. 19 converts km to radians
A 1 in Col. 20 converts cycles/sec to radians/sec.
A 1 in Col. 21 converts feet to radians
A 1 in Col. 22 converts sec to radians/sec.
(or wavelength to wave number)

TABLE 2

SAMPLE PRINTED OUTPUT

ACOUSTIC-GRAVITY WAVE WITH WINDS BUT WITHOUT ABSORPTION

001 TEST RUN FOR ACOUSTIC-GRAVITY RAY TRACING PROGRAM 06/14/71

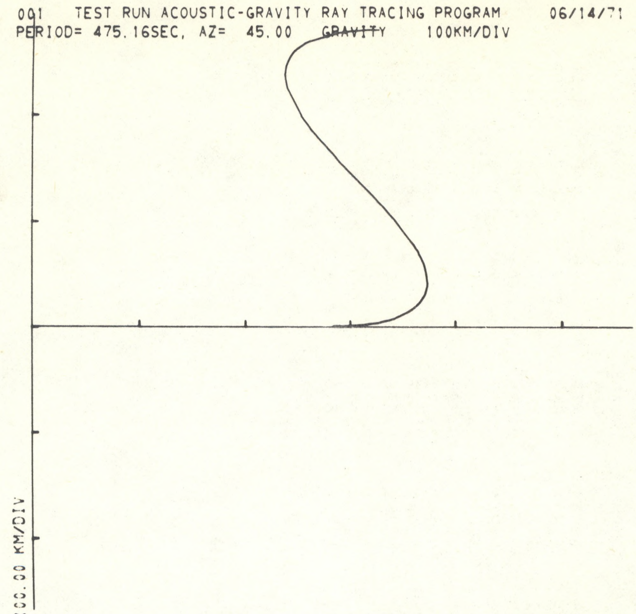
MODELS- TLNEAR WCONST

PERIOD= 475.157835 SEC., AZIMUTH OF TRANSMISSION = 45.000000 DEG., MODE = -1 , XLAT = 0.00+000 DEG., XLONG = 0.00+000 DEG.

ELEVATION ANGLE OF TRANSMISSION = -50.000000 DEG

ERROR		HEIGHT KM	RANGE KM	TRAVEL TIME SEC	AZIMUTH DEVIATION XMTR DEG	AZIMUTH DEVIATION LOCAL DEG	ELEVATION XMTR DEG	ELEVATION LOCAL DEG	INTRINSIC PERIOD SEC	POLARIZATION MAG X/Z	POLARIZATION PHASE DEG	PHASE VELOCITY M/S	DOPPLER SHIFT HZ	ABSORB. DB
0+000	XMTR	0.00	0.00	0.00				-50.000	475.158	1.209	3.879	21.046	0.000	0.00
7-008		87.267	74.946	4900.00	11.174	11.174	48.813	-38.989	400.662	0.827	5.562	25.797	0.000	0.00
1-007	MAX LONG	110.545	86.297	6100.00	15.309	15.310	51.394	-35.592	384.841	0.733	6.246	27.086	0.000	0.00
3-007		164.010	102.280	8900.00	28.907	28.909	57.261	-26.417	353.183	0.514	8.836	30.075	0.000	0.00
8-008		219.501	116.615	12700.00	59.169	59.171	61.087	-11.271	325.822	0.219	20.902	33.211	0.000	0.00
2-007	APOGEE	232.001	144.964	15500.00	82.634	82.637	56.881	1.146	320.291	0.081	-75.225	33.919	0.000	0.00
2-007	WAVE REV	232.001	144.964	15500.00	82.634	82.637	56.881	1.146	320.291	0.081	-75.225	33.919	0.000	0.00
2-007		227.915	162.641	16700.00	89.868	89.871	53.274	6.503	322.076	0.141	-33.671	33.685	0.000	0.00
4-008		143.312	245.935	22700.00	100.503	100.511	28.843	30.262	364.728	0.600	-7.577	28.903	0.000	0.00
3-008	MAX LONG	104.543	261.506	24700.00	97.916	97.929	20.450	36.491	388.772	0.757	-6.052	26.741	0.000	0.00
5-007		37.052	276.751	28300.00	88.684	88.708	6.358	45.586	440.125	1.038	-4.476	23.034	0.000	0.00
2-007	GRND REF	0.000	283.093	30494.84	80.830	80.862	-1.273	-50.000	475.158	1.209	3.876	21.034	0.000	0.00
2-007	RCVR	0.000	283.093	30494.84	80.830	80.862	-1.273	-50.000	475.158	1.209	3.876	21.034	0.000	0.00

THIS RAY CALCULATION TOOK 1.586 SEC



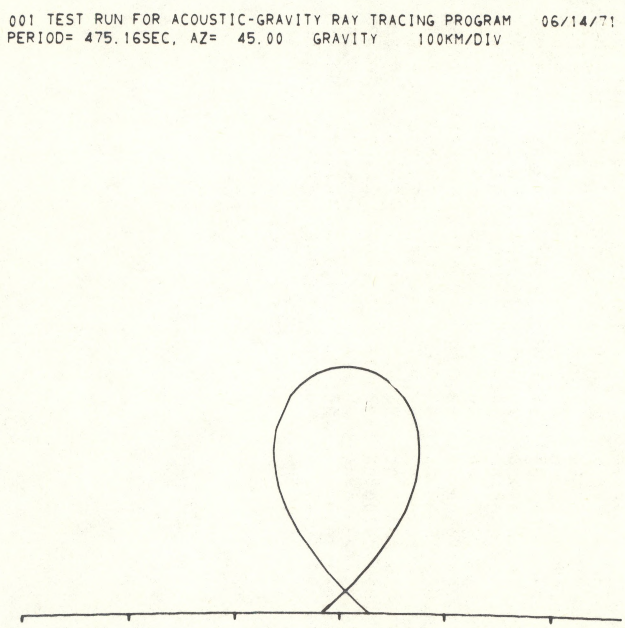


Figure 1. Raypath for an internal gravity wave in an isothermal atmosphere and a wind that increases with height by 0.1 m/s/km. The wind blows westward and the initial azimuth of transmission is 45°; thus, the raypath does not remain in a plane. The lower plot is a projection on a vertical plane oriented at 45° azimuth, and the upper plot is a projection of the raypath on a horizontal plane tangent to the earth at the center of the horizontal axis, which coincides with the azimuth of transmission.

5. DECK SET-UP

Arrangement of subroutine and data decks follows the same basic format as for ionospheric ray-tracing:

1. PROGRAM NITIAL
2. SUBROUTINE TRACE [+]
3. SUBROUTINE BACK UP
4. SUBROUTINE PRINTR
5. SUBROUTINE RAYPLT
6. SUBROUTINE PLOT
7. SUBROUTINE LABPLT
8. SUBROUTINE HASEL
9. SUBROUTINE RKAM
10. SUBROUTINE (ANY NAME)
    ENTRY TEMP
11. SUBROUTINE (ANY NAME)
    ENTRY WIND
12. (APPROPRIATE CONTROL CARDS)
13. DATA DECK [*]
    Title card
    Non-zero W cards
    Blank to signify end of W's

[+] The order of subroutine decks is unimportant.

[*] This sequence may be repeated indefinitely, changing only those W values that are to be different from previous sequence.

6. ACCURACY

The numerical integration subroutine RKAM performs error-checking functions and adjusts the integration step length to achieve roughly the magnitude of integration error specified by the user (W42). Thus, by adjusting W42, the user can trade low cost for high accuracy or vice-versa. The maximum single-step integration error is printed out in the "ERROR" column. User may also select from several available modes of integration and error checking by setting the value of W41. W41 = 3. is normally used (see Table 1). Further details on the operation of the numerical integration subroutine may be found in Jones [1966] .

7. TEST CASE

Those using the program for the first time, or those making changes in the program and wishing to check its operation, can run a "test case" and compare the output with that shown here. Input (W array) for the test case is given in Table 3. The printed and plotted output have already been shown as Table 2 and figure 1. Note that two runs are required to produce the two plots; one with W(272) = 1, and the other with W(272) = 2. No "absorption" is indicated in the output because an absorption equation has not yet been incorporated into HASEL.

TABLE 3

W VALUES INPUT FOR SAMPLE RUN

19	6370.		RADIUS OF THE EARTH, KM
14	0.	1	EAST GEOGRAPHIC LONGITUDE OF TRANSMITTER, KM
16	0.	1	NORTH GEOGRAPHIC LATITUDE OF TRANSMITTER, KM
20	0.		HEIGHT OF TRANSMITTER ABOVE GROUND, KM
41	3.		ADAMS-MOULTON INTEGRATION WITH RELATIVE ERROR CHECKING
40	0.		RECEIVER HEIGHT ABOVE EARTH
42	1.00000E-06		STEP SIZE DECREASED IF ERROR LARGER THAN THIS
43	50.		STEP SIZE INCREASED IF ERROR SMALLER THAN W42/W43
44	100.		INITIAL INTEGRATION STEP SIZE, SEC
45	1000.		MAXIMUM STEP SIZE, SEC
46	1.0		MINIMUM STEP LENGTH, SEC
47	5.00000E-01		FACTOR TO INCREASE OR DECREASE STEP LENGTH BY
93	400.		MAXIMUM STEPS PER HOP
180	10.		NUMBER OF STEPS BETWEEN PERIODIC PRINTOUT
254	1.0		MAXIMUM NUMBER OF HOPS
255	-50.0	1	INITIAL ELEVATION ANGLE IN DEG
263	45.0	1	INITIAL AZIMUTH ANGLE OF TRANSMISSION, DEG
272	1.0		PLOT PROJECTION ON VERTICAL PLANE
274	-200.0	1	LATITUDE OF LEFT EDGE OF PLOT IN KM
275	-200.	1	LONGITUDE OF LEFT EDGE OF PLOT IN KM
276	200.	1	LATITUDE OF RIGHT EDGE OF PLOT IN KM
277	200.	1	LONGITUDE OF RIGHT EDGE OF PLOT, KM
286	100.	1	DISTANCE BETWEEN TICK MARKS IN KM
287	0.0		HEIGHT OF BOTTOM OF PLOT ABOVE GROUND
302	1000.		MAXIMUM GROUND RANGE IN KM
303	500.		MAXIMUM HEIGHT, KM
305	-1.0		GRAVITY WAVE MODE
309	10.0	1	TOTAL WAVELENGTH IN KM
320	29.0		ATMOSPHERIC MOLECULAR WEIGHT (TLNEAR)
340	224.		GROUND TEMPERATURE, DEG K (TLNEAR)
341	0.1		TEMPERATURE GRADIENT, DEG K/KM (TLNEAR)
372	0.0		EASTWARD WIND AT GROUND, M/S (WCONST)
376	-0.1		HEIGHT GRADIENT OF UPHI, M/S/KM (WCONST)

8. ACKNOWLEDGEMENT

The program described in this report is an adaptation of the ionospheric radio ray-tracing program developed by Dr. R. M. Jones and Mrs. Judith J. Stephenson, both of the Office of Telecommunications. At least half of their program, including most of the logic and bookkeeping, has been retained in its original form. Where credit is due for a smoothly and efficiently running program, it is certainly theirs; where blame is due for programming errors or other bugs that inevitably creep into complex programs, it is almost certainly mine. It is a tribute to Dr. Jones' ingenuity and foresight that the adaptation to acoustic-gravity waves was so readily carried out. His help, and that of Mrs. Stephenson, in the conversion were indispensable and are here gratefully acknowledged.

9. REFERENCES

Brandstatter, J. J. (1959), The theory of propagation of rays in an inhomogeneous and anisotropic medium, Final Report - Project 2241, Stanford Research Institute, California.

Budden, K. G. (1961), Radio Waves in the Ionosphere, Cambridge University Press.

Haselgrove, J. (1955), Ray theory and a new method for ray tracing, in The Physics of the Ionosphere, The Physical Society, London, 355-364.

Hines, C. O. (1960), Internal atmospheric gravity waves at ionospheric heights, Can. J. Phys. 38, 1441-1481.

Jones, R. M. (1966), A three-dimensional ray-tracing computer program, ESSA Technical Report IER 17-ITSA 17.

Jones, R. M. and J. J. Stephenson (1971), A three-dimensional ray-tracing computer program - II, to be published as an ITS/OT technical report.

Jones, W. L. (1969), Ray tracing for internal gravity waves, J. Geophys. Res. 74, 2028-2033.

Landau, L. D., and E. M. Lifshitz (1959), Fluid Mechanics, Pergamon Press, Oxford, 256 ff.

Urick, R. J. (1967), Principles of Underwater Sound For Engineers, McGraw-Hill, New York.

APPENDIX A

Program Listing

```
      PROGRAM NITIAL
C           READS INPUT DATA, SETS INITIAL CONDITIONS BEFORE CALLING
C              TRACE, WHICH CALCULATES THE RAY PATH
      COMMON / SHARE / N, STEP, MODE, E1MAX, E1MIN, E2MAX, E2MIN, FACT,
     A RSTART, SSE
      COMMON / MODELS / MODEL(4)
      COMMON R(12), T / WW / ID(10), DUM, W(400)
      COMMON / CC / C, PCPR, PCPTH, PCPPH, PCPT
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (LON, W(14)), (LAT, W(16)), (BETA, W(17)), (AZ1, W(18
     A)), (EARTHR, W(19)), (XMTRH, W(20)), (INTYP, W(41)), (MAXERR, W(42
     B)), (ERATIO, W(43)), (STEP1, W(44)), (STPMAX, W(45)), (STPMIN, W(4
     C6)), (FACTR, W(47)), (RAY, W(67)), (MAXSTP, W(93)), (SKIP, W(180))
     D, (LINES, W(181)), (NUTEST, W(251)), (IHOP, W(253)), (ELBEG, W(255
     E)), (ELEND, W(256)), (ELSTEP, W(257)), (AZBEG, W(263)), (AZEND, W(
     F264)), (AZSTEP, W(265)), (PLT, W(272)), (KTOT, W(309)), (RNEWW, W(
     G291)), (WN, W(299)), (PNEWW, W(300)), (AORG, W(305))
      REAL MAXSTP, INTYP, MAXERR, LON, LAT, KTOT
      INTEGER DEG, CYCLE, FEET, SEC
      DATA (PI = 3.1415926536), (DEGS = 57.295779513), (MODEL = 4(1H ))
      CALL Q9EXUN
      NDATE = IDATE(0)
      SECOND = KLOCK(0) * .001
C                       READS W ARRAY
    1 READ 9, ID
      IF (EOF, 60) 2, 3
    2 CALL EXIT
    3 READ 10, K, W(K), DEG, KM, CYCLE, FEET, SEC
      IF (K.LT.0.OR.K.GT.400) CALL EXIT
      IF (DEG) W(K) = W(K) * PI / 180.
      IF (KM) W(K) = W(K) / EARTHR
      IF (CYCLE) W(K) = W(K) * 2. * PI
      IF (FEET) W(K) = W(K) * 3.048006096E - 4
      IF (SEC) W(K) = 2. * PI / W(K)
      IF (K.GT.0) GO TO 3
      IF (WN.LT.6.) WN = 6.
      IF (SKIP.EQ.0.) SKIP = MAXSTP
C          LET SUBROUTINES PRINTR AND RAYPLT KNOW THERE IS A NEW W ARRAY
      PNEWW = RNEWW = 1.
      IF (RAY.EQ.1.) PRINT 11
      IF (RAY.EQ.2.) PRINT 12
      IF (RAY.EQ.3.) PRINT 13
      PRINT 14, ID(1), ID, NDATE, MODEL
      DO 4 I = 1, 400
      IF (W(I).NE.0.) PRINT 15, I, W(I)
    4 CONTINUE
C
C             INITIALIZE PARAMETERS FOR INTEGRATION SUBROUTINE RKAM
      N = WN
      MODE = INTYP
      STEP = STEP1
      E1MAX = MAXERR
      E1MIN = MAXERR / ERATIO
      E2MAX = STPMAX
      E2MIN = STPMIN
      FACT = FACTR
C             LOOP ON AZIMUTH AND ELEVATION ANGLE
      NAZ = 1
      IF (AZSTEP.NE.0.) NAZ = (AZEND - AZBEG) / AZSTEP + 1.5
      NBETA = 1
      IF (ELSTEP.NE.0.) NBETA = (ELEND - ELBEG) / ELSTEP + 1.5
      DO 8 J = 1, NAZ
      AZ1 = AZBEG + (J - 1) * AZSTEP
      AZA = AZ1 * DEGS
      GAMMA = PI - AZ1
      SGAMMA = SIN(GAMMA)
      CGAMMA = COS(GAMMA)
      DO 7 I = 1, NBETA
      BETA = ELBEG + (I - 1) * ELSTEP
      CBETA = COS(BETA)
C           INITIAL COMPONENTS OF RAY POINT AND WAVE VECTOR K
      R(1) = EARTHR + XMTRH
      R(2) = PI / 2. - LAT
      R(3) = LON
      R(4) = SIN(BETA) * KTOT
      R(5) = CBETA * CGAMMA * KTOT
      R(6) = CBETA * SGAMMA * KTOT
C           CALCULATE WAVE FREQUENCY IN FIXED AND MOVING COORDINATES
      CALL WIND
      CALL TEMP
      C2 = C * C
      WA2 = 4.7E - 5 / C2
      WG2 = 3.85E - 5 / C2
      UDOTK = UR * R(4) + UTH * R(5) + UPH * R(6)
      COMEG = SQRT(((WA2 + C2 * KTOT * * 2) + AORG * SQRT((WA2 + C2 * K
     ATOT * * 2) * * 2 - 4. * WG2 * C2 * (R(5) * * 2 + R(6) * * 2))) / 2
     B.)
      W(3) = COMEG + UDOTK
      T = 0.
      RSTART = 1.
      EL = BETA * DEGS
C           INITIAL PRINTOUT
      IF (I.NE.1.AND.NPAGE.LT.3.AND.LINES.LE.17) GO TO 5
      NPAGE = LINES = 0
      IF (RAY.EQ.1.) PRINT 11
      IF (RAY.EQ.2.) PRINT 12
      IF (RAY.EQ.3.) PRINT 13
      PRINT 14, ID(1), ID, NDATE, MODEL
    5 FF = 2. * PI / W(3)
      XLAT = LAT * DEGS
      XLONG = LON * DEGS
      PRINT 16, FF, AZA, W(305), XLAT, XLONG
      NPAGE = NPAGE + 1
      PRINT 17, EL
      DO 6 NN = 7, N
    6 R(NN) = 0.
      CALL TRACE
      OSEC = SECOND
```

```
      SECOND = KLOCK(0) * .001
      DIFF = SECOND - OSEC
      PRINT 18, DIFF
    7 CONTINUE
    8 CONTINUE
      IF (PLT.NE.0.) CALL ENDPLT
      GO TO 1
C
    9 FORMAT (10A8)
   10 FORMAT (I3, E14.7, 5I1)
   11 FORMAT (1H1, 20X*ACOUSTIC WAVE WITH WINDS BUT WITHOUT GRAVITY OR
     AABSORPTION *)
   12 FORMAT (1H1, 20X*ACOUSTIC-GRAVITY WAVE WITHOUT WINDS OR ABSORPTIO
     AN*)
   13 FORMAT (1H1, 40X*ACOUSTIC-GRAVITY WAVE WITH WINDS BUT WITHOUT ABS
     AORPTION *)
   14 FORMAT (1X, A3, 2X, R5, 9A8, 20X, A8 / 1X, *MODELS-  *4(1X, A7))
   15 FORMAT (I4, E14.6)
   16 FORMAT(//* PERIOD=*F12.6*  SEC.,   AZIMUTH OF TRANSMISSION =*F12.
     A6* DEG.,  MODE =*F3.0* ,  XLAT =*E9.2* DEG.,  XLONG =*E9.2* DEG.*)
   17 FORMAT (   / 31X, 33HELEVATION ANGLE OF TRANSMISSION =, F12.6, 4H
     A DEG / )
   18 FORMAT (9X, 26HTHIS RAY CALCULATION TOOK , F8.3, 4H SEC)
      END
```



```
      SUBROUTINE TRACE
C             CONTROLS RAYPATH INTEGRATION FOR A GIVEN RAY, AND RETURNS
C             TO NITITAL FOR NEW RAY CONDITIONS
      DIMENSION ROLD(12), DROLD(12)
      COMMON / SHARE / N, STEP, MODE, E1MAX, E1MIN, E2MAX, E2MIN, FACT,
     A RSTART
      COMMON R(12), T, STP, DRDT(12) / WW / ID(10), DUM, W(400)
      EQUIVALENCE (EARTHR, W(19)), (GROUND, W(25)), (PERIGE, W(26)), (T
     AHERE, W(27)), (MINDIS, W(28)), (UNDER, W(29)), (HS, W(40)), (MAXST
     BP, W(93)), (SKIP, W(180)), (NUTEST, W(251)), (IHOP, W(253)), (HOP,
     C W(254)), (TPOLAR, W(266)), (RPOLAR, W(268)), (HMAX, W(284)),
     D (PLT, W(272)), (RAYBEG, W(292)), (RANGE, W(301)), (KTOT,W(309))
      REAL MAXSTP, KTOT
      LOGICAL HOME, WASNT, PASSED, UNDRGD, GROUND, PERIGE, THERE, MINDI
     AS, UNDER
      NHOP = HOP
      MAX = MAXSTP
      NSKIP = SKIP
      RSTART = 1.
      CALL HASEL
      H = R(1) - EARTHR
      HOME = DRDT(1) * (H - HS).GE.0.
      RAYBEG = 1.
      CALL PRINTR (8HXMTR    )
      IF (PLT.NE.0.) CALL RAYPLT
      RAYBEG = 0.
C
C             LOOP ON NUMBER OF HOPS
      DO 18 IHOP = 1, NHOP
      NUTEST = 0
C
C             LOOP ON MAXIMUM NUMBER OF STEPS PER HOP
      DO 10 J = 1, MAX
      DO 1 L = 1, N
      ROLD(L) = R(L)
    1 DROLD(L) = DRDT(L)
      TOLD = T
      CALL RKAM
      H = R(1) - EARTHR
      WASNT = .NOT.HOME
      HOME = DRDT(1) * (H - HS).GE.0.
      X = (DRDT(1) - DROLD(1)) * (T - TOLD)
      SMT = 0.
      IF (X.NE.0.) SMT = 0.5 * (R(1) - ROLD(1) + 0.5 * X) * * 2 / ABS(X
     A)
      UNDRGD = H.LT.0..OR.DRDT(1).GT.0..AND.DROLD(1).LT.0..AND.SMT.GT.H
      PASSED = (H - HS) * (ROLD(1) - EARTHR - HS).LT.0.
      IF (PASSED.AND.(.NOT.UNDRGD.OR.HS.GT.0.)) GO TO 13
      IF (HS.EQ.ROLD(1) - EARTHR.AND.DROLD(1) * DRDT(1).LT.0..AND.HOME)
     A GO TO 16
      IF (HOME.AND.WASNT.AND.(.NOT.UNDRGD.OR.HS.GT.0.)) GO TO 2
      IF (UNDRGD) GO TO 3
      GO TO 7
C
```

```
C              RAY MAY HAVE MADE A CLOSEST APPROACH
    2   IF (SMT.GT.ABS(H - HS)) GO TO 14
        NUTEST = 4
        CALL GRAZE (HS)
        IF (UNDER) GO TO 4
        IF (NUTEST.EQ.0.) GO TO 14
        GO TO 12
C
C              RAY WENT UNDERGROUND
    3   IF (DRDT(1).LT.0.) GO TO 6
    4   UNDER = .FALSE.
        DO 5 L = 1, N
        R(L) = ROLD(L)
    5   DRDT(L) = DROLD(L)
        T = TOLD
    6   CALL BACKUP (0.)
        R(1) = EARTHR
        DRDT(1) = - DRDT(1)
        R(4) = - R(4)
        RSTART = 1.
        CALL PRINTR (8HGRND REF)
        IF (HS.EQ.0.) GO TO 17
        H = 0.
        GO TO 9
C
    7   IF (DROLD(1).LT.0..AND.DRDT(1).GT.0.) CALL PRINTR (8HPERIGEE )
        IF (DROLD(1).GT.0..AND.DRDT(1).LT.0.) CALL PRINTR (8HAPOGEE  )
        IF (DROLD(2) * DRDT(2).LT.0.) CALL PRINTR (8HMAX LAT )
        IF (DROLD(3) * DRDT(3).LT.0.) CALL PRINTR (8HMAX LONG)
        DO 8 I = 4, 6
        IF (ROLD(I) * R(I).LT.0.) CALL PRINTR (8HWAVE REV)
    8   CONTINUE
    9   IF (PLT.NE.0.) CALL RAYPLT
        IF (RANGE.GE.W(302)) GO TO 11
        IF (MOD(J, NSKIP).EQ.0) CALL PRINTR (8H        )
       IF (H.GT.W(303)) GO TO 30
   10   CONTINUE
C
C              EXCEEDED MAXIMUM NUMBER OF STEPS
        NUTEST = 2
        CALL PRINTR (8HMAX STEP)
        RETURN
C
C              RAY PASSED MAXIMUM HEIGHT
30     CALL PRINTR(8HMAX HT   )
       RETURN
C              RAY REACHED MAXIMUM GROUND RANGE
   11   CALL PRINTR (8HMAX RNGE)
        RETURN
C
C              RAY MADE A CLOSEST APPROACH
   12   NUTEST = 4
        DRDT(1) = 0.
        CALL PRINTR (8HMIN DIST)
        IF (PLT.NE.0.) CALL RAYPLT
        GO TO 18
C
C              RAY CROSSED RECEIVER HEIGHT
   13   IF (HOME) GO TO 16
   14   DO 15 L = 1, N
        R(L) = ROLD(L)
   15   DRDT(L) = DROLD(L)
        T = TOLD
        RSTART = 1.
   16   CALL BACKUP (HS)
        R(1) = EARTHR + HS
   17   CALL PRINTR (8HRCVR    )
        IF (PLT.NE.0.) CALL RAYPLT
   18   HOME = .TRUE.
        RETURN
        END
```

```
      SUBROUTINE BACKUP (HS)
C           CONTROLS INTEGRATION WHEN RAY NEARS RECEIVER HEIGHT
      COMMON / SHARE / NN, STEP, MODE, E1MAX, E1MIN, E2MAX, E2MIN, FACT
     A, RSTART
      COMMON R(12), T, STP, DRDT(12) / WW / ID(10), DUM, W(400)
      EQUIVALENCE (EARTHR, W(19)), (UNDER, W(29)), (INTYP, W(41)), (STE
     AP1, W(44)), (NUTEST, W(251))
      REAL INTYP
      LOGICAL UNDER
      DO 1 I = 1, 10
      IF (DRDT(1).EQ.0.) GO TO 5
      STEP = - (R(1) - EARTHR - HS) / DRDT(1)
      STEP = SIGN(AMIN1(ABS(STP), ABS(STEP)), STEP)
      IF (ABS(R(1) - EARTHR - HS).LT..5E - 4.AND.STEP.LT.1.) GO TO 5
C             CALL PRINTR(8HHOMING   )
      MODE = 1
      RSTART = 1.
      CALL RKAM
    1 RSTART = 1.
C
      ENTRY GRAZE
      DO 2 I = 1, 10
      IF (DRDT(4).EQ.0.) GO TO 5
      STEP = - R(4) / DRDT(4)
      STEP = SIGN(AMIN1(ABS(STP), ABS(STEP)), STEP)
      IF (ABS(R(4)).LE.1.E - 6.AND.STEP.LT.1.) GO TO 5
C             CALL PRINTR(8HHOMING   )
      ROLD = R(1)
      MODE = 1
      RSTART = 1.
      CALL RKAM
      RSTART = 1.
      IF (R(1) - EARTHR.LT.0.) GO TO 4
      IF ((R(1) - EARTHR - HS) * (ROLD - EARTHR - HS).LT.0.) GO TO 3
    2 CONTINUE
      GO TO 5
    3 NUTEST = 0
      GO TO 5
    4 UNDER = .TRUE.
    5 MODE = INTYP
      STEP = STEP1
      RETURN
      END
```

```
      SUBROUTINE PRINTR (NWHY)
C           CONTROLS PRINTOUT OF RAYPATH QUANTITIES
      DIMENSION POLAR(4), G(3, 3), G1(3, 3), A1(3), B1(3), C1(3)
      COMMON / SHARE / N, STEP, MODE, E1MAX, E1MIN, E2MAX, E2MIN, FACT,
     A RSTART, SSE
      COMMON R(12), T / WW / ID(10), DUM, W(400)
      COMMON / CC / C, PCPR, PCPTH, PCPPH, PCPT
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (F, W(3)), (LON, W(14)), (LAT, W(16)), (BETA, W(17)),
     A (AZ1, W(18)), (EARTHR, W(19)), (XMTRH, W(20)), (LINES, W(181)), (
     BNUTEST, W(251)), (IHOP, W(253)), (APHT, W(270)), (RAYBEG, W(292)),
     C (PNEWW, W(300)), (THETA, R(2)), (PHI, R(3)), (RANGE, W(301))
      REAL LON, LAT, KSQ
      DATA (PI = 3.141592654), (DEGS = 57.295779513)
C
      IF (PNEWW.EQ.0.) GO TO 1
C             NEW W ARRAY -- REINITIALIZE
      PNEWW = 0.
      ABSORB = 0.
C           MATRIX TO CONVERT TO RECTANGULAR COORDINATE SYSTEM
      SPL = - SIN(LON)
      CPL = COS(LON)
      SL = SIN(LAT)
      CL = COS(LAT)
      G(1, 1) = CPL * CL
      G(1, 2) = SPL
      G(1, 3) = - SL * CPL
      G(2, 1) = - SPL * CL
      G(2, 2) = CPL
      G(2, 3) = SL * CPL
      G(3, 1) = SL
      G(3, 2) = 0.
      G(3, 3) = CL
      DENM = G(1, 1) * G(2, 2) * G(3, 3) + G(1, 2) * G(3, 1) * G(2, 3)
     A+ G(2, 1) * G(3, 2) * G(1, 3) - G(2, 2) * G(3, 1) * G(1, 3) - G(1,
     B 2) * G(2, 1) * G(3, 3) - G(1, 1) * G(3, 2) * G(2, 3)
      G1(1, 1) = (G(2, 2) * G(3, 3) - G(3, 2) * G(2, 3)) / DENM
      G1(1, 2) = (G(3, 2) * G(1, 3) - G(1, 2) * G(3, 3)) / DENM
      G1(1, 3) = (G(1, 2) * G(2, 3) - G(2, 2) * G(1, 3)) / DENM
      G1(2, 1) = (G(3, 1) * G(2, 3) - G(2, 1) * G(3, 3)) / DENM
      G1(2, 2) = (G(1, 1) * G(3, 3) - G(3, 1) * G(1, 3)) / DENM
      G1(2, 3) = (G(2, 1) * G(1, 3) - G(1, 1) * G(2, 3)) / DENM
      G1(3, 1) = (G(2, 1) * G(3, 2) - G(3, 1) * G(2, 2)) / DENM
      G1(3, 2) = (G(3, 1) * G(1, 2) - G(1, 1) * G(3, 2)) / DENM
      G1(3, 3) = (G(1, 1) * G(2, 2) - G(2, 1) * G(1, 2)) / DENM
      RO = EARTHR + XMTRH
C           XMTR LOCATION IN EARTH-CENTERED RECTANGULAR COORDINATES
      XR = RO * G(1, 1)
      YR = RO * G(2, 1)
      ZR = RO * G(3, 1)
      COSTHR = G(3, 1)
      SINTHR = SIN(ACOS(COSTHR))
      PHIR = ATAN2(YR, XR)
      ALPH = ATAN2(G(3, 2), G(3, 3))
```

```
1     HKSQ = R(5) * * 2 + R(6) * * 2
      KSQ = HKSQ + R(4) * * 2
      UDOTK = UR * R(4) + UTH * R(5) + UPH * R(6)
      COMEG2 = (W(3) - UDOTK) * * 2
      C2 = C * C
      WA2W = 4.7E - 5 / C2 / COMEG2
      WG2W = 3.85E - 5 / C2 / COMEG2
      V = 1. - WA2W - C2 * KSQ / COMEG2 + WG2W * C2 * HKSQ / COMEG2
      V = SSE
C
      H = R(1) - EARTHR
      SINTH = SIN(THETA)
      COSTH = COS(THETA)
      ANGDEG = ATAN2(R(4), SQRT(R(5) * * 2 + R(6) * * 2)) * DEGS
      TIME = T
      VPHASE = W(3) / SQRT(KSQ) * 1.E3
      TAU = 2. * PI / SQRT(COMEG2)
      DENOM = COMEG2 - HKSQ * C2
      POLARR = R(4) * SQRT(HKSQ) * C2 / DENOM
      POLARI = - 0.3 * .0098 * SQRT(HKSQ) / DENOM
      POLMAG = SQRT(POLARR * * 2 + POLARI * * 2)
     IF(POLARR.NE.0.)POLANG=ATAN(POLARI/POLARR)*57.2958
      DOPPLER = R(7) / 2. / PI
      IF (RAYBEG.EQ.0.) GO TO 2
      PRINT 6
C             CARTESIAN COORDINATES OF RAY POINT RELATIVE TO XMTR
2     XP = R(1) * SINTH * COS(PHI) - XR
      YP = R(1) * SINTH * SIN(PHI) - YR
      ZP = R(1) * COSTH - ZR
      EPS = XP * G1(1, 1) + YP * G1(1, 2) + ZP * G1(1, 3)
      ETA = XP * G1(2, 1) + YP * G1(2, 2) + ZP * G1(2, 3)
      ZETA = XP * G1(3, 1) + YP * G1(3, 2) + ZP * G1(3, 3)
      RCE2 = ETA * * 2 + ZETA * * 2
      RCE = SQRT(RCE2)
      RANGE = EARTHR * ATAN2(RCE, EARTHR + EPS)
      SR = SQRT(RCE2 + EPS * * 2)
      IF (SR.GE.1.E - 6) GO TO 3
      PRINT 7, V, NWHY, H, RANGE, TIME, ANGDEG, TAU, POLMAG, POLANG, VP
     AHASE, DOPPLER, ABSORB
      GO TO 5
3     ANGE = ATAN2(EPS, RCE)
      EL = ANGE * DEGS
      IF (RCE.NE.0.) GO TO 4
      PRINT 8, V, NWHY, H, RANGE, TIME, EL, ANGDEG, TAU, POLMAG, POLANG
     A, VPHASE, DOPPLER, ABSORB
      GO TO 5
4     ANGA = ATAN2(ETA, ZETA)
      ANA = ANGA - ALPH
      SINANA = SIN(ANA)
      SINPHI = SINANA * SINTHR / SINTH
      COSPHI = - COS(ANA) * COS(PHI - PHIR) + SINANA * SIN(PHI - PHIR)
     A* COSTHR
      AZA = 180. - AMOD(540. - (ATAN2(SINPHI, COSPHI) - ATAN2(R(6), R(5
     A))) * DEGS, 360.)
      AZDEV = 180. - AMOD(540. - (AZ1 - ANGA) * DEGS, 360.)
      PRINT 9, V, NWHY, H, RANGE, TIME, AZDEV, AZA, EL, ANGDEG, TAU, PO
     ALMAG, POLANG, VPHASE, DOPPLER, ABSORB
5     LINES = LINES + 1
      RETURN
C
6     FORMAT (51X, 7HAZIMUTH / 38X, 6HTRAVEL, 6X, 9HDEVIATION, 9X, 9HEL
     AEVATION, 5X, 9HINTRINSIC, 3X, 12HPOLARIZATION, 5X, 5HPHASE, 3X, 7H
     BDOPPLER, 2X, 7HABSORB. / 1X, 5HERROR, 14X, 6HHEIGHT, 3X, 5HRANGE,
     C5X, 4HTIME, 5X, 4HXMTR, 5X, 5HLOCAL, 4X, 4HXMTR, 5X, 5HLOCAL, 3X,
     D6HPERIOD, 4X, 3HMAG, 6X, 5HPHASE, 3X, 8HVELOCITY, 2X, 5HSHIFT, 5X,
     E 2HDB / 22X, 2HKM, 7X, 2HKM, 7X, 3HSEC, 6X, 3HDEG, 6X, 3HDEG, 6X,
     F3HDEG, 6X, 3HDEG, 5X, 3HSEC, 6X, 3HX/Z, 7X, 3HDEG, 5X, 3HM/S, 7X,
     G2HHZ)
7     FORMAT (1X, E6.0, 1X, A8, 3F9.2, 27X, 6F9.3, F8.2)
8     FORMAT (1X, E6.0, 1X, A8, 3F9.2, 18X, 7F9.3, F8.2)
9     FORMAT (1X, E6.0, 1X, A8, 2F9.3,F9.2,9F9.3,F8.2)
      END
```

```
      SUBROUTINE RAYPLT
C             W(272)=1. PLOTS PROJECTION OF RAYPATH ON VERTICAL PLANE
C             W(272)=2. PLOTS PROJECTION OF RAYPATH ON GROUND
      COMMON / PLT / XL, XR, YB, YT, TICKX, TICKY, RESET
      COMMON R(6) / WW / ID(10), DUM, W(400)
      EQUIVALENCE                          (EARTHR, W(19)), (PLT,
     AW(272)), (LLAT, W(274)), (LLON, W(275)), (RLAT, W(276)), (RLON, W(
     B277)), (TIC, W(286)), (HB, W(287)), (RNEWW, W(291)), (RAYBEG, W(29
     C2)), (FACTR, W(271))
      EQUIVALENCE (TH, R(2)), (PH, R(3))
      REAL LLAT, LLON, LTIC
      IF (RNEWW.EQ.0.) GO TO 2
C
C             NEW W ARRAY -- REINITIALIZE
      RNEWW = 0.
      RESET = 1.
      SW=1.
      CW=0.
      SLM = SIN(LLAT)
      CLM = COS(LLAT)
      SRM = SIN(RLAT)
      CRM = COS(RLAT)
     CDPHI=COS(LLON)
     PHL=ATAN2(SIN(LLON)*CLM,CDPHI*CLM)
     CTHL=SLM
      STHL = SIN(ACOS(CTHL))
     CDPHI=COS(RLON)
     PHR=ATAN2(SIN(RLON)*CRM,CDPHI*CRM)
     CTHR=SRM
      STHR = SIN(ACOS(CTHR))
      CLR = CTHL * CTHR + STHL * STHR * COS(PHL - PHR)
      SLR = SQRT(1. - CLR * * 2)
      IF (PLT.EQ.2.) GO TO 1
      FACTR = 1.
      RO = EARTHR + HB
      ALPHA = .5 * ACOS(CLR)
      XR = RO * SIN(ALPHA)
      XL = - XR
      YB = RO * COS(ALPHA)
      GO TO 2
    1 SPHLR = SIN(PHR - PHL)
      XL = 0.
      XR = EARTHR * ACOS(CLR)
      SKRU = ATAN2(STHR * SPHLR, (CTHR - CTHL * CLR) / STHL)
C
    2 STH = SIN(TH)
      CTH = COS(TH)
      CR = CTHR * CTH + STHR * STH * COS(PHR - PH)
      CL = CTHL * CTH + STHL * STH * COS(PHL - PH)
      CEA = ATAN2(CR - CL * CLR, CL * SLR)
      NEW = 0
      IF (RAYBEG.EQ.1.) NEW = 1
      IF (PLT.EQ.2.) GO TO 3
      CALL PLOT (R(1) * SIN(CEA - ALPHA), R(1) * COS(CEA - ALPHA), NEW)
      RETURN
    3 SL = SQRT(1. - CL * * 2)
      ALPHA = SKRU - ATAN2(STH * SIN(PH - PHL), (CTH - CTHL * CL) / STH
     AL)
      CALL PLOT (EARTHR * CEA, EARTHR * ASIN(SL * SIN(ALPHA)), NEW)
      RETURN
C
C             DRAW AXES AND CALL FOR LABELING AND TERMINATION OF THIS PLOT
      ENTRY ENDPLT
      DTIC = TIC * EARTHR
      IF (PLT.EQ.2.) GO TO 5
      R1 = EARTHR - TICKX
      X = XL
      Y = YB
      CALL PLOT (X, Y, 1)
      NTIC = 2
      IF (TIC.NE.0.) NTIC = NTIC + 2. * ALPHA / TIC
      NLINE = MAXOF(1, 100 / NTIC)
      DO 4 I = 1, NTIC
      ANG = - ALPHA + (I - 1) * TIC
      CALL PLOT (R1 * SIN(ANG), R1 * COS(ANG), 0)
      CALL PLOT (X, Y, 0)
      DO 4 J = 1, NLINE
      ANG = ANG + TIC / NLINE
      X = EARTHR * SIN(ANG)
      Y = EARTHR * COS(ANG)
    4 CALL PLOT (X, Y, 0)
      CALL PLOT (XR, YB, 0)
      GO TO 8
    5 LTIC = DTIC / FACTR
      TICY = XL + TICKY
      NTIC = YT / LTIC
      TIC1 = - LTIC * NTIC
      CALL PLOT (XL, YB, 1)
      NTIC = 2 * NTIC + 1
      DO 6 I = 1, NTIC
      Y = TIC1 + (I - 1) * LTIC
      CALL PLOT (XL, Y, 0)
      CALL PLOT (TICY, Y, 0)
    6 CALL PLOT (XL, Y, 0)
      CALL PLOT (XL, YT, 0)
      CALL PLOT (XL, 0., 1)
      NTIC = (XR - XL) / DTIC
      DO 7 I = 1, NTIC
      X = I * DTIC
      CALL PLOT (X, 0., 0)
      CALL PLOT (X, TICKX, 0)
    7 CALL PLOT (X, 0., 0)
      CALL PLOT (XR, 0., 0)
    8 CALL LABPLT
      CALL PLTEND
      RETURN
      END
```

```
      SUBROUTINE PLOT (X, Y, NEW)
      COMMON / PLT / XMINO, XMAXO, YMINO, YMAXO, TICKX, TICKY, RESET
      COMMON / DD / INT, IOR, IT, IS, IC, ICC, IX, IY
      COMMON / WW / ID(10), WO, W(400)
      EQUIVALENCE (PLT, W(272)), (FACTR, W(271))
      DATA (INITAL = 1), (FACTR = 1.), (MINX = 0), (MINY = 0), (MAXX =
     A1023), (MAXY = 1023), (MINXO = 23), (MINYO = 23), (MAXXO = 1023),
     B(MAXYO = 1023)
C
C            INITIALIZE LIBRARY PLOTTING ROUTINES
      IF (INITAL.EQ.0) GO TO 1
      INITAL = 0
      CALL DDINIT (2, 20HGEORGES TO RB3   RB3)
C
C            COMPUTE SCALE FACTORS
    1 IF (RESET.EQ.0.) GO TO 2
      RESET = 0.
      XSCALE = (MAXXO - MINXO) / (XMAXO - XMINO)
      YSCALE = XSCALE * FACTR
      IF (PLT.EQ.2.) YMINO = (MINYO - (MAXYO - MINYO) / 2) / YSCALE
      YMAXO = YMINO + (MAXYO - MINYO) / YSCALE
      TICKX = 0.01 * (MAXYO - MINYO) / YSCALE
      TICKY = TICKX * FACTR
      XMIN = XMINO - (MINXO - MINX) / XSCALE
      YMIN = YMINO - (MINYO - MINY) / YSCALE
      XMAX = XMAXO + (MAXX - MAXXO) / XSCALE
      YMAX = YMAXO + (MAXY - MAXYO) / YSCALE
C
C            START A NEW LINE
    2 IF (NEW.EQ.0) GO TO 3
      CALL DDQT
      IX = MINXO + (X - XMINO) * XSCALE
      IY = MINYO + (Y - YMINO) * YSCALE
      IF (IX.GE.MINX.AND.IX.LE.MAXX.AND.IY.GE.MINY.AND.IY.LE.MAXY) CALL
     A DDBP
      GO TO 17
C
    3 XS = X - XOLD
      YS = Y - YOLD
      IF (XS) 4, 5, 8
C             XS NEGATIVE
    4 X1 = XMAX
      X2 = XMIN
      GO TO 9
C             XS ZERO
    5 IF (YS) 6, 17, 7
    6 S1 = (YMAX - YOLD) / YS
      S2 = (YMIN - YOLD) / YS
      GO TO 14
    7 S1 = (YMIN - YOLD) / YS
      S2 = (YMAX - YOLD) / YS
      GO TO 14
C             XS POSITIVE
    8 X1 = XMIN
      X2 = XMAX
C
    9 IF (YS) 10, 11, 12
C             YS NEGATIVE
   10 Y1 = YMAX
      Y2 = YMIN
      GO TO 13
C             YS ZERO
   11 S1 = (X1 - XOLD) / XS
      S2 = (X2 - XOLD) / XS
      GO TO 14
C             YS POSITIVE
   12 Y1 = YMIN
      Y2 = YMAX
C
   13 S1 = AMAX1((X1 - XOLD) / XS, (Y1 - YOLD) / YS)
      S2 = AMIN1((X2 - XOLD) / XS, (Y2 - YOLD) / YS)
C
   14 S = SQRT(XS * * 2 + YS * * 2)
      IF (S2.LT.0..OR.S * S1 - S.GT.0.) GO TO 17
      IF (S1.LT.0.) GO TO 15
C             PREVIOUS POINT OFF GRAPH
      IX = MINXO + (XOLD + XS * S1 - XMINO) * XSCALE + 0.5
      IY = MINYO + (YOLD + YS * S1 - YMINO) * YSCALE + 0.5
      CALL DDBP
   15 IF (S * S2 - S.GT.0.) GO TO 16
C             CURRENT POINT OFF GRAPH
      IX = MINXO + (XOLD + XS * S2 - XMINO) * XSCALE + 0.5
      IY = MINYO + (YOLD + YS * S2 - YMINO) * YSCALE + 0.5
      CALL DDVC
      GO TO 17
C             CURRENT POINT ON GRAPH
   16 IX = MINXO + (X - XMINO) * XSCALE + 0.5
      IY = MINYO + (Y - YMINO) * YSCALE + 0.5
      CALL DDVC
C
C             EXIT ROUTINE
   17 XOLD = X
      YOLD = Y
      RETURN
C
C             TERMINATE THE CURRENT PLOT
      ENTRY PLTEND
      CALL DDFR
      RETURN
      END
```

```
      SUBROUTINE LABPLT
C             LABEL THE CURRENT PLOT
      DIMENSION LABEL(9), TYPE(3)
      COMMON / DD / INT, IOR, IT, IS, IC, ICC, IX, IY
      COMMON / WW / ID(10), DUM, W(400)
      EQUIVALENCE (F, W(3)), (AZ1, W(18)), (EARTHR, W(19)),
     A    (FACTR, W(271)), (PLT, W(272)), (TIC, W(286))
      REAL LTIC
     DATA (DEGS=57.295779513),(TYPE=8H GRAVITY,8H        ,8HACOUSTIC)
     NTYP=W(305)+2.
      IOR = IT = 0
      IS = 2
      IDSAV = ID(8)
      ID(8) = IDATE(0)
      IX = 0
      IY = 1023
      CALL DDTAB
      CALL DDTEXT (8, ID)
      ID(8) = IDSAV
      AZA = AZ1 * DEGS
      DTIC = TIC * EARTHR
      T = 6.28318 / F
      ENCODE (72, 1, LABEL) , T, AZA, TYPE(NTYP), DTIC
      IX = 0
      IY = 991
      CALL DDTAB
      CALL DDTEXT (9, LABEL)
      IF (PLT.EQ.1.) RETURN
      IOR = 1
      LTIC = DTIC / FACTR
      ENCODE (32, 2, LABEL) , LTIC
      IX = 0
      IY = 0
      CALL DDTAB
      CALL DDTEXT (4, LABEL)
      IOR = 0
      RETURN
C
    1 FORMAT (*PERIOD=*F7.2*SEC, AZ=*F7.2, 2X, A8, 2X, F5.0*KM/DIV*)
    2 FORMAT (F7.2, 24H KM/DIV                 '.)
      END
```

```
      SUBROUTINE HASEL
C            DIFFERENTIAL EQUATIONS FOR RAYPATH AND DOPPLER SHIFT
C            DRDT(1)=DR/DT.... DRDT(4)=DKR/DT, ETC.
      COMMON R(12), T, STP, DRDT(12) / WW / ID(10), DUM, W(400)
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      COMMON / CC / C, PCPR, PCPTH, PCPPH, PCPT
      REAL OMEG, OMEG2, KSQ
      DATA (W(67) = 3.)
      CALL TEMP
      CALL WIND
      COSTH = COS(R(2))
      SINTH = SIN(R(2))
      C2 = C * C
      WA2 = 4.7E - 5 / C2
      WG2 = 3.85E - 5 / C2
      OMEG = W(3) - R(4) * UR - R(5) * UTH - R(6) * UPH
      OMEG2 = OMEG * OMEG
      KSQ = R(4) * * 2 + R(5) * * 2 + R(6) * * 2
      DENOM = 2. * OMEG2 - WA2 - C2 * KSQ
      DIFF = (WA2 - C2 * KSQ) / DENOM * OMEG / C
      DRDT(1) = C2 * R(4) * OMEG / DENOM + UR
      DRDT(2) = (C2 * R(5) * (OMEG2 - WG2) / OMEG / DENOM + UTH) / R(1)
      DRDT(3) = (C2 * R(6) * (OMEG2 - WG2) / OMEG / DENOM + UPH) / R(1)
     A / SINTH
      DRDT(4) = DIFF * PCPR - R(4) * PUPX(1, 1) - R(5) * PUPX(2, 1) - R
     A(6) * PUPX(3, 1) + R(5) * DRDT(2) + R(6) * SINTH * DRDT(3)
      DRDT(5) = (DIFF * PCPTH - R(4) * PUPX(1, 2) - R(5) * PUPX(2, 2) -
     A R(6) * PUPX(3, 2) - R(5) * DRDT(1) + R(1) * R(6) * COSTH * DRDT(3
     B)) / R(1)
      DRDT(6) = (DIFF * PCPPH - R(4) * PUPX(1, 3) - R(5) * PUPX(2, 3) -
     A R(6) * PUPX(3, 3) - R(6) * SINTH * DRDT(1) - R(1) * COSTH * R(6)
     B* DRDT(2)) / R(1) / SINTH
C            DOPPLER SHIFT
      DRDT(7) = R(4) * PURPT + R(5) * PUTHPT + R(6) * PUPHPT - DIFF * P
     ACPT
      RETURN
      END
```

```
      SUBROUTINE RKAM
C             NUMERICAL INTEGRATION OF DIFFERENTIAL EQUATIONS
C             SEE JONES, 1966, ESSA TECH REPT IER-17/ITSA-17
      COMMON / SHARE / NN, SPACE, MODE, E1MAX, E1MIN, E2MAX, E2MIN, FAC
     AT, RSTART, SSE
      COMMON Y(12), T, STEP, DYDT(12)
      DIMENSION DELY(4, 12), BET(4), XV(5), FV(4, 12), YU(5, 12)
      DOUBLE PRECISION YU
      IF (RSTART.EQ.0.) GO TO 2
      LL = 1
      MM = 1
      IF (MODE.EQ.1) MM = 4
      ALPHA = T
      EPM = 0.0
      BET(1) = 0.5
      BET(2) = 0.5
      BET(3) = 1.0
      BET(4) = 0.0
      STEP = SPACE
      R = 19.0 / 270.0
      XV(MM) = T
      IF (E1MIN.LE.0.) E1MIN = E1MAX / 55.
      IF (FACT.LE.0.) FACT = 0.5
      CALL HASEL
      DO 1 I = 1, NN
      FV(MM, I) = DYDT(I)
    1 YU(MM, I) = Y(I)
      RSTART = 0.
      GO TO 3
    2 IF (MODE.NE.1) GO TO 9
C             RUNGE-KUTTA
    3 DO 5 K = 1, 4
      DO 4 I = 1, NN
      DELY(K, I) = STEP * FV(MM, I)
      Z = YU(MM, I)
    4 Y(I) = Z + BET(K) * DELY(K, I)
      T = BET(K) * STEP + XV(MM)
      CALL HASEL
      DO 5 I = 1, NN
    5 FV(MM, I) = DYDT(I)
      DO 6 I = 1, NN
      DEL = (DELY(1, I) + 2.0 * DELY(2, I) + 2.0 * DELY(3, I) + DELY(4,
     A I)) / 6.0
    6 YU(MM + 1, I) = YU(MM, I) + DEL
      MM = MM + 1
      XV(MM) = XV(MM - 1) + STEP
      DO 7 I = 1, NN
    7 Y(I) = YU(MM, I)
      T = XV(MM)
      CALL HASEL
      IF (MODE.EQ.1) GO TO 15
      DO 8 I = 1, NN
    8 FV(MM, I) = DYDT(I)
      IF (MM.LE.3) GO TO 3
C             ADAMS-MOULTON
    9 DO 10 I = 1, NN
      DEL = STEP * (55. * FV(4, I) - 59. * FV(3, I) + 37. * FV(2, I) -
     A9. * FV(1, I)) / 24.
      Y(I) = YU(4, I) + DEL
   10 DELY(1, I) = Y(I)
      T = XV(4) + STEP
      CALL HASEL
      XV(5) = T
      DO 11 I = 1, NN
      DEL = STEP * (9. * DYDT(I) + 19. * FV(4, I) - 5. * FV(3, I) + FV(
     A2, I)) / 24.
      YU(5, I) = YU(4, I) + DEL
   11 Y(I) = YU(5, I)
      CALL HASEL
      IF (MODE.LE.2) GO TO 15
C             ERROR ANALYSIS
      SSE = 0.0
      DO 12 I = 1, NN
      EPSIL = R * ABS(Y(I) - DELY(1, I))
      IF (MODE.EQ.3.AND.Y(I).NE.0.) EPSIL = EPSIL / ABS(Y(I))
      IF (SSE.LT.EPSIL) SSE = EPSIL
   12 CONTINUE
      IF (E1MAX.GT.SSE) GO TO 13
      IF (ABS(STEP).LE.E2MIN) GO TO 15
      LL = 1
      MM = 1
      STEP = STEP * FACT
      GO TO 3
   13 IF (LL.LE.1.OR.SSE.GE.E1MIN.OR.E2MAX.LE.ABS(STEP)) GO TO 15
      LL = 2
      MM = 3
      XV(2) = XV(3)
      XV(3) = XV(5)
      DO 14 I = 1, NN
      FV(2, I) = FV(3, I)
      FV(3, I) = DYDT(I)
      YU(2, I) = YU(3, I)
   14 YU(3, I) = YU(5, I)
      STEP = 2.0 * STEP
      GO TO 3
C             EXIT ROUTINE
   15 LL = 2
      MM = 4
      DO 16 K = 1, 3
      XV(K) = XV(K + 1)
      DO 16 I = 1, NN
      FV(K, I) = FV(K + 1, I)
   16 YU(K, I) = YU(K + 1, I)
      XV(4) = XV(5)
      DO 17 I = 1, NN
      FV(4, I) = DYDT(I)
   17 YU(4, I) = YU(5, I)
      IF (MODE.LE.2) RETURN
```

```
      E = ABS(XV(4) - ALPHA)
      IF (E.LE.EPM) GO TO 9
      EPM = E
      RETURN
      END
```

```
      SUBROUTINE WGAUSS
      COMMON / MODELS / MODEL(4)
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (UPHO, W(381)), (WH, W(382)), (WTH, W(383)), (WPH, W(
     A384)), (HO, W(385)), (THO, W(386)), (PHO, W(387))
      DATA (MODEL(2) = 6HWGAUSS)
      ENTRY WIND
      H = R(1) - W(19)
      IF (WH.NE.0.) GO TO 1
      EXR = 1.
      GO TO 2
 1    EXR = EXP( - (((HO - H) / WH) * * 2))
 2    IF (WTH.NE.0.) GO TO 3
      EXTH = 1.
      GO TO 4
 3    EXTH = EXP( - (((THO - R(2)) / WTH) * * 2))
 4    IF (WPH.NE.0.) GO TO 5
      EXPH = 1.
      GO TO 6
 5    EXPH = EXP( - (((PHO - R(3)) / WPH) * * 2))
 6    UPH = UPHO * EXR * EXTH * EXPH *  1.E-3
      IF (WH.NE.0.) GO TO 7
      PUPX(3, 1) = 0.
      GO TO 8
 7    PUPX(3, 1) = - 2. * UPH * (H - HO) / WH / WH
 8    IF (WTH.NE.0.) GO TO 9
      PUPX(3, 2) = 0.
      GO TO 10
 9    PUPX(3, 2) = - 2. * UPH * (R(2) - THO) / WTH / WTH + UPH
10    IF (WPH.NE.0.) GO TO 11
      PUPX(3, 3) = 0.
      GO TO 12
11    PUPX(3, 3) = - 2. * UPH * (R(3) - PHO) / WPH / WPH
12    RETURN
      END
```

```
      SUBROUTINE TLNEAR
      COMMON / MODELS / MODEL(4)
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / CC / C, PCPR, PCPTH, PCPPH, PCPT
      EQUIVALENCE (TGND, W(340)), (A, W(341)), (MW, W(320))
      REAL MW
      DATA (MODEL(1) = 6HTLNEAR), (TGND = 0.), (A = 0.)
      ENTRY TEMP
      H = R(1) - W(19)
      T = TGND + A * H
      C = SQRT(T / MW) * .108
      PCPR = A / 2. / MW / C * 108. * 108.E - 6
      RETURN
      END
```

```
      SUBROUTINE TTANH
      DIMENSION D(5), Z(4), B(4)
      COMMON / CC / C, PCPR, PCPTH, PCPPH, PCPT
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / MODELS / MODEL(4)
      EQUIVALENCE (MW, W(320))
      REAL MW
      DATA (D(1) = - 6.5), (D(2) = 3.5), (D(3) = - 3.), (D(4) = 18.), (
     AD(5) = 1.), (Z(1) = 15.), (Z(2) = 52.), (Z(3) = 95.), (Z(4) = 165.
     B), (B(1) = 10.), (B(2) = 7.5), (B(3) = 10.), (B(4) = 50.), (TO = 2
     C88.), (SUM = 0.), (MODEL(1) = 6HTTANH )
      COSH(X) = (EXP(X) + 1. / (EXP(X))) / 2.
      ENTRY TEMP
      H = R(1) - W(19)
      SUM = 0.
      DO 1 I = 1, 4
 1    SUM = SUM + B(I) * (D(I + 1) - D(I)) / 2. * LOGF(COSH((H - Z(I))
     A/ B(I)) / COSH(Z(I) / B(I)))
      T = TO + SUM + D(1) * H / 2.
      SUM = 0.
      DO 2 I = 1, 4
 2    SUM = SUM + (D(I + 1) - D(I)) / 2. * (TANH((H - Z(I)) / B(I)))
      PTPR = D(1) / 2. + SUM
      C = SQRT(T / MW) * 108.*1.E-3
      Q = 108. * 108. / 2. / C / MW
      PCPR = PTPR * Q*1.E-6
      RETURN
      END
```

```
      SUBROUTINE WCONST
      COMMON / MODELS / MODEL(4)
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (URO, W(370)), (UTHO, W(371)), (UPHO, W(372)), (WGRAD
     A, W(376))
      DATA (MODEL(2) = 6HWCONST)
      ENTRY WIND
      H = R(1) - W(19)
      UR = URO * 1.E - 3
      UTH = UTHO * 1.E - 3
      UPH = (UPHO + WGRAD * H) * 1.E - 3
      PUPX(3, 1) = WGRAD * 1.E - 3
      RETURN
      END
```

```
      SUBROUTINE WTIDE
      COMMON / MODELS / MODEL(4)
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (UPHO, W(390)), (UTHO, W(391)), (LAMZ, W(392)), (TP,
     AW(393)), (TAU, W(394)), (TIME, W(303))
      REAL LAMZ
      DATA (PI2 = 6.2831853), (MODEL(2) = 6H WTIDE)
      ENTRY WIND
      H = R(1) - W(19)
      ARG = PI2 * (H / LAMZ + TP)
      UPH = UPHO * COS(ARG)*1.E-3
      UTH = UTHO * SIN(ARG)*1.E-3
      Q = PI2 / LAMZ
      PUPX(3, 1) = - SIN(ARG) * Q * UPHO*1.E-3
      PUPX(2, 1) = COS(ARG) * Q * UTHO*1.E-3
      S = PI2 / TAU
      PUTHPT = COS(ARG) * S * UTHO*1.E-3
      PUPHPT = - SIN(ARG) * S * UPHO*1.E-3
      RETURN
      END
```

```
      SUBROUTINE VVORTEX
      COMMON / MODELS / MODEL(4)
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (RO, W(331)), (THO, W(332)), (PHO, W(333)), (WR, W(33
     A4)), (UO, W(335)), (RE, W(19))
      DATA (MODEL(2) = 6HVVORTX)
      ENTRY WIND
      DTH = R(2) - THO
      DPH = R(3) - PHO
      RAD = RE * SQRT(DTH * DTH + DPH * DPH)
      A = 1.397
      B = - 1.26
      EXB = EXP(B * RAD * * 2 / RO * * 2)
      DUM = A * RE * UO * RO / RAD * * 2*1.E-3
      DUX = (1. - EXB) / RAD + RAD * B * EXB / RO * * 2
      UTH = - DUM * (1. - EXB) * DPH
      UPH = DUM * (1. - EXB) * DTH
      PUPX(2, 2) = 2. * DUM * RE * * 2 * DTH * DPH / RAD * DUX
      PUPX(3, 3) = - PUPX(2, 2)
      PUPX(2, 3) = 2. * DPH * * 2 * DUM * RE * * 2 / RAD * DUX - DUM *
     A(1. - EXB)
      PUPX(3, 2) = - 2. * DTH * * 2 * DUM * RE * * 2 / RAD * DUX + DUM
     A* (1. - EXB)
      RETURN
      END
```

```
      SUBROUTINE ULOGZ
      COMMON / MODELS / MODEL(4)
      COMMON R(6) / WW / ID(10), DUM, W(400)
      COMMON / UU / UR, UTH, UPH, PUPX(3, 3), PURPT, PUTHPT, PUPHPT
      EQUIVALENCE (UO, W(372))
      DATA (MODEL(2) = 6HULOGZ )
      ENTRY WIND
      H = R(1) - W(19)
     IF ( H.LE.0. ) H=0.
      UPH = UO / LOGF(2.) * LOGF(H + 1.)*1.E-3
      PUPX(3, 1) = UO / LOGF(2.) / (H + 1.)*1.E-3
      RETURN
      END
```

APPENDIX B

Atmospheric Models

Brief descriptions of several simple models of atmospheric wind and temperature fields follow. The program requires one subroutine for the temperature field and one for the wind field; even if an isothermal atmosphere with no winds is being modeled, two of these routines (TLNEAR and WCONST, in this case, with appropriate input parameters zero) must be used.

The user may readily devise his own models by writing analytic descriptions of temperature and wind fields, and their spatial derivatives, modeling the programming after the simple examples given here.

Caution must be exercised when using wind fields near the earth's poles; only VVORTEX, of all the wind models given here, has winds that are continuous and have continuous derivatives there. Otherwise, large integration errors (and time-consuming step-size reductions) can be expected.

SUBROUTINE TLNEAR

This subroutine specifies an atmospheric temperature that increases linearly with height.

$$T = T_o + \left(\frac{dT}{dz}\right) z$$

Input to the subroutine are:

(W340) T_o, the ground temperature, ^oK

(W341) $\frac{dT}{dz}$, the temperature gradient, $^oK/km$
(set = 0. for isothermal atmosphere)

SUBROUTINE TTANH +

This subroutine represents the atmospheric temperature profile by a series of segments in which the temperature gradient is linear. These segments are smoothly joined by hyperbolic functions:

$$T = T_o + \frac{c_1}{2} z + \sum_{i=1}^{n} \delta_i \left(\frac{c_{i+1} - c_i}{2} \right) \ell n \left\{ \frac{\cosh\left(\frac{z - z_i}{\delta_i}\right)}{\cosh\left(\frac{z_i}{\delta_i}\right)} \right\}$$

$$\frac{dT}{dz} = \frac{c_1}{2} + \sum_{i=1}^{n} \left(\frac{c_{i+1} - c_i}{2} \right) \tanh \left(\frac{z - z_i}{\delta_i} \right)$$

Thus, δ_i is the half-thickness of a region centered at approximately z_i km, in which $\frac{dT}{dz}$ goes from c_i to c_{i+1}.

The parameters c_i, δ_i, z_i may be chosen to fit virtually any temperature profile.

The U.S. Standard Atmosphere (1962) may be closely represented by letting $n = 4$, $T_o = 288\ ^oK$, and

$c_1 = -6.5$	$z_1 = 15.$	$\delta_1 = 10.$
$c_2 = 3.5$	$z_2 = 52.$	$\delta_2 = 7.5$
$c_3 = -3.0$	$z_3 = 95.$	$\delta_3 = 10.$
$c_4 = 18.0$	$z_4 = 165.$	$\delta_4 = 50.$
$c_5 = 1.0$		

These parameters are presently specified in the subroutine's DATA statement, rather than input through the W array.

+ Adapted from a form used by R. Lindzen.

SUBROUTINE WCONST

This subroutine specifies constant radial (upward), zonal (eastward) and meridional (southward) winds, except that a possible linear height gradient of the zonal component is allowed.

$$u_\theta = U_{\theta o}$$

$$u_\phi = U_{\phi o} + \left(\frac{du_\phi}{dz}\right) z$$

$$u_r = U_{ro}$$

Input to the subroutine are:

(W370) U_{ro}, the constant upward wind, m/s

(W371) $U_{\theta o}$, the constant southward wind, m/s

(W372) $U_{\phi o}$, the ground value of the eastward wind, m/s

(W376) $\frac{du_\phi}{dz}$, the height gradient of u_ϕ, m/s/km

(This subroutine should be used, with its input parameters zero, when no wind field is desired.)

SUBROUTINE ULOGZ

This subroutine represents the wind profile of the atmospheric boundary layer:

$$u_\phi = \frac{U_{\phi o}}{\ell n 2} \ell n(z + 1)$$

Input to the subroutine is:

(W372) $U_{\phi o}$, the value of the eastward wind (m/s) at a height of 1 km.

SUBROUTINE WGAUSS

This subroutine specifies a zonal wind field whose intensity decays in a Gaussian manner in all three space dimensions.

$$u_\phi = U_{\phi o} \exp \left\{ -\left(\frac{z-z_o}{W_z}\right)^2 - \left(\frac{\theta-\theta_o}{W_\theta}\right)^2 - \left(\frac{\phi-\phi_o}{W_\phi}\right)^2 \right\}$$

Input to the subroutine are:

(W381) $U_{\phi o}$, the maximum value of u_ϕ, m/s

(W385) z_o, the height where u_ϕ, maximizes

(W382) W_z, the Gaussian width in height of u_ϕ, km *

(W386) θ_o, the colatitude where u_ϕ maximizes, in radians

(W383) W_θ, the meridional width of u_ϕ, in radians *

(W387) ϕ_o, the longitude where u_ϕ maximizes, in radians

(W384) W_ϕ, the zonal width of u_ϕ, in radians *

* Setting W_z, W_θ or $W_\phi = 0.$

results in no space variation in that direction.

SUBROUTINE WTIDE

This subroutine represents the wind field of the atmospheric tides by zonal and meridional height profiles that are sinusoidal and in phase quadrature. The profiles progress downward with time, giving a "corkscrew" effect:

$$u_\theta = U_{\theta o} \sin \left\{ 2\pi \left(\frac{z}{\lambda_z} + \frac{t}{\tau} \right) \right\}$$

$$u_\phi = U_{\phi o} \cos \left\{ 2\pi \left(\frac{z}{\lambda_z} + \frac{t}{\tau} \right) \right\}$$

Input to the subroutine are:

(W391) $U_{\theta o}$, the amplitude of the meridional component, m/s

(W390) $U_{\phi o}$, the amplitude of the zonal component, m/s

(W392) λ_z, the vertical wavelength in km

(W393) t/τ, the time in wave periods

(W394) τ, the wave period in seconds

(The earth's poles should be avoided in ray calculations, as discontinuities appear there.)

SUBROUTINE VVORTEX

This subroutine models a cylindrical vortex with a viscous core. The axis of the vortex is vertical and may be positioned above any geographic latitude and longitude. The vortex rotates anticlockwise looking down. The "core" (inside r_o) is essentially a solid-rotating fluid, while outside r_o, $|u|$ falls off as the inverse radius.

$$u_\theta = \frac{1.397\, R_e U_o r_o}{r^2} \left(1 - e^{-1.26\, r^2/r_o^2}\right) \left(\phi - \phi_o\right)$$

$$u_\phi = \frac{1.397\, R_e U_o r_o}{r^2} \left(1 - e^{-1.26\, r^2/r_o^2}\right) \left(\theta - \theta_o\right) ,$$

where r is the radial distance from the vortex center. The numerical constants arise from a transcendental equation in the development of the vortex field expression.

Input to the subroutine are:

(W335) U_o, the maximum tangential wind, m/s

(W331) r_o, the radius of the vortex core (to $u = U_o$), km

(W332) θ_o, the colatitude of the vortex center, radians

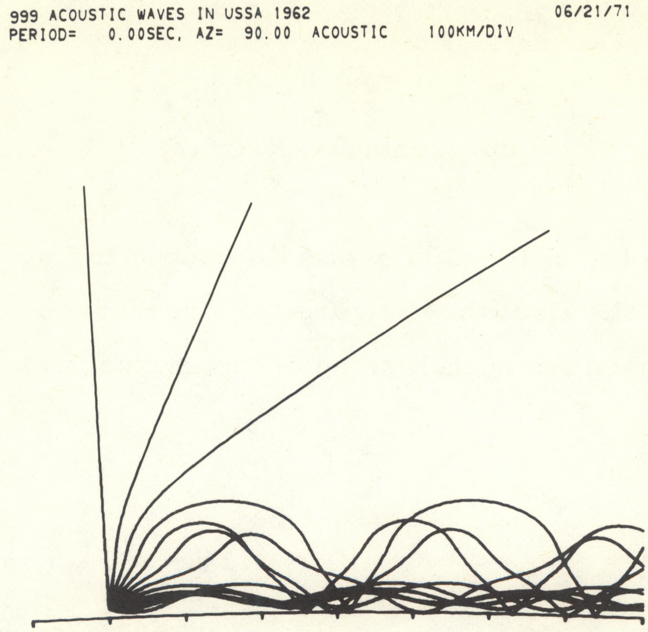
(W333) ϕ_o, the longitude of the vortex center, radians

(W19) R_e, the earth's radius, km

APPENDIX C

Representative Rayplots

A few examples of the CRT plotted output of the ray-tracing program follow. The first three figures are reproduced directly from microfilm and consequently lack some of the neatness of a drafted figure.



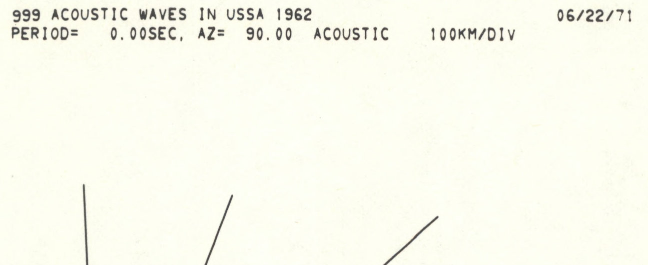


Figure 2. Raypaths of acoustic waves with a frequency of about 300 Hz in an atmosphere with no winds and a temperature profile resembling USSA 1962. At the top, the source is at an altitude of 13 km; at the bottom, the source is at the ground. Scale is 100 km/div.; increment in launch angle is 10°. All rays lie in the plane of the page.

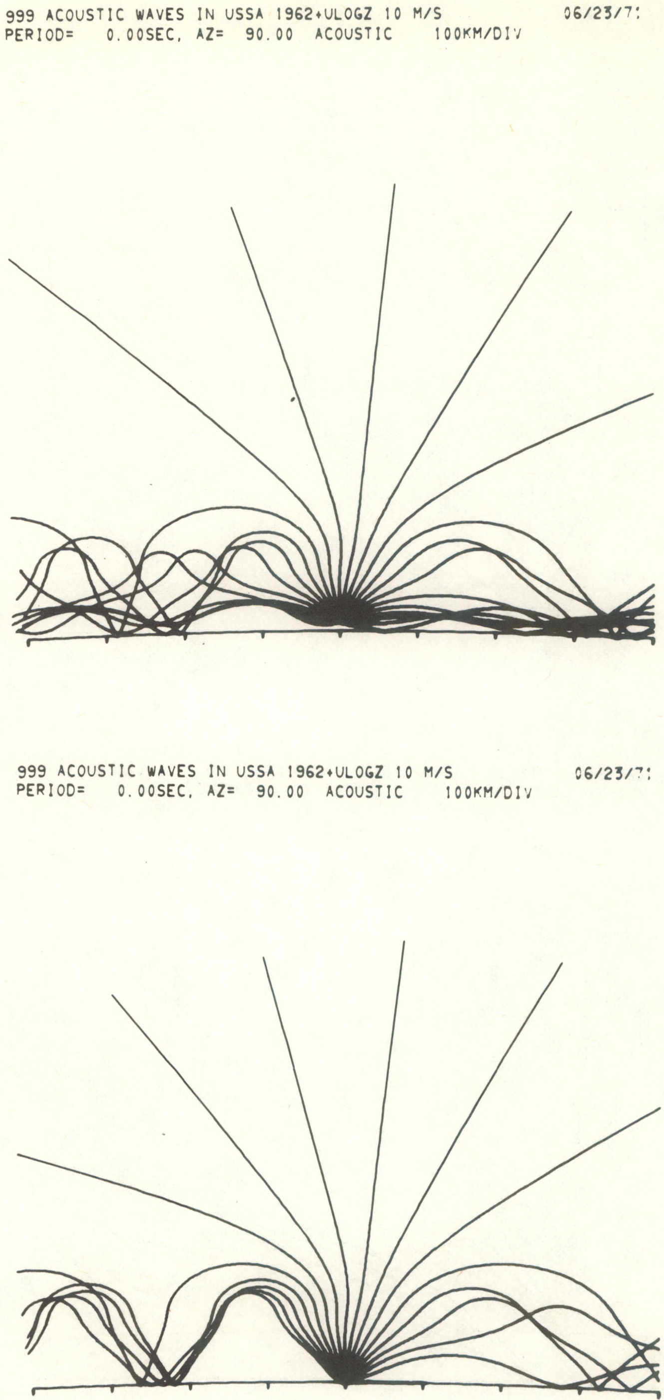


Figure 3. Raypaths of acoustic waves with a frequency of about 300 Hz in an atmosphere with a temperature profile resembling USSA 1962 and a wind logarithmically increasing with height and blowing to the right. The wind at a height of 1 km is 10 m/s and at 100 km is 66 m/s. Scale: 100 km/div; increment in elevation angle: 10°.

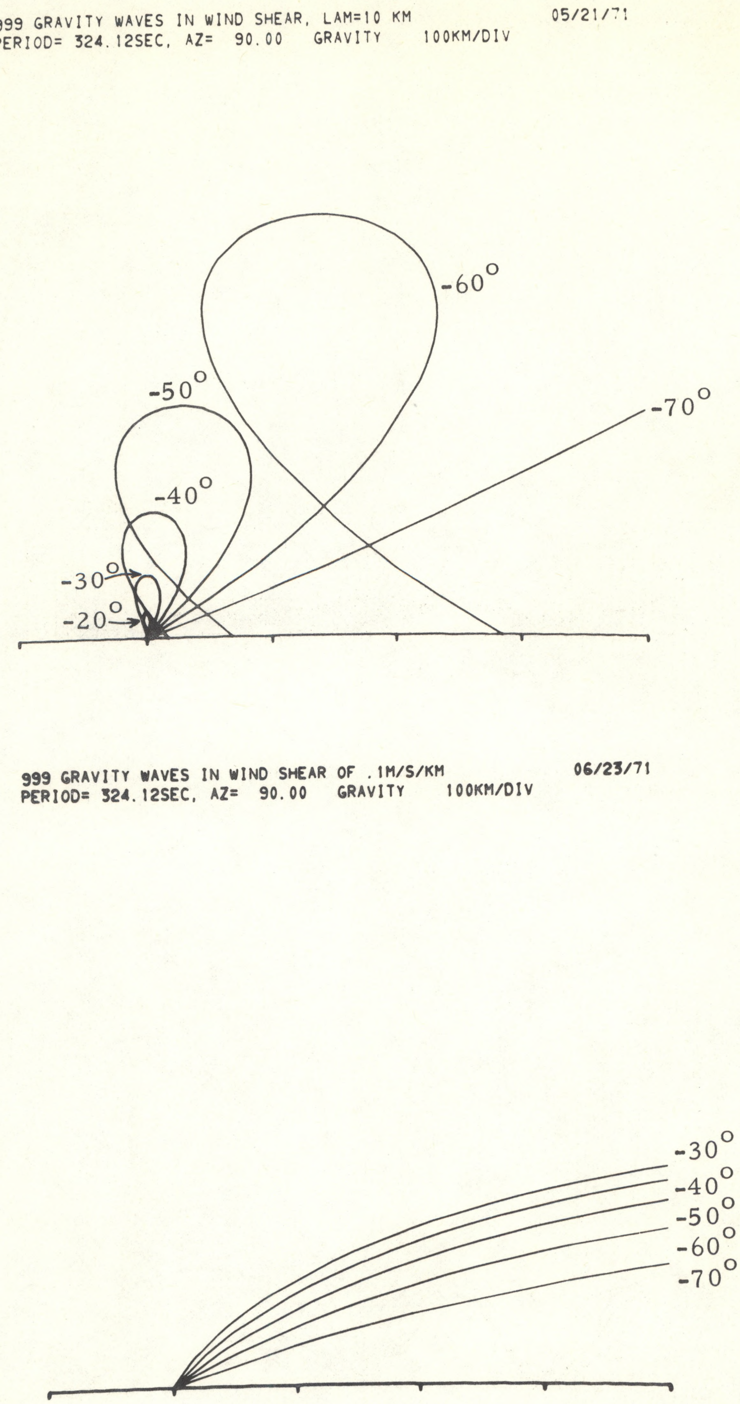


Figure 4. Raypaths for internal gravity waves, all with a total wavelength of 10 km, as the initial direction of phase propagation (labeled on the curves) is varied. The atmosphere is isothermal and wind increases linearly with height by 0.1 m/s/km, blowing to the left in the top figure, and to the right in the bottom. Scale is 100 km/div.

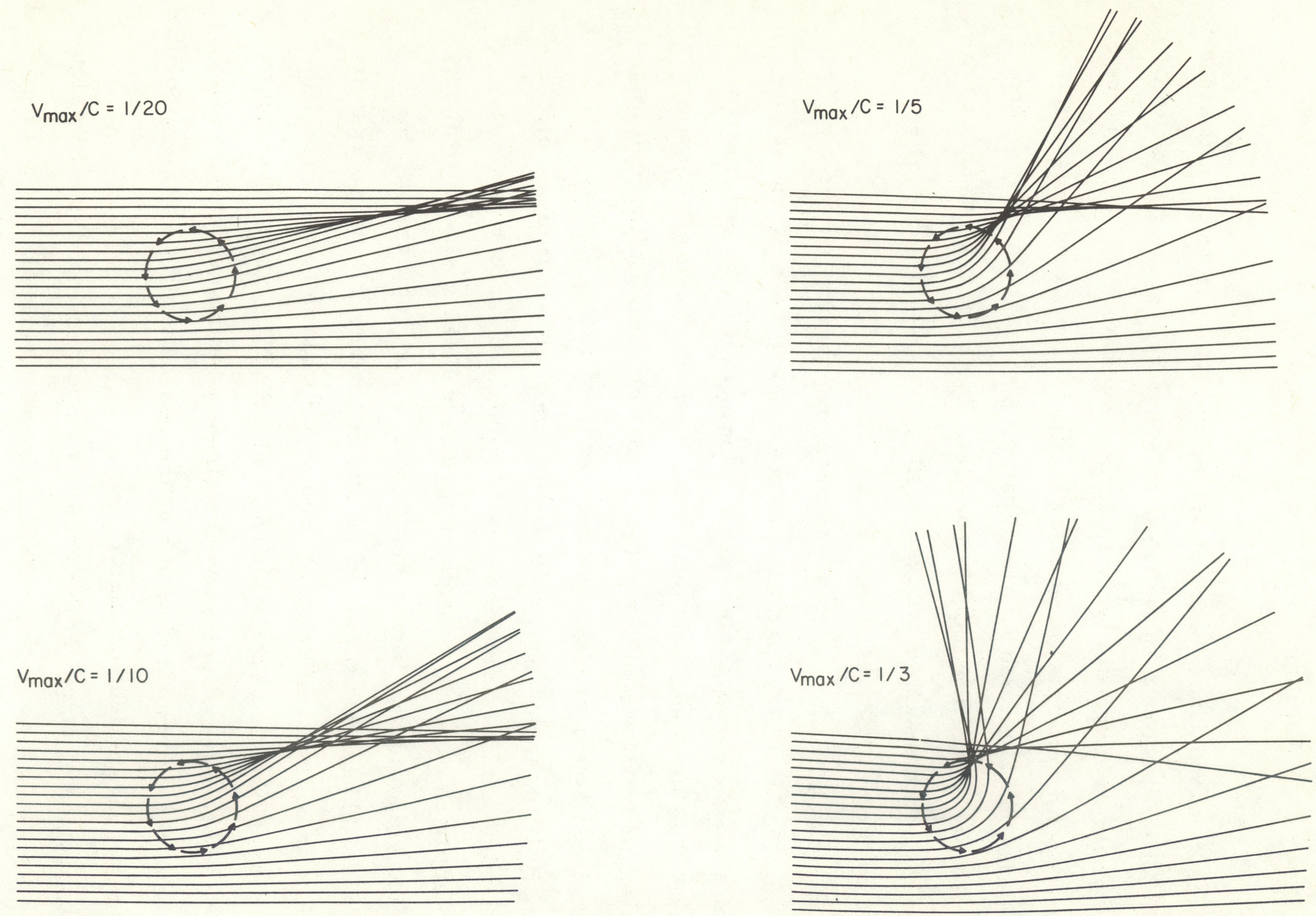


Figure 5. Raypaths of acoustic rays with a frequency of about 1 kHz through an isothermal wind field of a viscous cylindrical vortex. The dashed circle shows where the velocity reaches the maximum value labeled on each plot. Inside the circle there is nearly rigid-body rotation, while outside, tangential velocity decays as the inverse radius.

APPENDIX D

Application to Ocean Acoustics

Although subroutines have not yet been developed to permit direct application of the program to underwater sound propagation, such an extension would involve relatively little reprogramming. The effects of gravity and ocean currents (the analog of winds) on acoustic propagation in the sea are clearly much smaller than on atmospheric waves, and appear to be negligible for most purposes. The effects of earth curvature are also much smaller because of the small depth of the sea compared to the earth's radius.

Thus, much of the power of the present program is "wasted" on ocean acoustics, the 3D capability probably being the only asset to represent a distinct improvement over other existing ray-tracing programs. The three-dimensional field of sound speed has apparently not been studied extensively, and the effects of realistic three-dimensional fields on sound propagation has received little, if any, analytical attention. It might thus be profitable to examine propagation effects of simple models of certain three-dimensional ocean features.

To apply to ocean acoustics, subroutine HASEL should probably be modified, eliminating terms in ω_a and ω_g, and replacing all the Ω's with ω's. (These terms could be left in, however, just to examine the magnitude of their influence.) The statements defining ω and Ω in NITIAL should likewise be simplified. In TRACE, it is necessary to provide for specular reflection at the air-sea interface, at some (input) height above the ground, reflection at the sea bottom is already provided for by the ground-reflection logic.

Absorption of sound in the ocean is important, and can be calculated by adding an absorption differential equation in HASEL. Empirical formulas are presently used, and a recent one [Urick, 1967] is

$$\alpha = \frac{0.1 f^2}{1 + f^2} + \frac{40 f^2}{4100 + f^2} + 5 \times 10^{-4} f^2$$

in "decibels per kiloyard", where f is the wave frequency in kHz.

Finally, it is necessary to write new subroutines to describe various three-dimensional fields of sound speed.

Much more substantial reprogramming would be required to account for the non-planar and imperfect reflecting properties of the sea bottom and the sea surface, and is not presently contemplated.

GPO 837-855