

AD-A143 079

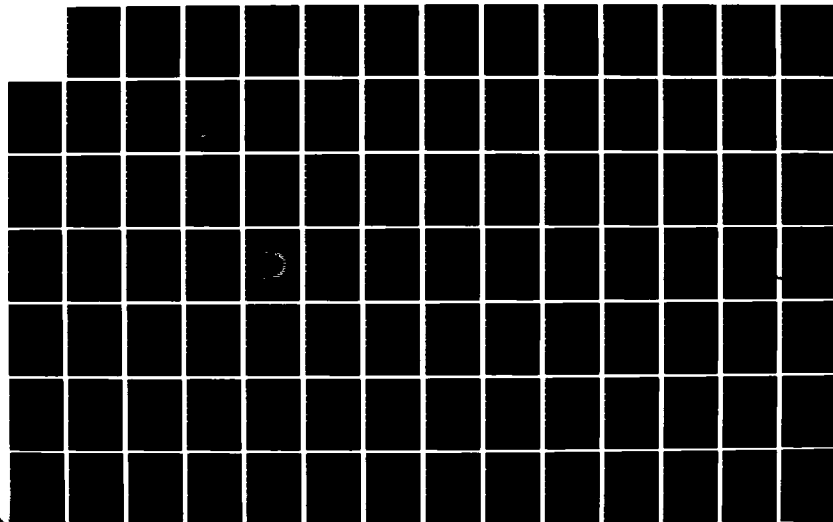
A COMPARISON OF TWO ACOUSTIC PARABOLIC TRANSMISSION
LOSS MODELS FOR COMP. (U) NAVAL POSTGRADUATE SCHOOL
MONTEREY CA J L BLANCHARD MAR 84

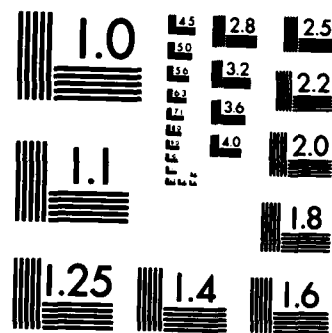
1/2

UNCLASSIFIED

F/G 20/1

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NAVAL POSTGRADUATE SCHOOL
Monterey, California

AD-A143 079



THESIS

JUL 17 1984

A COMPARISON OF TWO ACOUSTIC PARABOLIC EQUATION
TRANSMISSION LOSS MODELS FOR COMPATIBILITY
WITH THE WAVENUMBER TECHNIQUE IN THE
DETERMINATION OF SOURCE DEPTH

by

Joe Lane Blanchard II

March 1984

Thesis Advisor:

A. B. Coppens

Approved for public release; distribution unlimited

DTIC FILE COPY

84 07 16 061

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO. ADH-3076	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Comparison of two Acoustic Parabolic Transmission Loss Models for Compatibility with the Wavenumber Technique in the Determination of Source Depth		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis March 1984
7. AUTHOR(s) Joe Lane Blanchard II		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		8. CONTRACT OR GRANT NUMBER(s) PE 63785N, NORDA
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93943		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS N6846283WR30089 of 29 APRIL 1983
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE March 1984
		13. NUMBER OF PAGES 151
		15. SECURITY CLASS. (of this report)
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for Public Release, Distribution Unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Parabolic Wave Equation, Underwater Acoustics, Implicit Finite-Difference, Split-Step Fast Fourier Transform, Wavenumber Technique, Source Depth Determination, Acoustic Propagation, ASW, Lloyd Mirror		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The Brock version of the Split-Step Fast Fourier Transform (SSFFT) and the Jeager version of the Implicit Finite Difference (IFD) acoustic parabolic equation models are compared with a Lloyd mirror interference pattern in the range domain. The SSFFT displays the inability to place the transmission loss nulls at the correct ranges. It is also unable to utilize bottom loss information correctly. The		

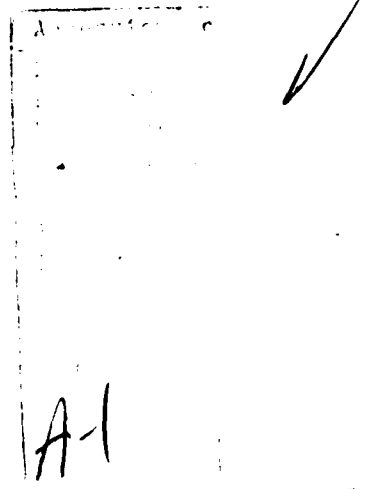
DD FORM 1 JAN 73 1473

EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

IFD produced nulls at the correct ranges; however, it inserted an unacceptable amount of noise except when small (1 m) vertical grid steps were used and the pressure release bottom was placed at extended depths. In shallow water cases, the IFD is able to properly represent the pressure information. Each model is explored in the wavenumber domain by use of a "Wavenumber Technique" (WT) model with emphasis on source depth determination. The source depth may be determined by measuring the distance between the equally spaced nulls in the wavenumber representation. Neither acoustic model was able to provide accurate source depth information when the null spacings were compared to a known source-depth determination curve. Since the null spacings were not uniformly spaced, this was to be expected. Some specific problem areas in the models were identified by the use of the WT.



Approved for public release; distribution unlimited.

**A Comparison of two Acoustic Parabolic Equation
Transmission Loss Models for Compatibility with the
Wavenumber Technique in the Determination of Source Depth**

by

Joe L. Blanchard II
Lieutenant, United States Navy
B.S., University of North Carolina at Charlotte, 1974

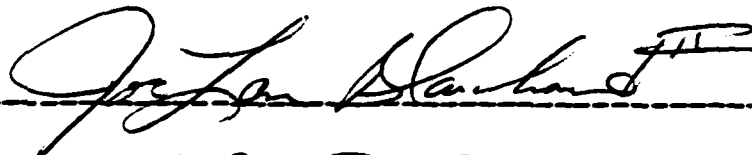
Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN METEOROLOGY AND OCEANOGRAPHY

from the

NAVAL POSTGRADUATE SCHOOL
March 1984

Author:



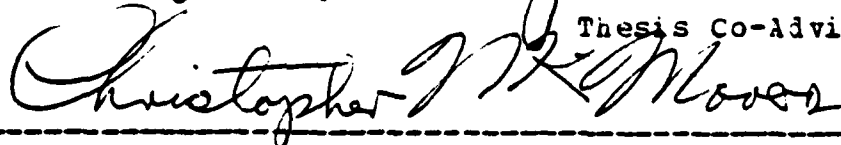
Approved by:



Thesis Advisor



Thesis Co-Advisor



Chairman, Department of Oceanography



Dean of Science and Engineering

ABSTRACT

The Brock version of the Split-Step Fast Fourier Transform (SSFFT) and the Jeager version of the Implicit Finite Difference (IFD) acoustic parabolic equation models are compared with a Lloyd mirror interference pattern in the range domain. The SSFFT displays the inability to place the transmission loss nulls at the correct ranges. It is also unable to utilize bottom loss information correctly. The IFD produced nulls at the correct ranges; however, it inserted an unacceptable amount of noise except when small (1 m) vertical grid steps were used and the pressure release bottom was placed at extended depths. In shallow water cases, the IFD is able to properly represent the pressure information. Each model is explored in the wavenumber domain by use of a "Wavenumber Technique" (WT) model with emphasis on source depth determination. The source depth may be determined by measuring the distance between the equally spaced nulls in the wavenumber representation. Neither acoustic model was able to provide accurate source depth information when the null spacings were compared to a known source-depth determination curve. Since the null spacings were not uniformly spaced, this was to be expected. Some specific problem areas in the models were identified by the use of the WT.

TABLE OF CONTENTS

I.	INTRODUCTION	11
II.	WAVENUMBER TECHNIQUE (WT)	13
	A. BACKGROUND (LLOYD MIRROR)	13
	B. PHYSICAL PROCESS	16
III.	ACOUSTIC MODEL ANALYSIS	22
	A. PARABOLIC EQUATION	22
	B. SSFFT	23
	1. Transmission Loss Comparison with Lloyd Mirror	23
	2. WT Analysis	28
	C. IFD	32
	1. Transmission Loss Comparison with Lloyd Mirror	32
	2. WT Analysis	39
IV.	CONCLUSIONS	45
	APPENDIX A: SSFFT SOURCE CODE (AT NPS)	47
	APPENDIX B: WT SOURCE CODE	102
	APPENDIX C: WT GRAPHICS SOURCE CODE	107
	APPENDIX D: IFD SOURCE CODE (AT NPS)	112
	LIST OF REFERENCES	147
	BIBLIOGRAPHY	149
	INITIAL DISTRIBUTION LIST	150

LIST OF FIGURES

2.1	Lloyd Mirror Geometry	13
2.2	Theoretical Lloyd Mirror Transmission Loss . . .	15
2.3	Quadrature Demodulation of a Signal	16
2.4	Example of WT Output at 50 Hz (From Lauer, 1979)	20
2.5	Source Depth Determination Curve	21
3.1	Test of SSFFT Model for Null Spacings	25
3.2	Test of SSFFT Model for Bottom Interaction . . .	27
3.3	SSFFT WT Plot with Source at 500 meters	29
3.4	SSFFT WT Plot with Receiver at 1000 meters . . .	30
3.5	SSFFT WT Plot with Source at 800 meters	31
3.6	IFD Environment	33
3.7	IFD Transmission Loss	34
3.8	IFD Detailed Transmission Loss	35
3.9	IFD Noise Free Transmission Loss	36
3.10	IFD: Shallow Water Transmission Loss	37
3.11	IFD: 500 meter Source, 15 meter Vertical Grid Step	40
3.12	IFD: 500 meter Source, 3.75 meter Vertical Grid Step	41
3.13	IFD: 500 meter Source, 0.8 meter Vertical Grid Step	42
3.14	IFD: Shallow Water WT, 1 meter Vertical Grid Step	44

Acronyms

AESD Acoustic Environmental Support Detachment
 of the Office of Naval Research, now
 the Numerical Modeling Division
 (Code 320) of the Naval Ocean Research
 and Development Activity (NORDA).

ASTREX Acoustic Storm Transfer and Response
 Experiment, conducted by the Naval
 Postgraduate School (NPS) in the
 northeastern Pacific during November and
 December of 1980.

NORDA Naval Ocean Research and Development
 Activity at Bay St. Louis, Mississippi

NPS Naval Postgraduate School at Monterey,
 California

NUSC Naval Underwater Systems Center at New
 London, Connecticut

Symbols

d	Receiver Depth
f	Frequency
h	Source Depth
j	Square Root of -1
n	Index of Refraction (C_0/C)
m	Index in Calculations or Null Number in the Range Domain
p	Time Independent Factor of Complex Pressure
r	Distance of the Direct Path Wave
r	Distance to the Image (Reflected Path)
z	Source Depth
A	Amplitude
C	Sound Speed ($C(r,z)$)
C_0	Reference Sound Speed (minimum sound speed in water mass profile)
$F(K)$	Pressure Field in the Wavenumber Space
FFT	Fast Fourier Transform
FFT^{-1}	Inverse Fast Fourier Transform
H	Hankel function
K	Wavenumber
K_0	Reference Wavenumber (ω/C_0)
K_r	Horizontal Wavenumber

NPT	Number of Points in the Wavenumber Spectrum
P	Complex Pressure
R	Range between Source and Receiver
ΔR	Range Increment
$U(r, z)$	Envelope function
U_i	Envelope function (Imaginary part)
U_r	Envelope function (Real part)
z_r	Receiver Depth
z_s	Source Depth
β	Scaled Wavenumber
$\Delta\beta$	Scaled Wavenumber Increment
ω	Angular Frequency
∂	Partial Derivative Operator
π	3.1415...
∇	Laplacian Operator
$\sqrt{}$	Square Root Operator

ACKNOWLEDGEMENT

I would like to express my deepest appreciation to Dr. A. B. Coppens for his guidance in analytical processes and fundamentals of acoustics. His understanding and encouragement provided the sound basis for this research effort. Dr. Coppens' knowledge of the subject matter combined with the ability to communicate this knowledge from many different perspectives is a unique talent recognized by his students.

LCDR C. R. Dunlap, USN (Ret) was very supportive and encouraging during my entire thesis research effort. I appreciate his assistance with this project and his continuing work with the Environmental Acoustics Research Group.

I would like to thank the personnel of the W.R. Church Computer Center and especially Mr. D. Marr and Mr. J. Norton.

I. INTRODUCTION

Conventional methods used to represent underwater sound transmission have depicted transmission loss (dB) as a function of range in either tabular or graphical form. Since the interest was in determining quantities such as range of bottom bounce, convergence zones, and probability of detection, these methods proved very satisfactory. However when it became necessary to compare various underwater acoustic propagation loss models with each other or actual in-situ experiments, they provided the analyst with less than adequate insight into the causes for the observed errors in the various models outputs. Furthermore, interest has been generated in analyzing the received signal for information from the sound source which cannot easily be induced from transmission loss as a function of range.

One method, the Wavenumber Technique (WT), described by F. R. DiNapoli of NUSC [Ref. 1] and applied by Richard Lauer of NORDA [Ref. 2], provides more information for model comparisons and may have the capability of determining sound source depth and range from the receiver in certain cases. DiNapoli used an analysis of acoustic propagation in the wavenumber domain as an intermediate step in getting the transmission loss curve in the Fast Field Program (FFP). Lauer studied the wavenumber domain information from the FFP and concluded that source localization might be possible. B. B. Stamey [Ref. 3] investigated the potential for determining the source depth by using the Brock Split-Step Fast Fourier Transform (SSFFT) [Ref. 4] parabolic equation model with the "Wavenumber Technique" (WT) model. His preliminary investigation involved the use of isospeed and ASTREX sound speed profiles with a fully absorbing bottom, varied source/receiver combinations, and multiple frequencies.

The present investigation uses two acoustic parabolic equation models for comparison with theoretical Lloyd mirror depictions, and the effects of observed inconsistencies in predicted transmission loss curves on the calculated results from application of the WT model. An analysis of the sensitivity in determining source depth based on model output is attempted to ascertain possible use in naval operations.

To facilitate a straight-forward analysis of the results and evaluation of the WT technique for the cases studied, only a selected number of the various computer runs are included. Only those curves which best illustrate the effects revealed by this research are presented.

II. WAVENUMBER TECHNIQUE (WT)

A. BACKGROUND (LLOYD MIRROR)

The WT can be elucidated with the help of the classical Lloyd mirror effect [Ref. 5] which describes the interaction between direct path and surface reflected sound signals from the same continuous wave (CW) source. Figure 2.1 illustrates the geometry involved in the Lloyd mirror effect.

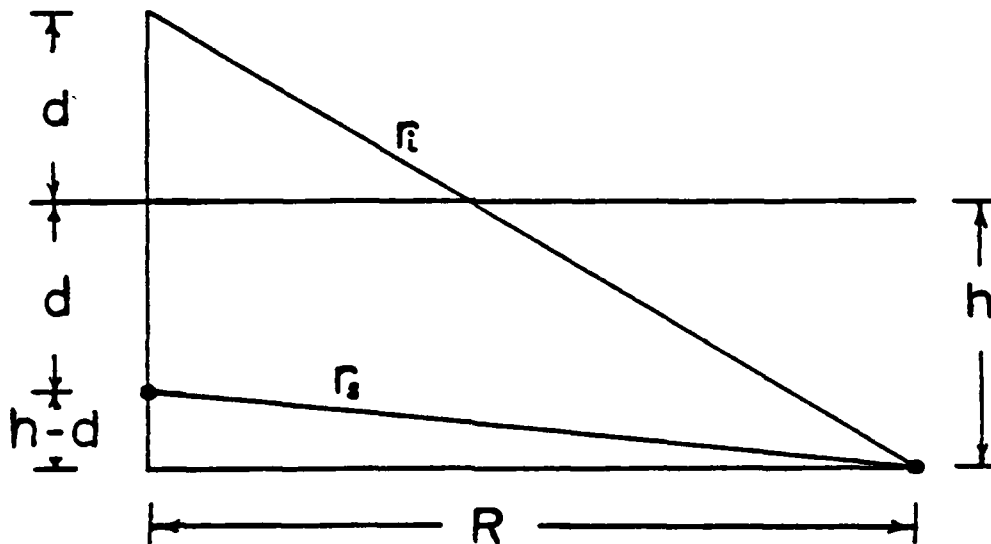


Figure 2.1 Lloyd Mirror Geometry.

The complex pressure (P) can be expressed by

$$P = A \left[\frac{1}{r_i} e^{-jk r_i} - \frac{1}{r_s} e^{-jk r_s} \right] (e^{+j\omega t}) \quad (\text{eqn 2.1})$$

when r_i and r_s are related by

$$r_i = \sqrt{R^2 + (d-h)^2} \quad r_s = \sqrt{R^2 + (d+h)^2} \quad (\text{eqn 2.2})$$

with the assumption $r_i \sim r_s$ the pressure amplitude (p) can be approximated by

$$R_m = \frac{2fhd}{mC}, m=1,2,3,\dots \quad (\text{eqn 2.3})$$

The interaction between the direct path and surface reflected waves produces constructive and destructive interference which is manifested as peaks and nulls for the in phase and out of phase conditions respectively. Figure 2.2 was produced by using equation 2.3. The null locations can be obtained by the relationship.

$$p = \frac{2A}{R} \sin\left(\frac{kh d}{R}\right) \quad (\text{eqn 2.4})$$

When equation 2.3 is expressed in the horizontal wavenumber domain (k_r), the details of the resultant functional dependency on horizontal wavenumber yield useful information.

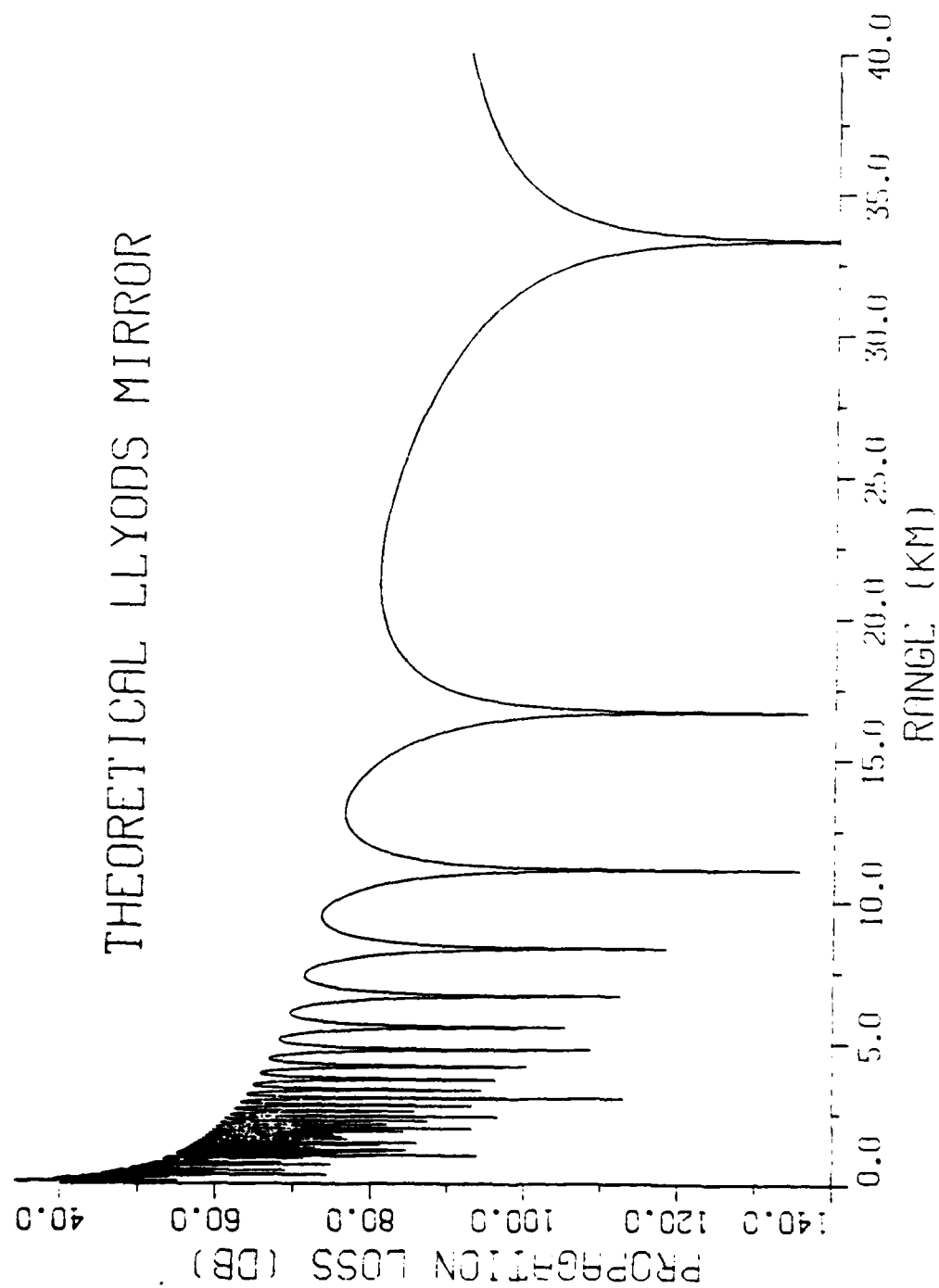


Figure 2.2 Theoretical Lloyd Mirror Transmission Loss.

B. PHYSICAL PROCESS

The WT is a process by which the complex pressure wave (with range dependent amplitude and phase) is transformed into the spectral density as a function of the horizontal wavenumber. The use of the WT in the operational environment would require a quadrature demodulation of the source signal to attain the complex pressure. The quadrature demodulation process can best be defined by the illustration in figure 2.3 [Ref. 6].

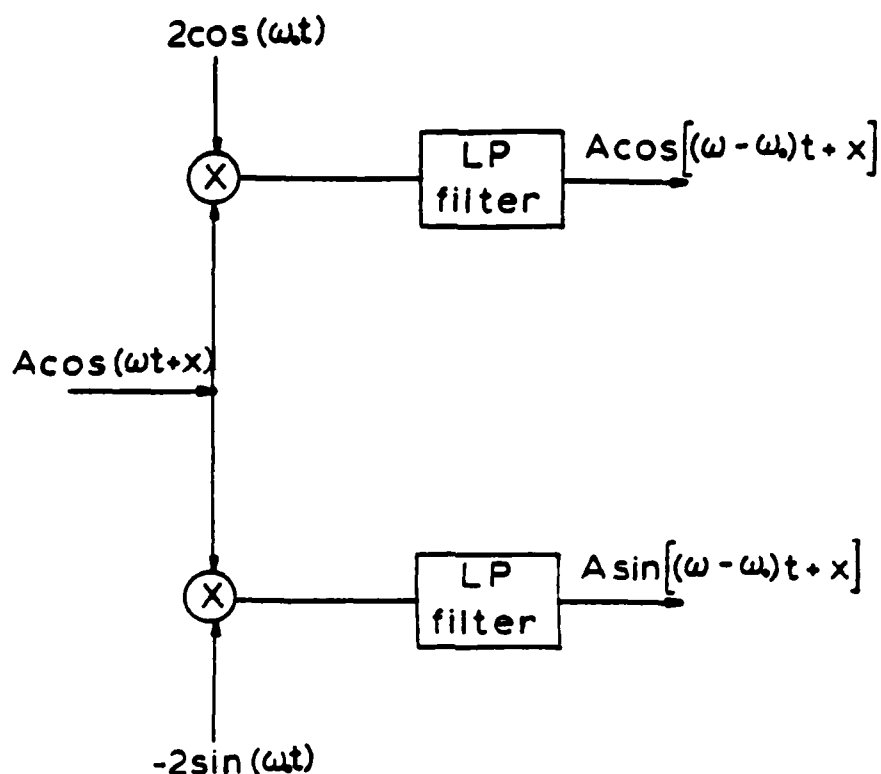


Figure 2.3 Quadrature Demodulation of a Signal.

When acoustic models are used to simulate the environment, the requirement to perform a quadrature demodulation is eliminated since the complex pressure is directly

accessible. In actuality, acoustic parabolic equation models provide the complex envelope function (U) which must be multiplied by the Hankel function in order to obtain the complex pressure. The acoustic wave is written in the form

$$P = U H_0^{(1)}(k r) e^{-j\omega t} \quad (\text{eqn 2.5})$$

where the reference wavenumber is defined by

$$k_0 = \frac{\omega}{c_0} \quad (\text{eqn 2.6})$$

and

$$U = U(r, z) \quad (\text{eqn 2.7})$$

is the solution to the appropriate parabolic equation. The complex pressure is corrected for volume attenuation and Fourier transformed to attain the spectral density. A plot displaying the results graphically with the horizontal wavenumber on the x-axis and the normalized spectral density on the y-axis is constructed and the spacings between the nulls measured. Figure 2.4 [from Ref. 2] is an example of the WT for a source and receiver in the same type of water mass.

Since the Lloyd mirror field in the wavenumber domain is given by [Ref. 2]

$$F(k) = \frac{\sin(\beta Z_s)}{\beta} e^{j(\beta Z_r)} \quad (\text{eqn 2.8})$$

an alternative formulation [Ref. 2], which produces evenly spaced nulls, is obtained by converting the horizontal wavenumber to beta

$$\beta = \sqrt{k_0^2 - k_r^2} \quad (\text{eqn 2.9})$$

By using beta instead of the horizontal wavenumber, the distance, delta beta, between any two nulls can be used to ascertain the source depth by

$$\Delta\beta = \frac{\pi}{Z_s} \quad (\text{eqn 2.10})$$

as illustrated in figure 2.5.

The application for implying the method of images, in isospeed cases, to describe the effects of a perfectly flat pressure release boundary can be justified by inspection. Therefore the parabolic equation can be used to produce the Lloyd's mirror effect and surface decoupling is not an issue, cf. Chapter 4 of Brekhovskikh [Ref. 7].

When the WT is implemented, care must be exercised to ensure that the complex pressure contains an adequate number of points to describe the shortest periodicity in the pressure field. This will preclude the possibility of aliasing in the wavenumber domain. The real and imaginary elements of the pressure signal must be modified so that the signal begins and ends with zero values. This modification is necessary because the pressure signal must represent a repetitive oscillation for the transform. The complex pressure array is zero-filled beyond the data in order to generate an array which has the length of a power of 2 for the Fast Fourier Transform (FFT). The horizontal wavenumber increment is generated within the computer code by

$$k_{rm} = k_o + \frac{2\pi}{\Delta r} \left(\frac{m}{NPT} - 1 \right) \quad (\text{eqn } 2.11)$$

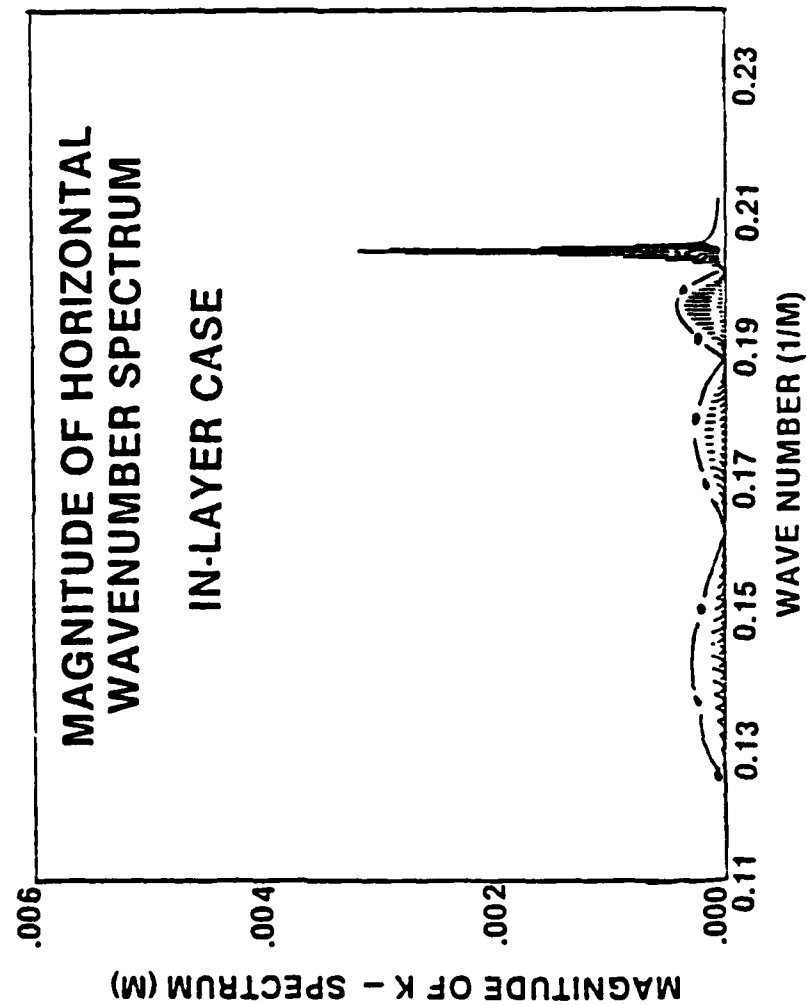


Figure 2.4 Example of WT Output at 50 Hz (From Lauer, 1979).

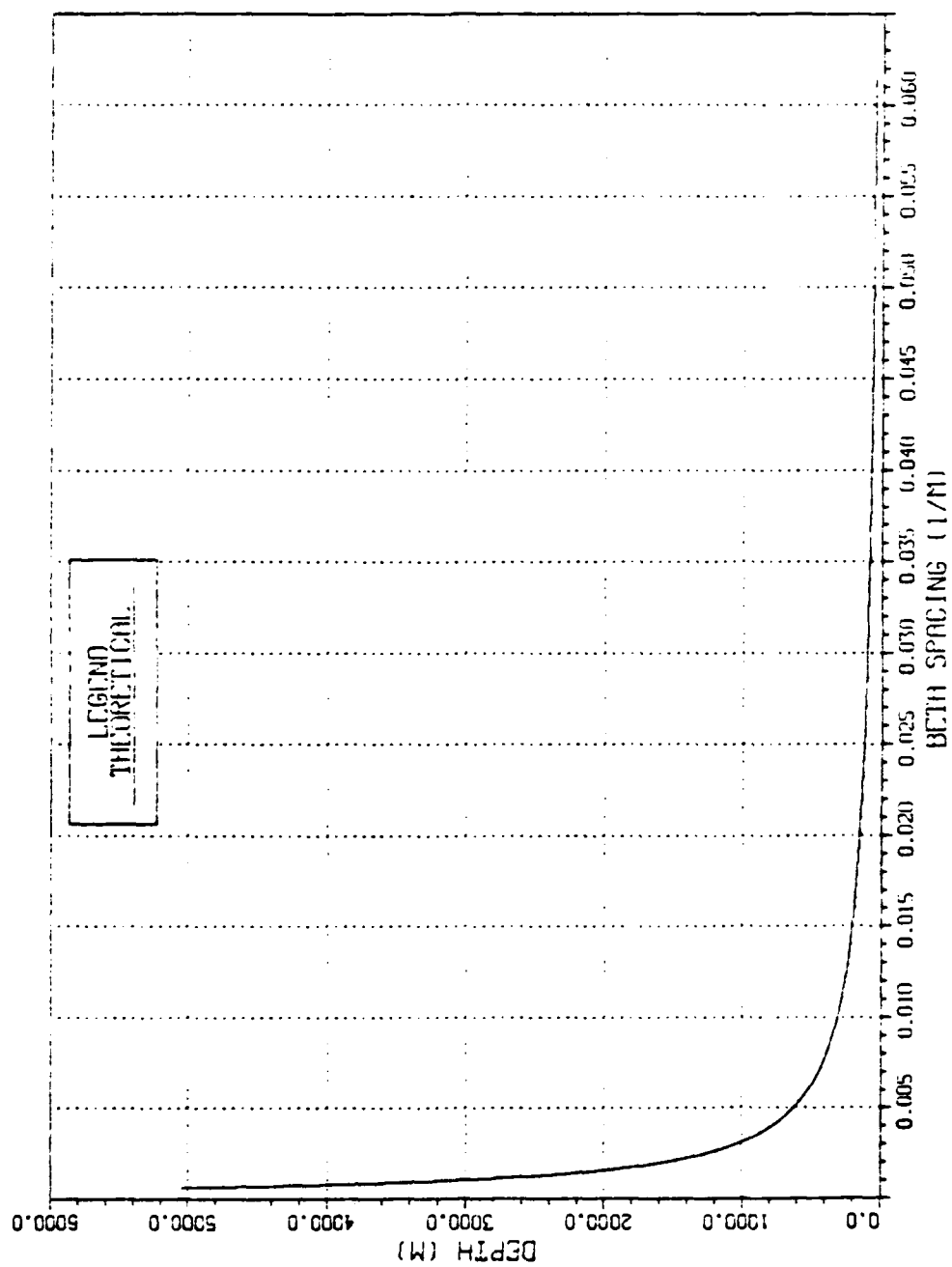


Figure 2.5 Source Depth Determination Curve.

III. ACOUSTIC MODEL ANALYSIS

A. PARABOLIC EQUATION

The elliptical wave equation can be approximated by a parabolic equation (PE) when it is assumed that the envelope function varies slowly with range. A detailed mathematical description of the PE from an acoustic point of view is presented by DeSanto [Ref. 8 and 9] and a simpler but less general description can be found in Coppens [Ref. 10]. The parabolic equation has the form

$$\frac{\partial^2 U}{\partial z^2} + 2jk_0 \left(\frac{\partial U}{\partial r} \right) + (k^2 - k_0^2)U = 0 \quad (\text{eqn 3.1})$$

where it is assumed that the pressure has the form of equation 3.2,

$$p = U(r, z) S(r) \quad (\text{eqn 3.2})$$

$S(r)$ represents the primary radial dependence of the field in terms of an outward propagating cylindrical wave of the form [Ref. 4]

$$S(r) = H_0^{(1)}(k_0 r) \quad (\text{eqn 3.3})$$

If it is assumed that the range of interest is many wavelengths from the source; then the asymptotic form of the Hankel function (equation 3.4) can be used [Ref. 11].

$$H_0^{(1)} = \sqrt{\frac{2}{\pi k_0 r}} e^{j(k_0 r - \pi/4)}, \quad k_0 r \gg 1 \quad (\text{eqn 3.4})$$

Two PE models, the Brock version of the Split-Step Fast Fourier Transform (SSFPT) and the Jeager version of the Implicit Finite Difference (IFD) were investigated for compatibility with the WT in determination of source depth.

B. SSFPT

1. Transmission Loss Comparison with Lloyd Mirror

To analyze sound waves in the wavenumber domain, we needed a model to provide the pressure as a function of range. Initially the SSFPT model [Ref. 4] was selected. This model generates successive values for U as a function of range with the help of the algorithm

$$U(r+\Delta r, z) = e^{j\Delta r k_0(n^2-1)/2}$$

(eqn 3.5)

$$\text{FFT}^{-1} \left\{ e^{j\Delta r k^2/2k_0} \text{FFT}(U(r, z)) \right\}$$

This model was a natural choice since it was the model used in the preliminary study dealing with the determination of the source depth by the WT [Ref. 3]. However, it soon became apparent that this model's inherent weaknesses would require the consideration of another model if ocean bottom interactions were to be studied. These weaknesses, and their impact upon the WT, will be discussed shortly.

The SSFPT is a range dependent acoustic wave model which, for this analysis, will be operated in a range independent manner. In other words, only one sound speed profile will be used, the water mass will be assumed homogeneous, and the bottom flat. To facilitate the study of the SSFPT model, the source code of the model was modified to generate the variables required for follow-on programs to

transform the pressure information and then display the results graphically. A copy of the source code listing for each program is provided in appendices A, B, and C. These programs were then linked by Job Control Language (JCL) so that the WT graphics were automatically executed.

A test run of the SSFFT was made for 100 Hz, 500 m source and receiver depths, 0.019 km range step, fully absorbing bottom, 2000 m water depth, and an isospeed profile at 1500 m/sec. These values were chosen so that a comparison could be made with the theoretical Lloyd mirror transmission loss (figure 2.2). The output was displayed as a transmission loss curve (figure 3.1). From equation 2.4, the location of null number 1 (R1) should be 33.33 km and null number 2 (R2) should be at 16.67 km. The SSFFT placed R1 at approximately 38.3 km and R2 at 17.1 km. Since the model kept the frequency, source depth, and receiver depth constant, then this equates to a reference sound speed of 1305 and 1462 m/sec respectively. It is apparent that as the range decreases the error is reduced which indicates the possibility of a problem in the range step section of the model which leads to the introduction of a systematic and increasing deviation of the calculated sound field from that predicted by the classical Lloyd's mirror interference pattern. The "washing out" of the interference pattern for ranges less than about 2 km is a result of the particular approximation of a point source used to initiate the program, which is to be expected. Rather regular fluctuations develop beyond about 30 km; from the geometry of the case, it is plausible that this arises from interference with bottom reflected signals.

SOURCE DEPTH 500.00 M, RECEIVER DEPTH 500.02 M, FREQUENCY 100.00 HZ
 WATER DEPTH 2000.00 M, RANGE INCR. 0.019 KM, ATTENUATION COEF 7.771×10^{-6} DB/M
 HALF BEAM WIDTH 20.000 DEG, REFERENCE SOUND SPEED 1499.980 M/SEC

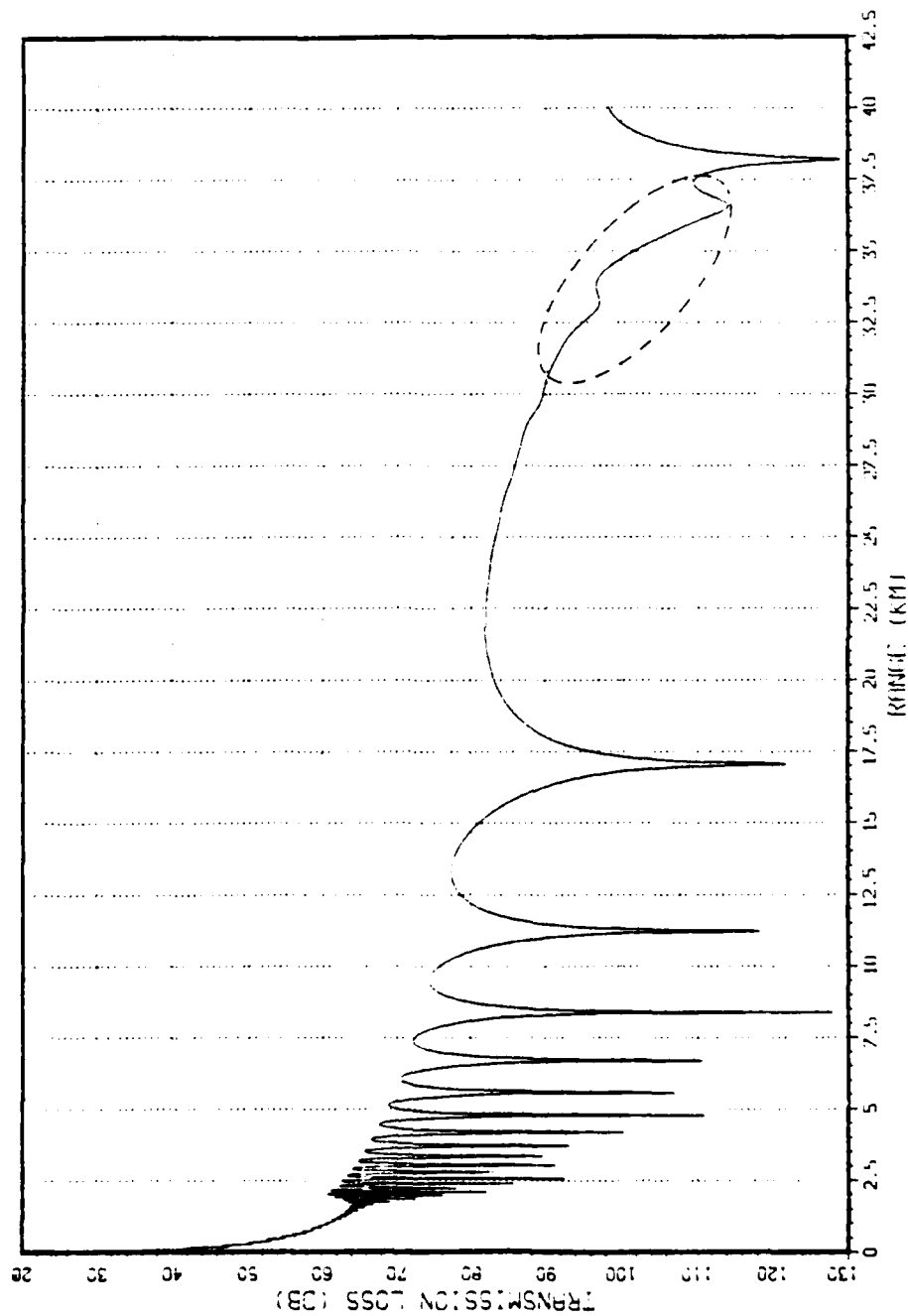


Figure 3.1 Test of SSFFT Model for Null Spacings.

Another test was made using the same inputs as above except that the source and receiver were moved to a depth of 1500 m (figure 3.2). Since the half beamwidth is 20 degrees and the bottom is fully absorbing, the shortest range at which a surface reflection could occur is approximately 8.2 km; beyond this range, surface reflections should be spaced in accordance with equation 2.4. The nulls beyond 8.2 km do occur at the correct ranges; however, there are slow modulations of the overall transmission loss signal. These slow modulations are another indication that the bottom is not fully absorbing as had been specified as input to the program.

SOURCE DEPTH 1500.00 M, RECEIVER DEPTH 1500.18 M, FREQUENCY 100.00 HZ
 WATER DEPTH 2000.00 M, RANGE INCR. 0.019 KM, ATTENUATION COEF 7.771×10^{-6} DB/M
 HALF BEAM WIDTH 20.000 DEG, REFERENCE SOUND SPEED 1499.990 M/SEC

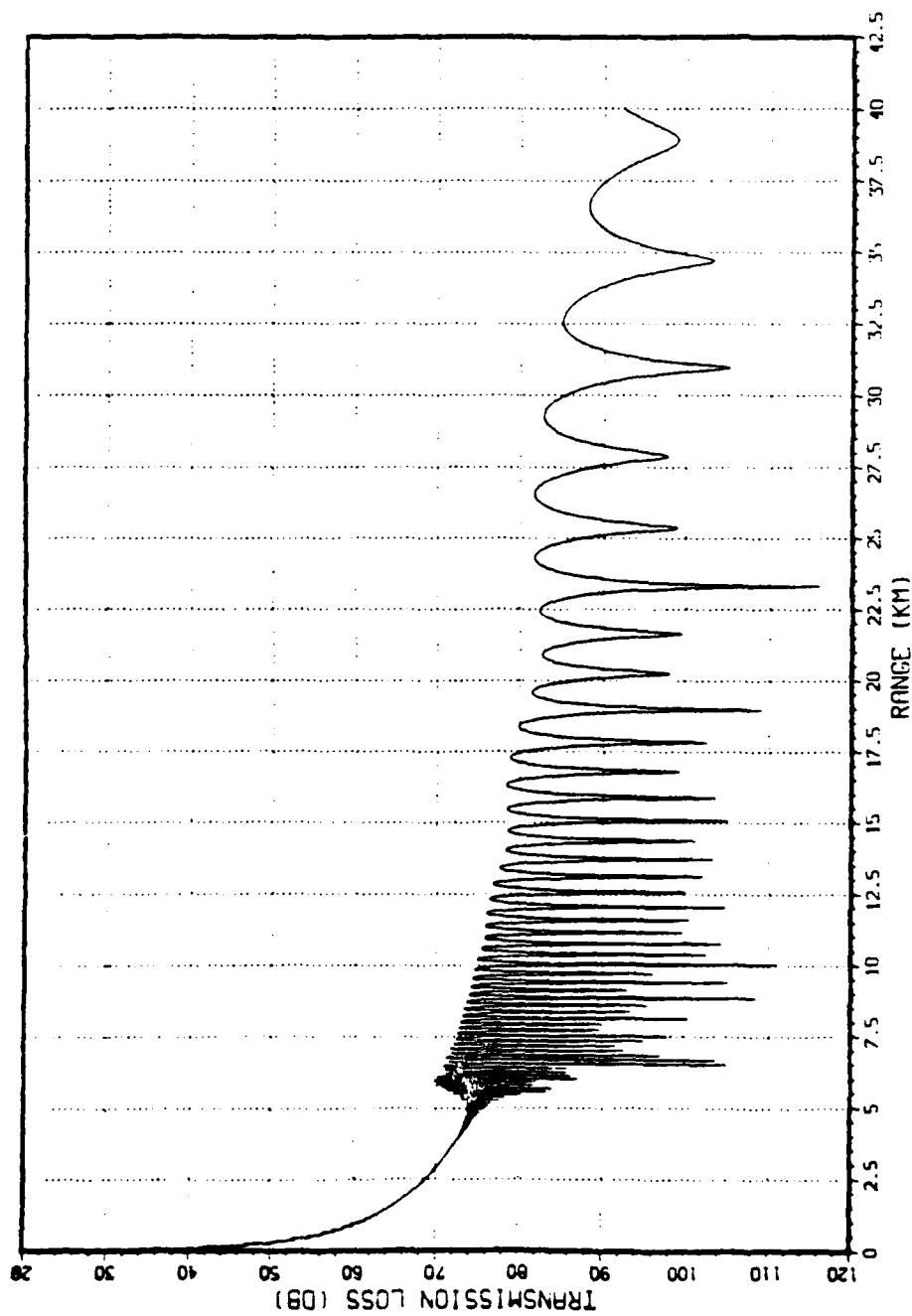


Figure 3.2 Test of SSFFT Model for Bottom Interaction.

2. WT Analysis

The WT output from the SSFFT model is in English units, which is a problem when comparing different models. However, for this investigation the problem was minimal. As expressed by equation 2.6, the reference wavenumber for 100 hz with a reference sound speed of 4921.26 ft/sec (1500 m/sec) is 0.1277 1/ft. After correction of the reference wavenumber for a range increment of 62.3 ft and the 2049 points used to represent the wavenumber spectrum, the maximum beta value should be 0.1248 1/ft which agrees reasonably well with the maximum beta values in figures 3.3, 3.4, and 3.5. Contrary to theory the nulls are not equally spaced and the expected delta beta value of 0.00192 1/ft (equation 2.10) is not represented by any of the null spacings. Stamey [Ref. 3], observed the same non-uniform spacing of beta nulls. The reason why the source depth is not represented by any of the null spacings may be traced to the improper null placement in the range domain. The lack of equal spacing between the nulls must be attributed to something else, possibly the pressure wave was improperly represented as a continuous signal or numerical errors exist within the source code. Whatever the reason for this difference, it must be resolved before the SSFFT can be used with the WT to determine the source depth.

SOURCE DEPTH 1640.42 FT., RECEIVER DEPTH 1640.48 FT., FREQUENCY 100.00 HZ
 WATER DEPTH 6561.68 FT., FIELD DEPTH 1674.59 FT., ATTENUATION COEF 2.368×10^{-3} DB/FT.
 AVERAGE WAVE NUMBER 1.277×10^{-1} 1/FT., REFERENCE SOUND SPEED 4921.26 FT./SEC

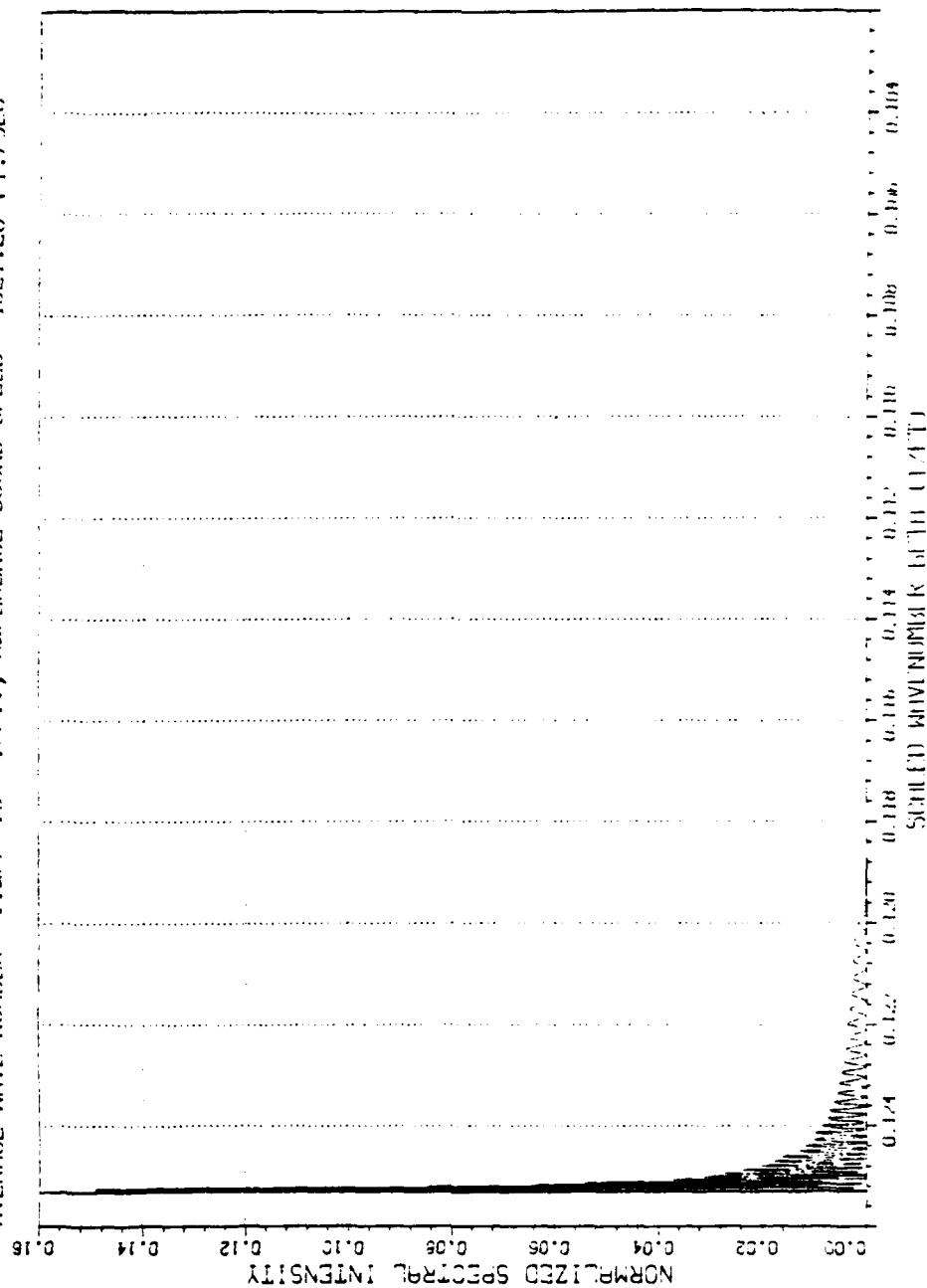


Figure 3.3 SSFFT WT Plot with Source at 500 meters.

SOURCE DEPTH 1640.42 FT., RECEIVER DLP111 3281.10 FT., FREQUENCY 100.00 HZ
 WATER DEPTH 6561.68 FT., FIELD DEPTH 1674.59 FT., ATTENUATION COEF 2.368×10^{-9} 0
 AVERAGE WAVE NUMBER 1.277×10^{-1} 1/FT., REFERENCE SOUND SPEED 4921.26 FT./SEC

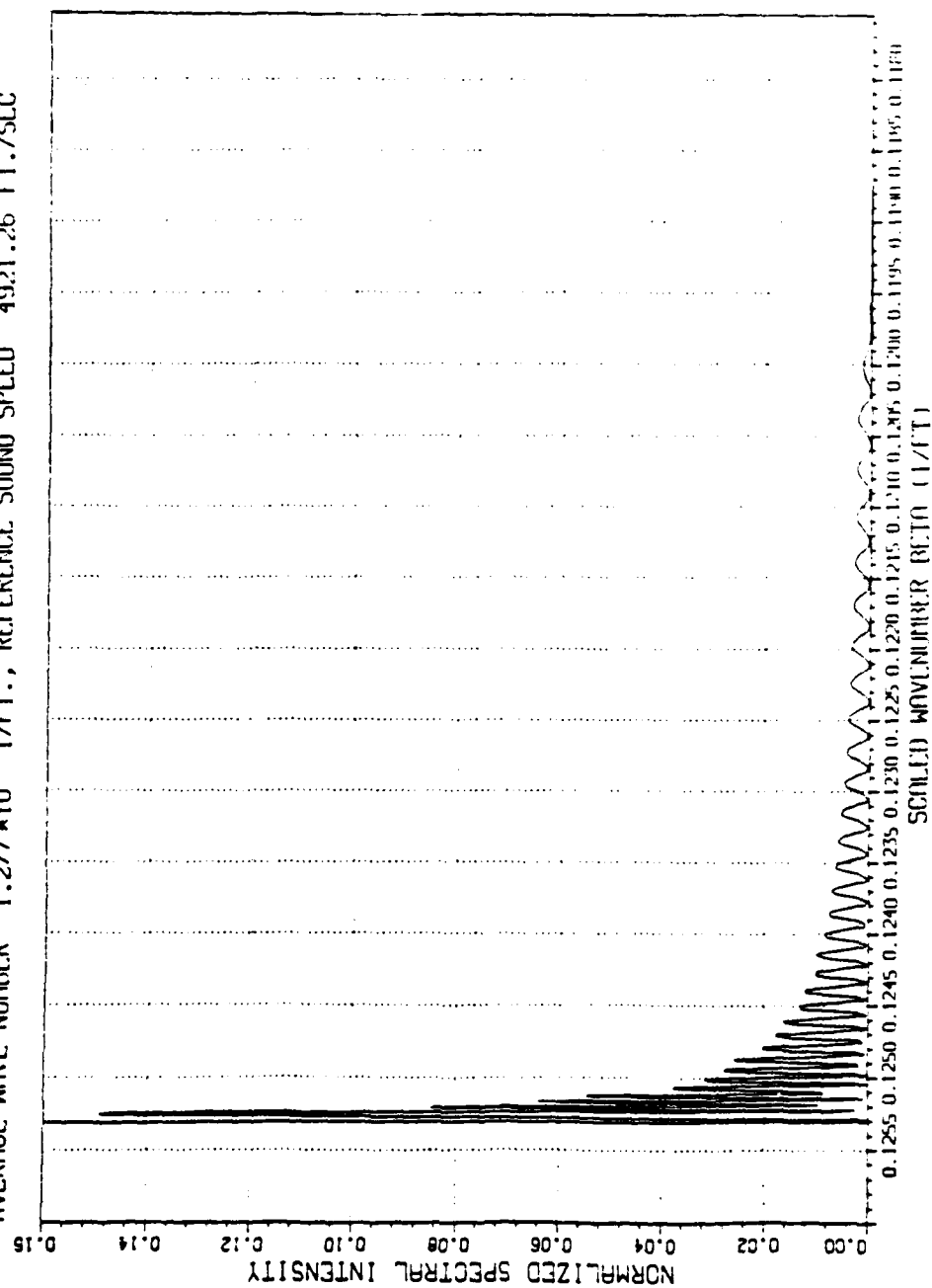


Figure 3.4 SSFFT WT Plot with Receiver at 1000 meters.

SOURCE DEPTH 2624.67 FT., RECEIVER DLP III 3281.10 FT., FREQUENCY 100.00 HZ
 WATER DEPTH 6561.68 FT., FIELD DEPTH 2631.51 FT., ATTENUATION COEF 2.368×10^{-6} [
 AVERAGE WAVE NUMBER 1.277×10^{-1} 1/FT., REFERENCE SOUND SPEED 4921.26 FT./SEC

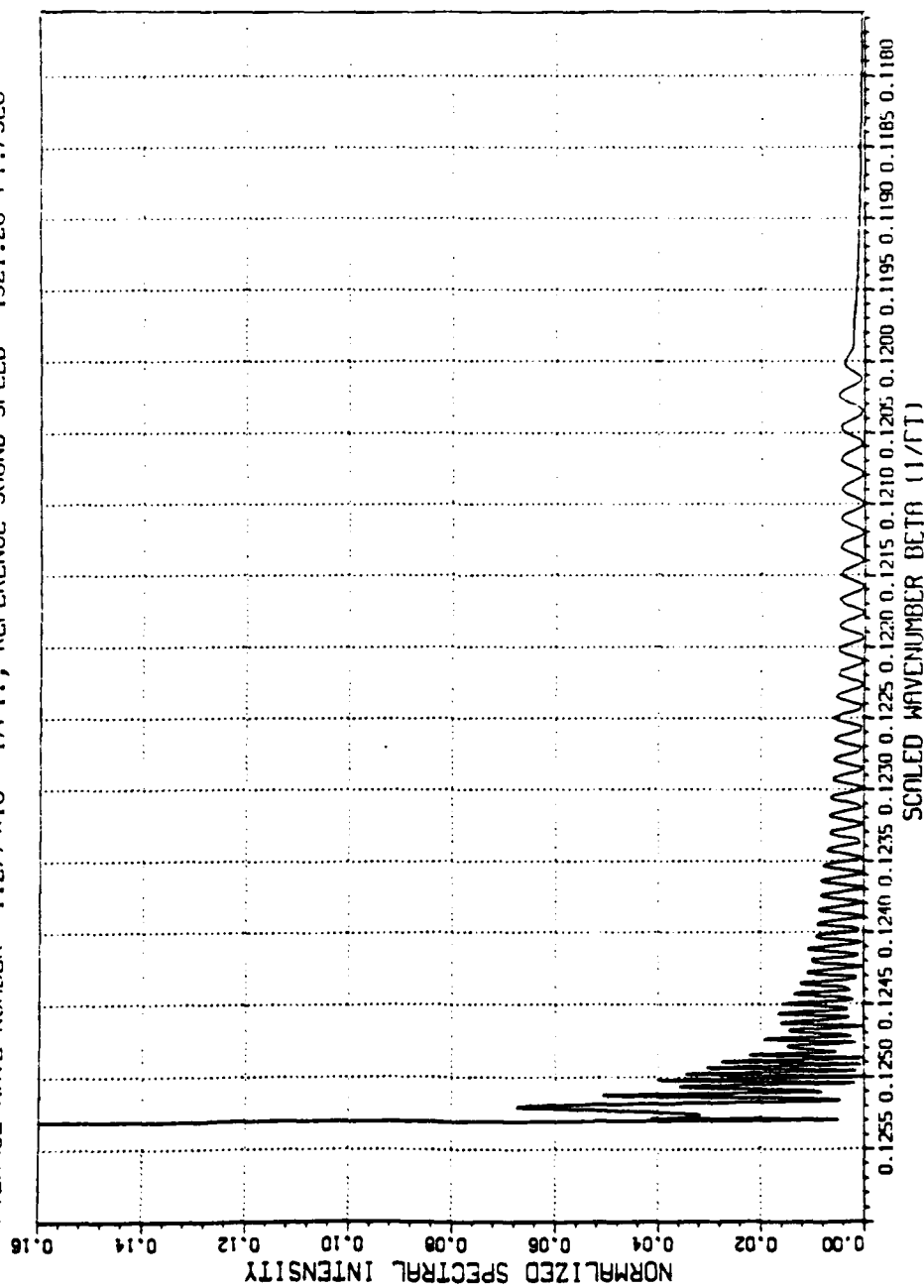


Figure 3.5 SSFFT WT Plot with Source at 800 meters.

C. IFD

1. Transmission Loss Comparison with Lloyd Mirror

The IFD model [Ref. 12] which was installed and tested in September 1983 at NPS [Ref. 13] was the next model chosen as a sound source for the WT. This decision was made because the IFD was capable of properly handling the ocean bottom interactions and was the only other acoustic PE model in residence at NPS. A copy of the source code listing is provided in appendix D.

The IFD also presented problems which will be discussed further. The same initial test run used with the SSFFT was used with the IFD. Since the IFD requires more bottom information, additional testing was performed in order to obtain a fully absorbing bottom. Figure 3.6 illustrates the environment as seen by the IFD model.

The densities in the water mass, sediment, and artificial attenuation layer were set to a constant of 1.0 and the attenuation in the sediment was adjusted until the elimination of bottom interference was observed. Figure 3.7 is an example of the IFD transmission loss output for a 500 m source/receiver depth, a water depth of 2000 m, an upper surface of the artificial attenuation layer at 3000 m, and a lower pressure release surface at 4000 m. The bottom attenuation used to produce figure 3.7 was 0.0016 dB/wavelength.

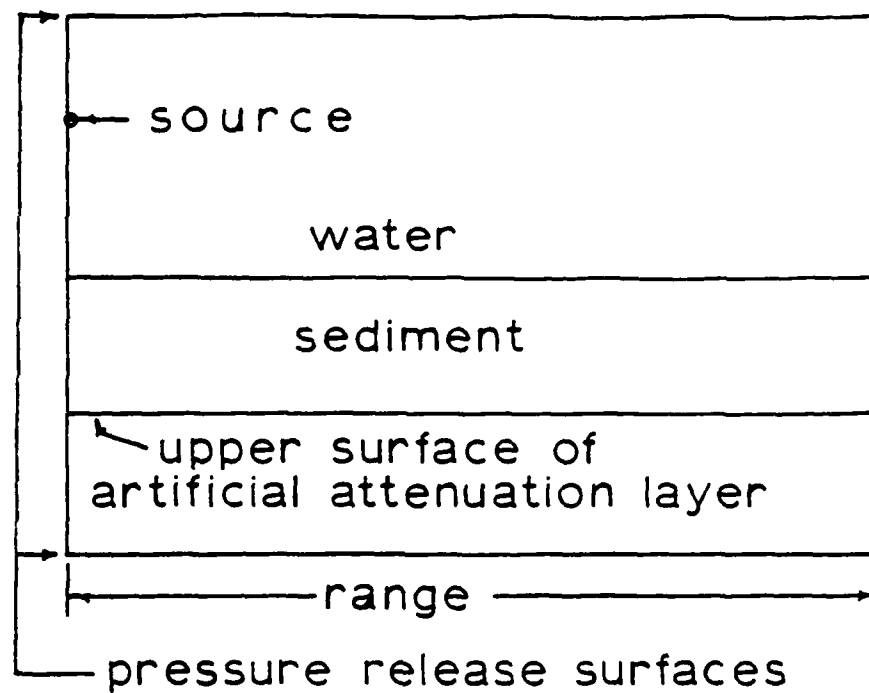


Figure 3.6 IFD Environment.

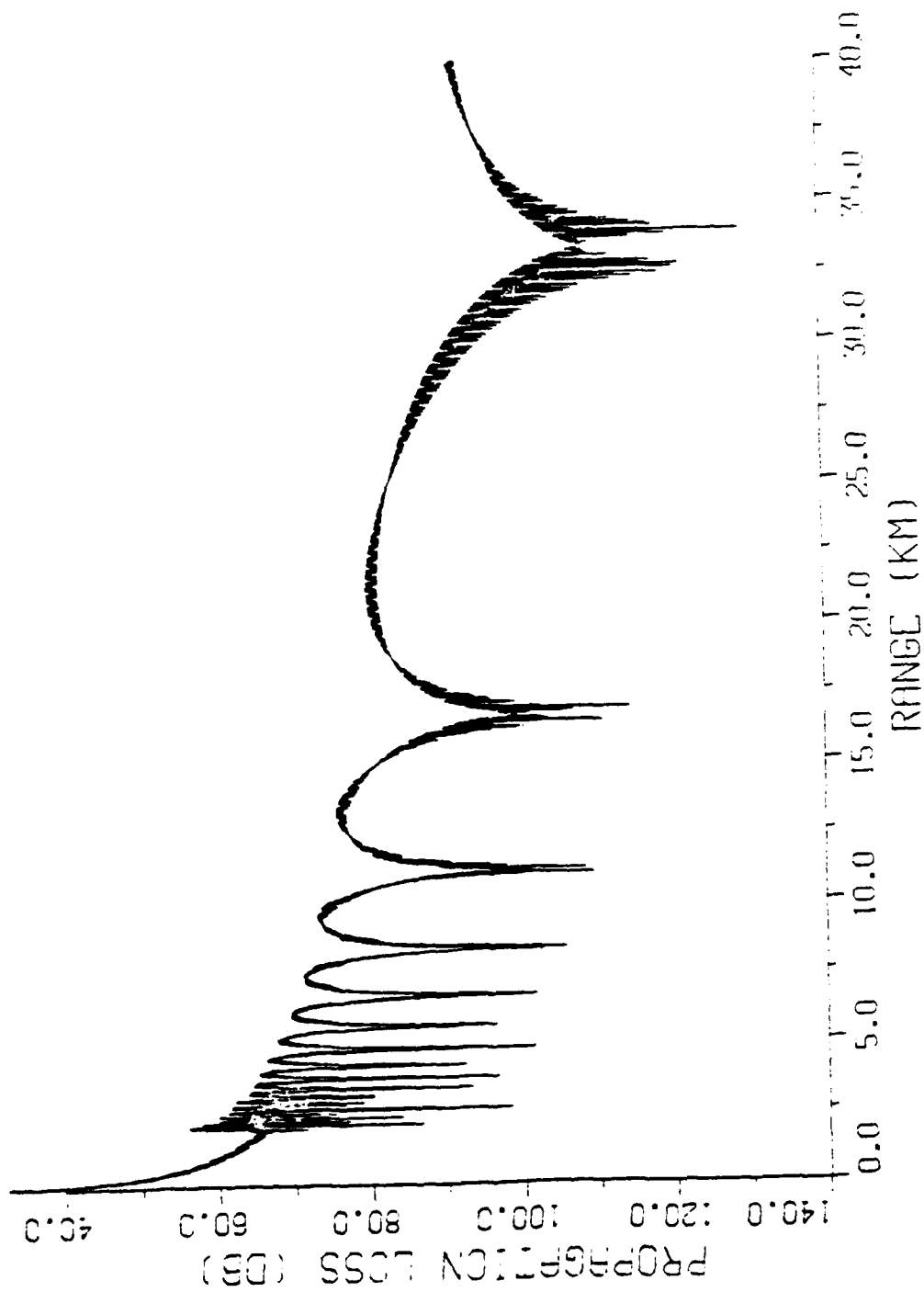


Figure 3.7 IFD Transmission Loss.

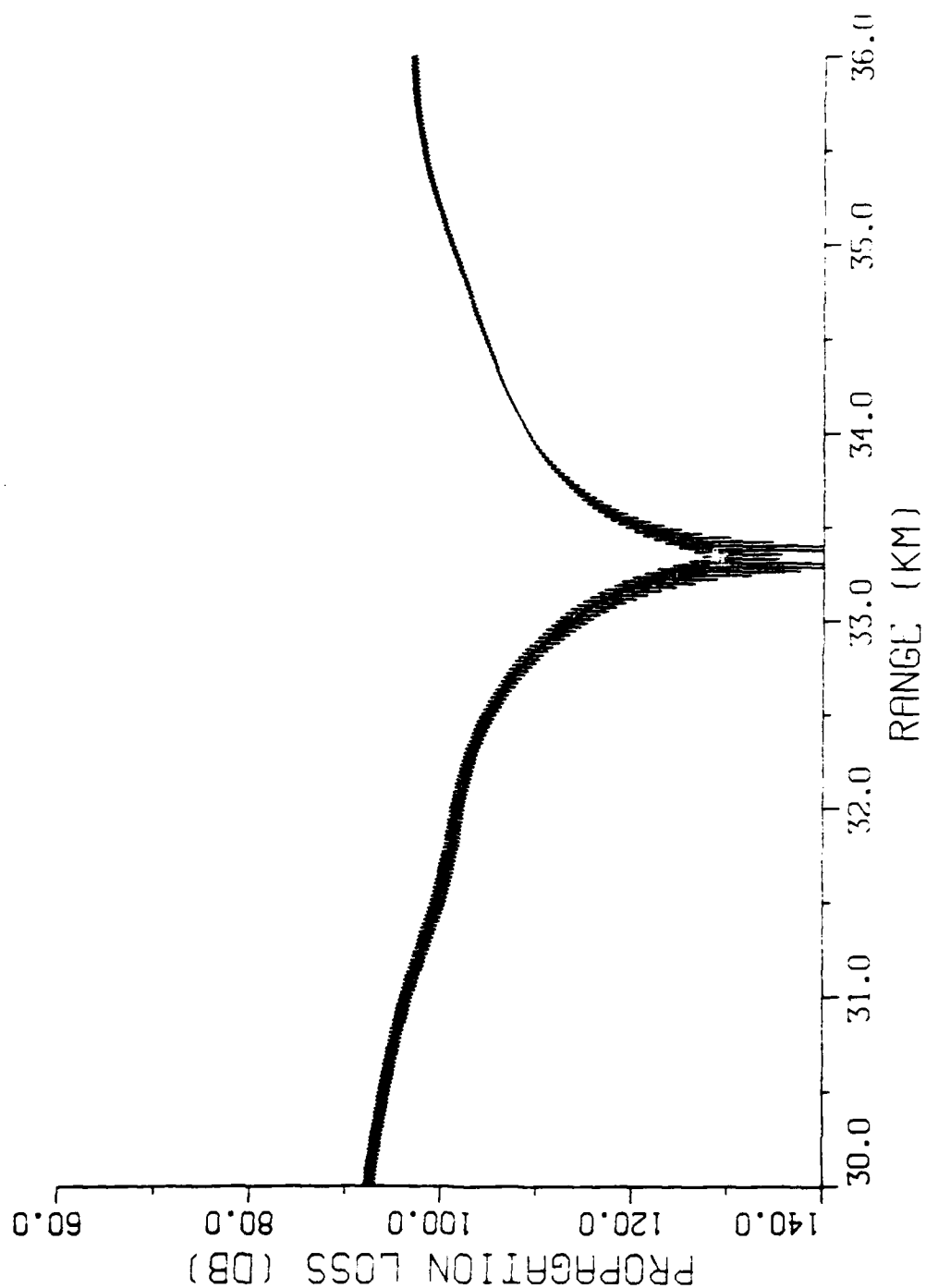


Figure 3.8 IFD Detailed Transmission Loss.

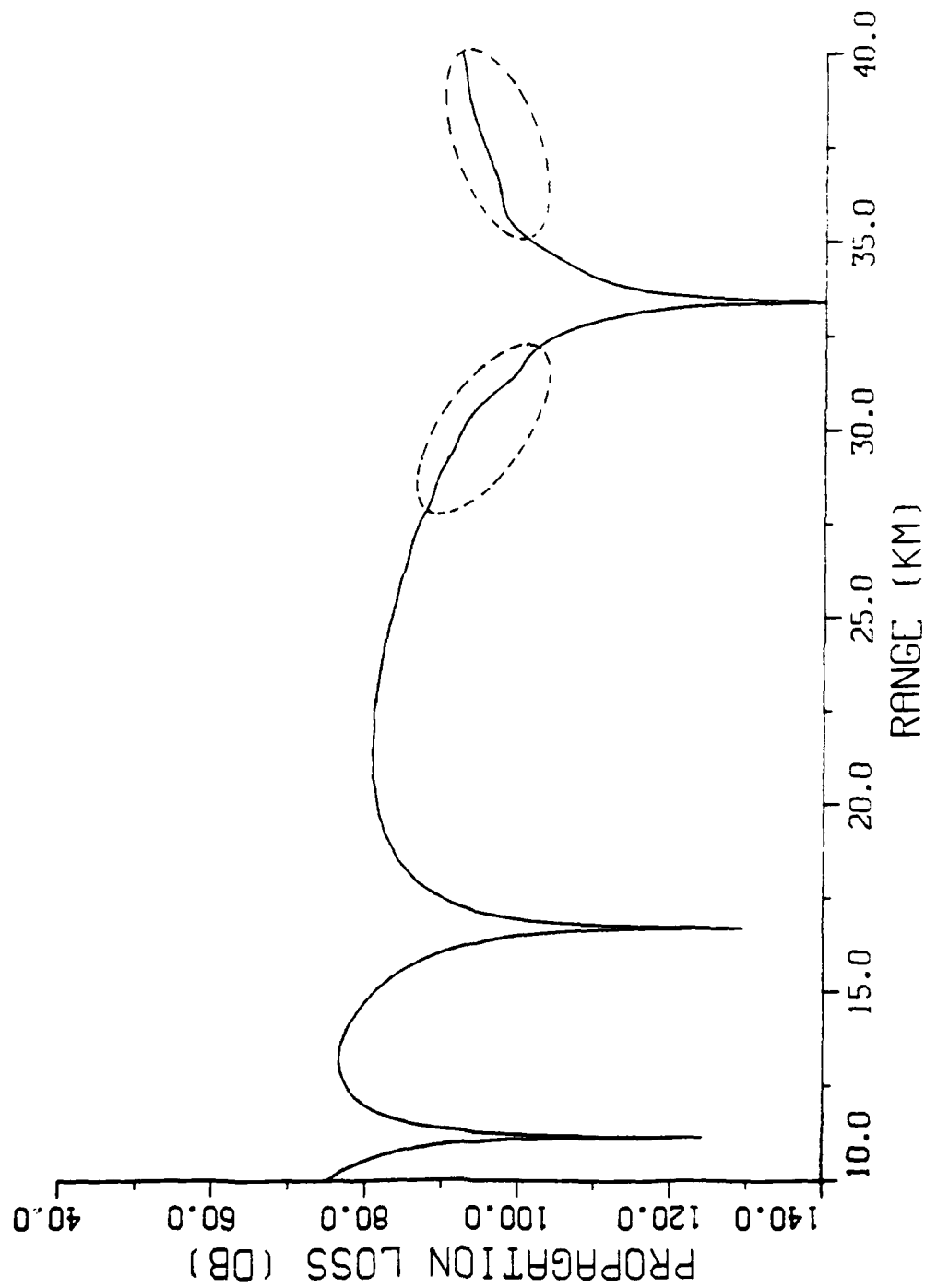


Figure 3.9 IFD Noise Free Transmission Loss.

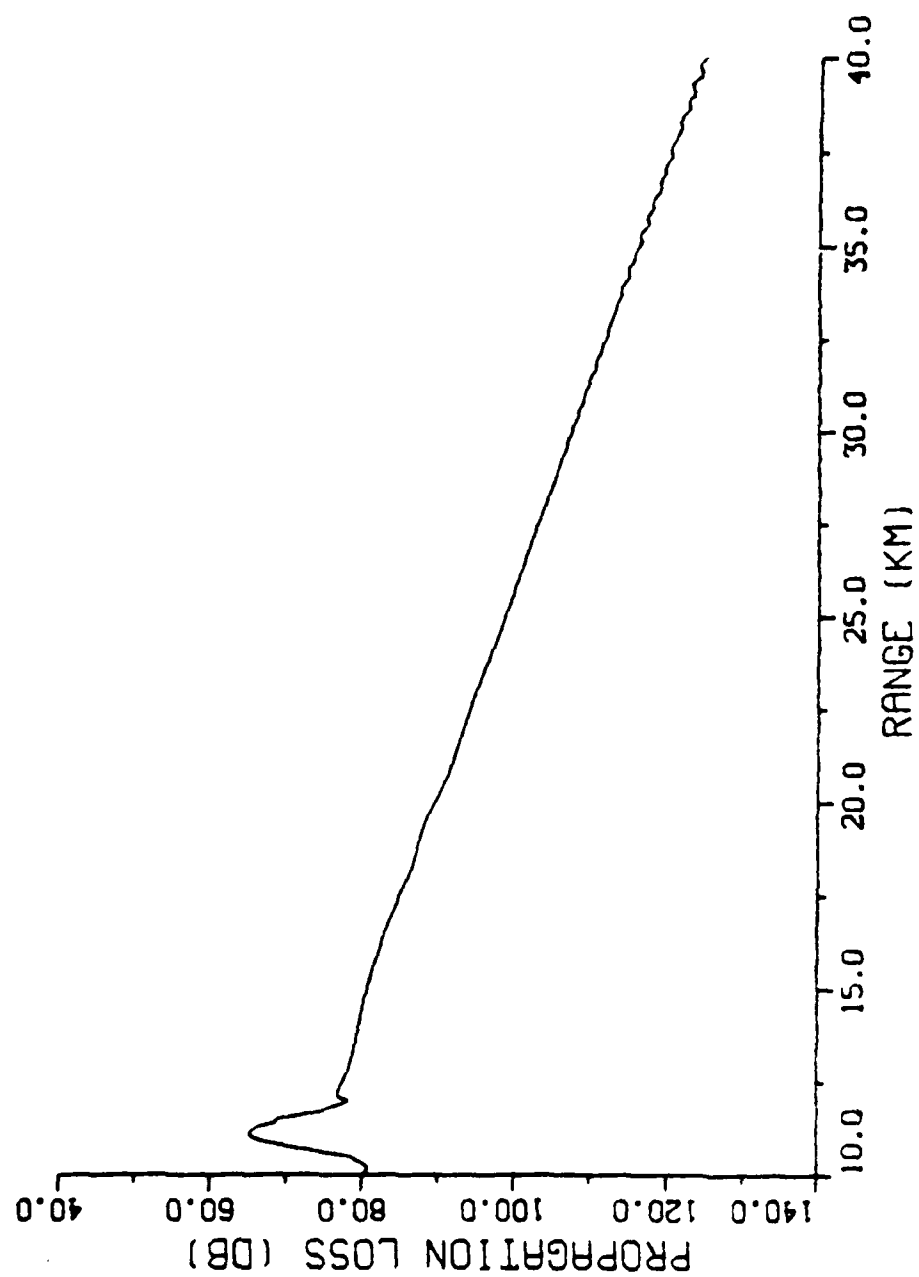


Figure 3.10 IFD: Shallow Water Transmission Loss.

The location of the nulls in figure 3.7 coincide with those predicted by equation 2.4 if the high frequency noise is filtered out by either a running average process or low pass digital filter. A careful examination of the transmission loss curve (figure 3.8) reveals the real problem, which is the long wavelength noise near the wavelength of interest. The long wavelength noise will cause erroneous spikes on the WT plots after being Fourier transformed. Continued analysis revealed that the noise could be eliminated if the thicknesses of the artificial attenuation layer and the bottom pressure release surface are chosen so that they are at least twice the water depth. The choice of a small vertical grid step will reduce the thickness required for the pressure release surface, but this version of the IFD program is limited to 5000 vertical grid points due to software restrictions. Figure 3.9 is a transmission loss curve obtained from a 500 m source/receiver depth, a 2000 m water depth, a 4000 m upper level artificial attenuation layer, and an 8000 m bottom pressure release surface. Comparison of figures 3.1 and 3.9 with the classical Lloyd's mirror transmission loss (figure 2.2) shows that the shape of the curve beyond 30 km is greatly improved with the IFD. The number of vertical grid steps was 5000. When the IFD is used in a shallow water situation and at 25 hz, the noise does not appear to be present. Figure 3.10 is a plot of the transmission loss for a signal at 25 hz, with the source/receiver depth at 50 m, the bottom sloping up from 350 to 50 m, and an artificial attenuation layer and bottom pressure release surface located at 750 and 1000 m, respectively. The noise which was so evident in figure 3.7 is absent from the shallow water case. Producing the noise free curves requires excessive computer time except for shallow water cases.

2. WT Analysis

To reduce the amount of the pressure information and yet preserve the details, the IFD tests were conducted between 10 and 40 km in range. The reference wavenumber for the test input with the IFD was 0.4139 1/m which, after correcting for a range increment of 7.5 m with 4096 points, produced a beta maximum of 0.4188 1/m . Figures 3.11, 3.12, and 3.13 indicate agreement with the beta maximum and a delta beta of 0.00623 1/m can be found among the null spacings. When the vertical grid step is reduced in size, the output signal contains more information of finer detail. The finer detailed information will, if noise is present, produce more clutter on the WT depiction (figure 3.13). In numerous WT depictions a peak was observed at the minimum and maximum wavenumbers. Stamey believed that the right and left intensity maxima corresponded to beam elevation angles of 0 and 30 degrees, respectively. He further stated that the left maxima also may be related to algorithmic inaccuracies [Ref. 3]. The "U-shaped phenomenon" observed during Stamey's investigation of the SSFFT, is present with the IFD and the nulls are not equally spaced.

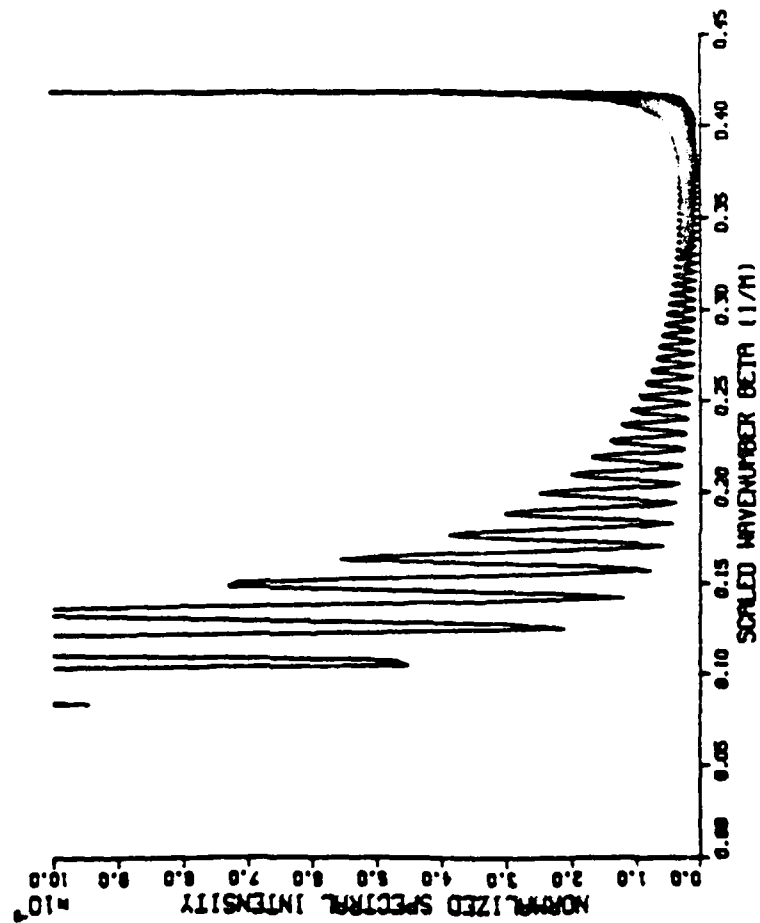


Figure 3.11 IFD: 500 meter Source, 15 meter Vertical Grid Step.

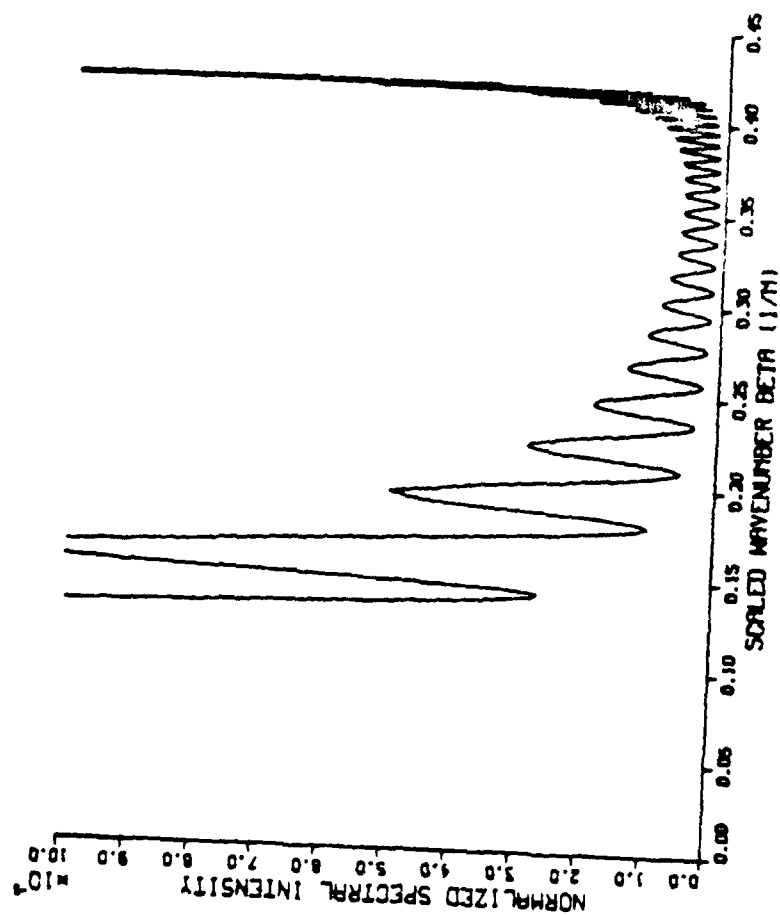


Figure 3.12 IPD: 500 meter Source, 3.75 meter Vertical Grid Step.

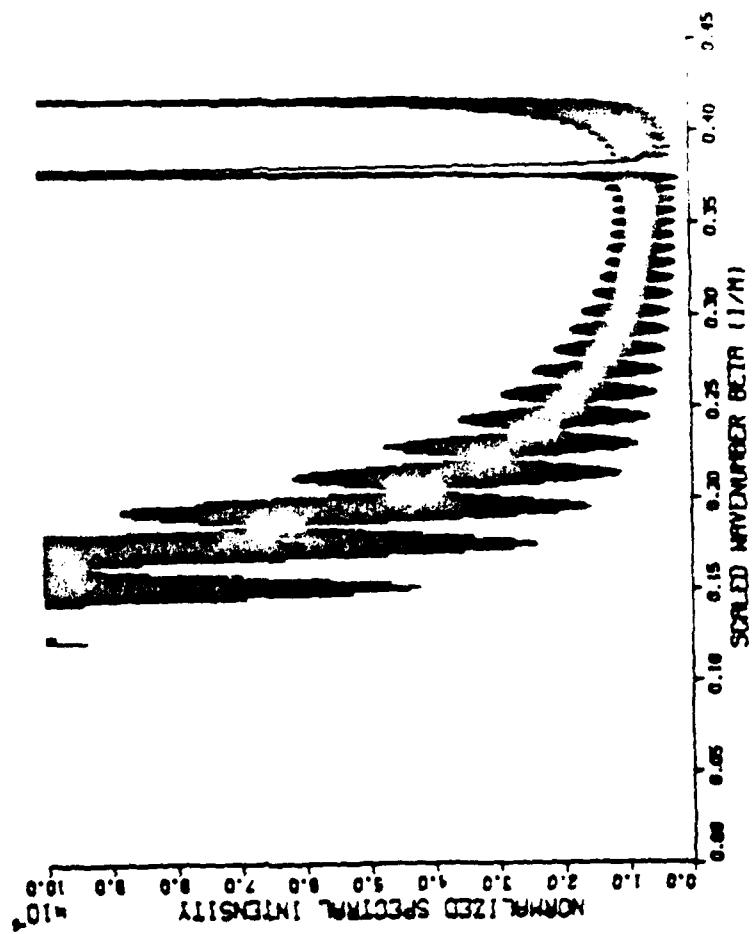


Figure 3.13 IFD: 500 meter Source, 0.8 meter Vertical Grid Step.

The reason for the "U-shaped phenomenon" appears to be related to noise in the envelope function field which is further amplified when the Hankel function is inserted to obtain the complex pressure. Using a low pass filter and increasing the vertical grid step size appear to reduce the problem. In figure 3.8, the nulls occur at 33.30 and 33.39 km when a single null should exist at 33.33 km. This condition or noise produces the most devastating effects by masking the desired density spectrum. A narrow band pass filter would probably solve this problem in the short term but eventually the IFD will require further study to eliminate the noise. When the shallow case is used in the WT (figure 3.14), the "U-shaped phenomenon" is absent and the null locations are more distinct. However, in the shallow water case, the bottom interaction produces high frequency interference but the null spacings occur near the expected value for the source depth (0.0628 1/m).

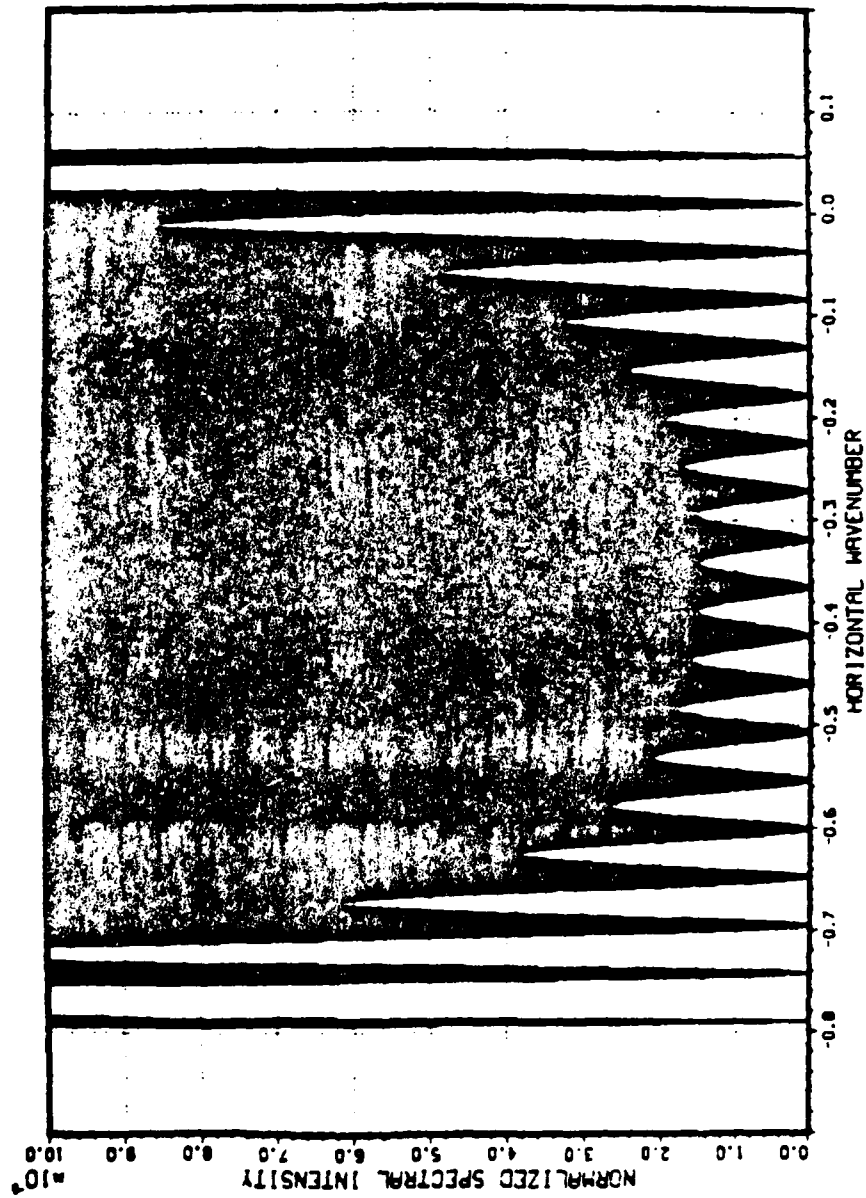


Figure 3.14 IFD: Shallow Water WT, 1 meter Vertical Grid Step.

IV. CONCLUSIONS

Two parabolic equation models, the SSFFT and IFD, were used to predict sound fields for comparison with the Lloyd's mirror interference pattern in the range domain and then in the wavenumber domain. In the range domain, the SSFFT improperly handled the location of pressure nulls and displayed bottom interference. While these weaknesses in the SSFFT were absent in the IFD, the inadvertent and deleterious insertion of noise was extensive. In the wavenumber domain, the noise in the IFD was prohibitive in all but the deep sediment and shallow water cases. The IFD produced results which were comparable with the SSFFT when the noise was not a factor. The range step and the bottom attenuation algorithms of the SSFFT should be reviewed in order to correct those problems described above. In the IFD, the noise problem will require further investigation. If the problem in the IFD is not corrected, this model will be severely limited. With the guidance provided by Lauer [Ref. 2], the wavenumber domain can provide the detailed information needed to quickly correct inconsistencies observed between acoustic models. Therefore, the WT should be investigated further to ensure that any software weaknesses are eliminated and a thorough understanding of the process is assured.

Experiments in the operational environment should be planned to test the WT with actual sound sources. The ship-board experiment could be easily accomplished since the WT can be implemented on micro-computers if the pressure information is provided from an outside source. The WT, if it is going to be considered as a method to determine source depth, appears to have three major shortcomings:

1. A clean CW signal with phase information is essential, which will require quadrature demodulation and extensive filtering.
2. Prediction models are required which will provide equally spaced beta nulls so that software can be developed to automate the process.
3. High resolution processing equipment is necessary because the change in beta spacing for shallow sound sources is small (figure 2.5).

APPENDIX A: SSFFT SOURCE CODE (AT NPS)

PROGRAM PE IS THE INPUT/OUTPUT DRIVER FOR THE PARABOLIC EQUATION MODEL. IT DEFINES THE FORTRAN UNITS, READS AND PRINTS THE INPUT DATA, CALLS PETL TO CREATE A TRANSMISSION LOSS FILE ON FORTRAN UNIT, LT (DISK OR TAPE), CHECKS FOR A CRASH, AND PRINTS A FIELD

INPUT - LC FORTRAN INPUT UNIT
 LP FORTRAN OUTPUT UNIT
 LT FORTRAN FILE UNIT (DISK OR TAPE)
 LWT FORTRAN OUTPUT UNIT FOR WAVENUMBER TECHNIQUE
 TITLE RUN TITLE (TEXT)
 ND NUMBER OF OUTPUT DEPTHS (.LE. 20)
 IFIAT FLAT BOTTOM FLAG
 IPENT PRINT FLAG
 NPIT NUMBER OF LINE PRINTER FIELD PLOT DEPTHS
 ZS INPUT DEPTH (FT)
 F FREQUENCY (HZ)
 CO REFERENCE SOUND SPEED (FT/SEC OR M/SEC)
 DMAX MAXIMUM DEPTH ON TRACK (FT)
 RMAX MAXIMUM RANGE OF TRACK (NAUTICAL MILES)
 DR RANGE STEP (NAUTICAL MILES)
 CD1 STARTING DEPTH FOR FIELD PLOT
 CD2 LAST DEPTH FOR FIELD PLOT
 CLMIN MINIMUM FIELD PLOT LOSS
 DCI FIELD PLOT LOSS INCREMENT
 D OUTPUT DEPTH ARRAY
 NR NUMBER OF OUTPUT RANGES
 NWARN WARNING MESSAGE COUNT FROM PETL

INPUT DECK - CARD 1 TITLE 80 COLUMNS OF TEXT
 FORMAT(20A4)

CARD 2 ... ND,IFLAT,IPRNT,NPIT,ISPH
 FORMAT(16I5)

ND NUMBER OF OUTPUT DEPTHS
 IFLAT FLAT BOTTOM FLAG
 IPRNT PRINT FLAG
 NPIT NUMBER OF FIELD PLOT DEPTHS
 ISPH SPHERICAL EARTH CORRECTION IF .EQ. 0
 IPLOT GRAPHICS PLOT INDICATOR IF .EQ. 0
 NO PLOT, IF .EQ. 1 FLOT SIZE
 FACTOR SCALING FACTOR FOR FLOT SIZE
 DMIN MINIMUM DEPTH USED BY PLOT
 DEFAULTS TO ZERO

CARD 3 ... ZS,F,BEAM,CO,VABSF

ZS SOURCE DEPTH (FT)
 F FREQUENCY (HZ)
 BEAM SOURCE BEAM SIZE (DEGREES)
 CO REFERENCE SOUND SPEED (FT/SEC OR M/S)
 VABSF VOLUME ATTENUATION FACTOR

CARD 4 DMAX, RMAX, DR, CD1, CD2, CLMIN, DCL
 FORMAT(7F10.2)

DMAX MAXIMUM DEPTH
 IF BOTTOM LOSS IS PROVIDED AS SOUND
 SPEED PROFILES IN THE BOTTOM THEN
 THIS PARAMETER MUST BE THE DEPTH AT
 THE BOTTOM OF THE DEEPEST BOTTOM
 PROFILE

RMAX MAXIMUM RANGE
 DR RANGE STEP
 CD1 MINIMUM FIELD PLOT DEPTH
 CD2 MAXIMUM FIELD PLOT DEPTH
 CLMIN MINIMUM FIELD PLOT LCSS
 DCL FIELD PLOT LOSS INCREMENT

CARD 5 D(I), I=1, ND
 FORMAT(8F10.2)

D OUTPUT DEPTHS

CARD 6 NPROF ... NUMBER OF PROFILES (FORMAT(I5))

(REPEAT CARDS 7 AND 8 FOR EACH PROFILE)

CARD 7 ... R(I), NPTS(I) ...
 RANGE OF I-TH PROFILE (NMI)
 NUMBER OF POINTS IN I-TH PROFILE
 FORMAT(F10.2, I5)

CARD 8 ... Z(J), C(J), J=1, NPTS(I)
 THE PROFILE IN DEPTH, SPEED PAIRS. ENGLISH OR
 METRIC. FORMAT(8F10.2)

CARD 9 ... BATHYMETRY
 IF IFLAT IS 0, SKIP THIS SEGMENT. ELSE, (RANGE, DEPTH)
 PAIRS A TOTAL OF (IFLAT) PAIRS. (NMI, FEET)
 FORMAT(8F10.2)

CARDS 10-12 ... BOTTOM LOSS INFORMATION ...

CARD 10 HORIZONTAL RANGE PERIOD FOR RAYS IN THE ABSORBING BOTTOM. (RECALL THAT THE RANGE PERIOD IN THE BOTTOM IS CONSERVED FOR ALL GRAZING ANGLES). SUGGESTED VALUE IS 6000.0 FEET. (FORMAT(F10.2) IF THIS VALUE IS LE 0.0, THE CODE WILL GIVE THE USUAL FULLY ABSORBING BOTTOM TREATMENT AND NO FURTHER INPUT IS NEEDED.)

CARD 11 NBOIM ... NUMBER OF DISTINCT RANGE REGIONS FOR PARTIALLY ABSORBING BOTTOM. IF GTR 0, THEN L(THETA) TABULATIONS WILL BE SPECIFIED FOR EACH REGION AND THE CODE WILL USE AN ANALYTIC INVERSE PARABOLIC PROFILE FORM. OTHERWISE, THE USER MUST SPECIFY THE ATTENUATION ALPHA(Z) (IN DB/FT) AND THE (TABULATED) FORM OF THE SOUND SPEED PROFILE IN THE BOTTOM.

(NBOIM FORMAT ... (15)

IF NBOIM GT 0 ... RANGE(I), NTHETA(I), RANGE (NMI) WHERE CARD 12A ... BOTTOM REGION STARTS NUMBER OF POINTS IN THE L(THETA) CURVE. FORMAT(F10.2,15) ... (8F10.2) CARD 12B (THETA(I,J), LOSS(I,J), J=1, NTHETA(I)) (8F10.2) (TABULATED) ANGLE-LOSS CURVE ... INCIDENT ANGLE, LOSS (DB) ***** REPEAT CARDS 12A-B FOR EACH OF NBOIM REGIONS

IF NBOIM LT 0 ... RANGE(I), NALPHA(I), (F10.2,15) RANGE FOR CARD 12C ... BOTTOM LOSS REGION (NMI) AND NUMBER OF THE I-TH BOTTOM LOSS REGION (NMI) AND FUNCTION ALPHA. POINTS IN THE TABULATED ATTENUATION FUNCTION ALPHA. CARD 12D ... (Z(I,J), ALPHA(I,J), CURVE. SEE BELOW. (8F10.2) TABULATED ATTENUATION VS DEPTH CURVE. SEE BELOW. I-TH PROF. CARD 12E ... NPRINT(I), J=1, NPRINT(I)) (8F10.2) CARD 12F (ZM(I,J), CM(I,J), J=1, NPRINT(I)) (8F10.2) USER SPECIFIED PROFILE IN THE ABSORBING BOTTOM.

***** NOTE ... IN THE USER SPECIFICATION OF ALPHA(Z) AND ***** SOUND SPEED PROFILE IN THE BOTTOM, THE DEPTHS ARE WITH ***** RESPECT TO THE BOTTOM OF THE WATER COLUMN AS ZERO. *****

REPEAT CARDS 12C-F FOR EACH OF LABS(NBOIM) REGIONS

THE BOTTOM LOSS INPUT IS READ IN BY SUBROUTINE PETL

CONVENTIONS - IFLAT = 0 THE BOTTOM IS FLAT
NE.0 BATHYMETRY WILL BE SUPPLIED BY A USER
IPRNT = 0 DO NOT PRINT TRANSMISSION LOSS TABLE
NE.0 PRINT OUTPUT TRANSMISSION LOSS TABLE

```

NPLT = 0 DO NOT PLOT FIELD.
.GT.0 PLOT FIELD ON LINE PRINTER.

CO = 0 TRANSFORM ENVIRONMENT TO REDUCE
.NE.0 PARABOLIC PHASE VELOCITY ERROR.
NO PARABOLIC PHASE VELOCITY CORRECTION.
RUN WITH SPECIFIED REFERENCE SOUND
SPEED.

VBSF = 0 VOLUME ATTENUATION IS CONSIDERED
.GT.0 ATTENUATION IGNORED

DR = 0 VARIABLE STEP SIZE RUN. RANGE STEP
WILL BE DERIVED FROM THE SPLIT-STEP
ALGORITHM TRUNCATION ERROR ESTIMATES.
.GT.0 RUN AT SPECIFIED RANGE STEP.

WARNING - FORCING THE STEP SIZE DISABLES
ALL ERROR CHECKS.

```

CCONSTANTS - FNM CONVERSION FACTOR FT/NAUTICAL MILE
SUBROUTINE - PETL (CREATE TRANSMISSION LOSS FILE)

STEP ADVANCES THE SOLUTION ONE RANGE STEP USING THE
TAPPERT-HARDIN SPLIT-STEP FOURIER ALGORITHM.

STEP OPERATES IN TWO MODES -

1. FIXED STEP - THE SOLUTION IS ADVANCED THE SPECIFIED STEP
SIZE. NO ERROR CHECKS ARE MADE.
2. VARIABLE STEP - THE INTEGRAL AND OPERATOR TRUNCATION ERROR
ESTIMATES ARE EXAMINED EVERY FIFTH CALL
AND THE RANGE STEP IS ADJUSTED ACCORDINGLY.
THE TRANSFORM IS ALSO CHECKED FOR EVIDENCE
OF ALIASING.

INPUT - PR, PI THE FIELD AT RANGE = R
DR THE CURRENT RANGE STEP

OUTPUT - PF, PI THE FIELD AT RANGE = R + DR
DR NEW RANGE STEP (VARIABLE STEP MODE ONLY)

RETURN CODE - FLAG = 0.0 SUCCESSFUL ADVANCE.

```

CC      .GT. 0.0 COMPUTED STEP SIZE IS SMALL.
CC      RANGE STEP NOT CHANGED.
CC      .LT. 0.0 EVIDENCE OF ALIASING.
CC      (FLAG = TRANSFORM ALIASING TEST)
CC
CC      ICCAL VARIAELES - AL2 TRANSFORM L2 NORM
CC      AL4 TRANSFORM ALIASING TEST
CC      ARMS (SIN(RMS ANGLE))**2
CC      RSTEP COMPUTED RANGE STEF
CC
CC      TEMPORARY VAKIABLES - AMP, FL, IR, TI
CC
CC      SUBROUTINES - RST (REAL VECTOR FAST SINE TRANSFORM)
CC      SET (CONSTRUCT STORED TABLES)
CC
CC      ##### START OF EXECUTABLE CODE #####
CC      INTEGER TITLE #####
CC      REAL LOSSI #####
CC      DIMENSION L(20), IEUFC(21) #####
CC
CC      CCDE BETWEEN $$$ HAS BEEN ADDED TO ENABLE DISSELA GRAPHICS
CC      AND WAVENUMBER TECHNIQUE FOR NPS INSTALLATION
CC
CC      CCOMMON /PI1G/ IDLOT, FACT, DDDD
CC      CCOMMON /WTFID/WPR(2049), WPI(2049), PRS(2049), GA
CC      CCOMMON /UNITS/ IC, LP, LT
CC      CCOMMON /EARTH/ ISPH
CC      CCOMMON /OUTEUF/ NOUT, BUFO(21)
CC      CCOMMON /HEFTZ/ CO, H, FK, F, FACTOR, WL
CC      CCOMMON /PLT/TITLE(20), NPL1, LCR, CLMIN, DCL, CD1, CD2, DCD, CD(120)
CC      CCOMMON /LOSFCN/ T HET1(50), LOSSI(50), NLTH, ZHAT(100), ALGEN(100),
CC      1 HORRAN, JA, NHAT, RAPRES, RANEXT, CMUD(30), ZMUD(30), NMUD, NBC1M
CC      REAL LOSSI
CC      CCOMMON /MESH/ R, DR, NR, KR, DZ, ZMAX, IB, N, NPTS, N2, N4, NL4, NA, NW, ZK,
CC      1 HALF, NEFFK
CC      CCOMMON / CCSTR / VABSF , ATEN
CC      CCOMMON /ERRCRF/FLAG
CC      EQUIVALENCE (BUFO(1), IBUFO(1))
CC      DATA FNM/6C76.1/
CC
CC      DEFINE FORTFAN UNITS.
CC
CC      IC=5
CC      LP=6
CC      LI=1

```

```

C      LWTF=3
C      LWTF=7
C      REWIND 4
C
C      READ INPUT DATA.
C
C      900  READ (LC,90C) TITLE
C           FORMAT (20A4)
C
C      910  READ (LC,910) ND, IFLAT, IPHNT, NPLT, ISPH, IPIOT, FACT, DMIN, IFLAGU
C           FORMAT (I5, 2F5.0, I5)
C           NPLT=MINO (NPLT, 120)
C
C      READ (LC,920) ZS, F, BEAM, CO, VABSF
C           ZSHOLD = ZS
C           FORMAT (8F10.2)
C
C      READ (LC,920) DMAX, HMAX, DR, CD1, CD2, CLMIN, DCL
C
C      RMAX=FMAX
C      READ (LC,92C) (D(I), I=1,ND)
C
C      READ (LC,910) NPROF
C      IF (NPROF.NE.0) CALL RDPROF (NPROF)
C
C      CALL GETBOT (IFLAT, DMAX)
C
C      930  WRITE (LP,930) TITLE
C           FORMAT (1H1, 20A4)
C
C      935  WRITE (LP,935) ND, IFLAT, IPHNT, NPLT, ISPH, IPICT
C           FORMAT (14HC INPUT DATA --/
C           13HO PARAMETERS -, 16I5)
C
C      IF (IFLAT.EC.0) WRITE (LP,950)
C           FORMAT (20HO THE BOTTOM IS FLAT.)
C
C      IF (IFLAT.NE.0) WRITE (LP,955)
C           FORMAT (31HO THE BOTTOM IS RANGE DEPENDENT.)
C
C      WRITE (LP,96C) DMAX, ZSF
C      FORMAT (17HO MAXIMUM DEPTH = , F7.1, 3H FT/,
C      1 17H SOURCE DEPTH = , F7.1, 3H FT/,
C      2 17H FREQUENCY = , F7.1, 3H HZ)

```

```

965 C IF (BEAM.LT.1.) BEAM=20.
C WRITE (LP,965) BEAM
C FORMAT (17H BEAM WIDTH = ,F7.1,8H DEGREES)

970 C WRITE (LP,970)
C FCORMAT (14H OUTPUT DEPTHS)

C DC 10 I=1, NL
C FI=I
10 C WRITE (LP,975) FI, D(I)
975 C FORMAT (F4.0, F8.1, 3H FT)
C
C SPHERICAL EARTH CORRECTION
C NO CORRECTION IF ISPH GT 0
C IF (ISPH.GT.0) GO TO 12
C
C DC 11 I=1, NL
C D(I)=D(I) * (1.0+D(I)/4.1807E7)
11 C CONTINUE
C ZS=ZS*(1.0+ZS/4.1807E7)
12 C CONTINUE
C
C
C IF (VALSF.E1.0.) WRITE (LP,979)
979 C FCORMAT (36H VOLUME ATTENUATION HAS BEEN IGNORED)
C
C IF (DR) 16,17,15
C
C WRITE (LP,980) DR
15 C FCORMAT (14H ORANGE STEP = ,F5.3,4H NM.)
980 C DRHOLD = DR
C DR=FNM*DR
C GC TO 18
C
C DR=0.
16 C WRITE (LP,985)
17 C DRHOLD = DR
985 C FCORMAT (24H VARIABLE STEP SIZE RUN.)
C
18 C CONTINUE
C FMAX=FNM*RMX
C
C COMPUTE TRANSMISSION LOSS.
C
C REKIND LT
C
C CALL PETL (ZS, BEAM, NL, D, DMAX, RMAX, IFLAT, NWARN)

```


[illegible]


```

C      FLD (FIELD PRINT PLOTTER)
SUBROUTINE FETL(ZS,BEAM,ND,D,DNAX,RMAX,IFLAT,NWARN)
INTEGER TITLE
DIMENSION I(20),BUFO(21)

C      CCDE BETWEEN $$$$ HAS BEEN ADDED TO ENABLE DISSPLA GRAPHICS
C      FOR NPS INSTALLATION AND WAVENUMBER TECHNIQUE
C      $$$$
C      COMMON /PLTG/ IPLOT,FACT,DDDD
C      DIMENSION CDOWN(5000)
C      COMMON /WFELD/ WPR(2049),WPI(2049),PRS(2049),GA
C      COMMON /FIELD/ PR(2049),PI(2049)
C      $$$$
C      COMMON /UNITS/ LC,LP,LT
C      COMMON /LOSFCN/ THETI(50),LOSSI(50),NLTH,ZHAT(100),ALGEN(100),
1 HORRAN,JA,NHAT,KAPRES,KANEXT,CMUD(30),ZMUD(30),NMUD,NBOTH
C      REAL LOSSI
C      COMMON /OUTEUF/ NCUT,RNM,TL(20)
C      COMMON /HEFTZ/ CO,H,EK,F,FK,FACTOR,WL
C      COMMON /PHASE/ NC,Z(100),C(100),MC,DM(20)
C      COMMON /PLT/TITLE(20),NPLT,LCR,CLMIN,DCL,CD1,CD2,DCD,CD(120)
C      COMMON /BATHY/ EE,KE,NB,BR(101),BZ(101)
C      COMMON /ERRCRF/ FLAG
C      COMMON /MESH/ R, DR, NR, KR, NW, NZ, ZMAX, IB, N, NPTS, N2,
A      N4, NL4, NA, NA', NW', ZW, HALF, NEFFW
C      COMMON / CCSTR/ VABSF,ATTEN
C      EQUIVALENCE (BUFO(1),RNM)
C      DATA FT,FNM,TWOPI,CUT/O.3048,6070.1,6.28318530717959,-14.0/
C      DATA RAD/O.17453252519943E-01/
C      DEFINE MAXIMUM TRANSFORM SIZE.
C      NMIN=5
C      NMAX=2048
C      EXTEND THE PHYSICAL EXTENT OF THE PROBLEM TO 4/3 THE
C      EFFECTIVE WATER COLUMN DEPTH, WHERE THE EFFECTIVE
C      WATER COLUMN IS GIVEN BY THE ACTUAL WATER COLUMN PLUS
C      TWENTY PERCENT OF THE HORIZONTAL RANGE PERIOD. THIS
C      SHOULD BE SUFFICIENT TO TURN AROUND RAYS OF UP TO 38 DEGRE
C      IN THE ELLIPTIC APPROXIMATION.
C      READ (LC,702) HORRAN
C      HCRKAN=AMAX1(HORRAN,0.0)
C      ZMAX = (4.C/3.0) * (CMAX + (HORRAN/5.0) )

```


DEFINE THE AVERAGE ACOUSTIC WAVELENGTH AND WAVE NUMBER
AND THE MESH INCREMENT IN TRANSFORM SPACE.

WL=CO/F
FK=TWOPI/WL
H=THOPI/(ZMAX+ZMAX)
HK=H/FK

--TRANSFORM SIZE--
DETERMINE THE NUMBER OF POINTS REQUIRED

STHC=SIN(RAD*BEAM)

N=1.+ALOG(4.*STHC/(3.*HK))/ALOG(2.0)

IF(N.LE.NMIN) N=NMIN

WRITE(LP,900)

FORMAT(30H0SELECTED TRANSFORM SIZE = 2**,I2)

DEFINE CONSTANTS.

NPTS=2**N

DETERMINE EFFECTIVE BEAM WIDTH

STHCP=(3./4.)*NPTS*HK

IF(STHCP.GT.1.0) STHCP=1.0

THC=ARSIN(STHCP)/RAD

WRITE(LP,902) THC

FORMAT('EFFECTIVE BEAM WIDTH (DEG) = ',F4.1)

CHECK TRANSFORM SIZE.

IF(NPTS.LE.NMAX) GO TO 15

WRITE(LP,905)

FORMAT(4H0TRANSFORM SIZE EXCEEDS ARRAY DIMENSIONS.)

STOP

N2=NPTS/2


```

710 CCNTINUE
GC TO 735
C
C      USER SPECIFIED C(Z) AND ALPHA(Z)
C
720 NNN=IAES(NECTM)
DC 730 I=1,NNN
READ(LC,702) R,NHAT
READ(LC,703) (ZHAT(I),ALGEN(L),L=1,NHAT)
READ(LC,701) NMUD
READ(LC,703) (ZMUD(L),CMUD(L),L=1,NMUD)
EACKSPACE 4
R=R*FNN
WRITE(4) R,NHAT,ZHAT,ALGEN,NMUD,ZMUD,CMUD
R1=1.0E15
WRITE(4) R1
WRITE(LP,706) (ZHAT(I),ALGEN(L),L=1,NHAT)
730 CCNTINUE
735 CONTINUE
REWIND 4
*****
C
C      INITIALIZE FOR RANGE LOOP.
C
R=0.
KE=0
NR=0
KR=1
LCR=0
NWARN=0
IE=NPTS
RE=0.0
RANEXT=1.0E10
READ(2)
CALL SVP(NC,Z,C,RNEXT)
IF (RE.GT.RNEXT) GO TO 17
17 CCNTINUE
IF(NBOTH-EG.0) GO TO 19
IF(NBOTH-GT.0) READ(4) RAPPRES,NHAT,ZHAT,ALGEN
IF(NBOTH-LT.0) READ(4) RAPPRES,NHAT,ZHAT,ALGEN,NMUD,ZMUD,CMUD
READ(4) RANEXT
EACKSPACE 4
IF(RE.GT.RANEXT) GO TO 18
19 CONTINUE
CALL FILTEF(ND,D)
DATE = 107E.

```

```

C      WRITE(LT)TITLE,WHEN,F,ZS,NR,ND,(D(I),I=1,ND)
C      CCNSTRUCT THE INITIAL FIELD.
C
C      ZSM=ZS
C      IF (MC.EQ.1) ZSM=ZSM*SQRT(CO/SPEED(ZSM))
C      CALL SOURCE(ZSM)
C
C      INTRODUCE BOTTOM LOSS TO FILTER FUNCTION IF APPLICABLE
C
C      IF(NBCTM.NE.0) CALL FILLOS
C      IF THE RANGE STEP HAS BEEN SPECIFIED, CONSTRUCT THE STORED
C      TABLES AND TURN OFF THE ERROR CHECKS BY ZEROING THE FLAGGED
C      STEP COUNTER.
C
C      IF(DR.GT.0.C) KK=0
C
C      CCDE BETWEEN $$$$$$ HAS BEEN ADDED TO ENABLE DISSPLA GRAPHICS
C      FOR NPS INSTALLATION AND WAVENUMBER TECHNIQUE
C      $$$$$$
C      DC 92 IWT=1,NW
C      DDDD = IWT * DZ
C      IF(ZS.LT. LDDD) GO TO 93
C      92 CONTINUE
C      93 CONTINUE
C      $$$$$$
C      BEGIN RANGE LOOP.
C
C      NR=NR+1
C
C      GET BOTTOM INDEX AT CURRENT RANGE.
C
C      IF (IFLAT.EC.0) GO TO 30
C
C      ZW=ZB(RE)
C      IF (MC.EQ.1) ZW=ZW*SQRT(CO/SPEED(ZW))
C      IE = ZW/DZ + 0.5
C      IF (IB.GE.NW) IB = NPTS
C
C      ADVANCE THE SOLUTION.
C
C      CALL STEP(FLAG)
C      R=R+DR
C      RE=R
C
C      30
C

```

[illegible]

```

C      CCOMPUTED RANGE STEP IS SMALL.
C
C      WRITE(LP,920) FLAG
920    FORMAT(15X,32H RATIO OF COMPUTED RANGE STEP TC,
C      1 22H ACCUSTIC WAVELENGTH =,E10.3)
C      NWARN=NWARN+1
C      IF (NWARN.GE.5) GO TO 100
C
C      WRITE(LP,930)
930    FORMAT(15X,33H PROCEEDING AT CURRNT RANGE STEP.)
C      GC TO 90
C      TRANSFORM ALIASING TEST FAILED.
C
C      WRITE(LP,910) RNM
80     WRITE(LP,940) FLAG
940    FCRMAT(15X,25H TRANSFORM ALIASING TEST =,F6.1,4H DB.)
C      NWARN=NWARN+1
C      TERMINATE THE RUN IF ALIASING IS SEVERE.
C      IF (FLAG.GT.CUT) NWARN=NWARN+4
C      IF (NWARN.GE.5) GC TO 100
C      END RANGE ICOP.
C      IF (R.IT.RMAX) GO TO 20
90     RETURN
100    END
C      SUBROUTINE STEP(FLAG)
C      COMMON /LOSFCN/ THETI(50), LOSSI(50), NLTH, ZHAT(100), ALGEN(100),
1  HORRAN, JA, NHAT, RAPRES, RANEXT, CMUD(30), ZMUD(30), NMUD, NBC1M
C      REAL LOSSI
C      COMMON /FIED/ PR(2049), PI(2049)
C      COMMON /HERIZ/ CO, H, HK, F, FFACTOR, WL
C      COMMON /BATHY/ RE, KE, NB, BR(101), BZ(101)
C      COMMON /MESH/ R, DR, NK, KR, DZ, ZMAX, IB, N, NPTS, N2, N4, NL4, NA, NW, ZW,
1  HALF, NEFFW
C      COMMON /TABLE/ SR(2047), SI(2047), UL(2047), UI(2047), FN(2047)

```

```

COMMON /OUTBUF/NOUI,RNM,TL(20)
COMMON /COSTR/VAESI,ATTEN
COMMON /FIL1/FIL(2047)
DATA ALIAS,DRMAX,RMSMIN/1.0E-02, 3038.05, 1.0E-02/
DATA NCB,NBEF/0,0/
DATA NOB/0,NBEF/0/

```

```
SET RETURN FLAG.
```

```
FLAG=0.0
```

```
FOURIER TRANSFORM (ONLY IF NOT THE FIRST STEP)
```

```
IF(NR.EQ.1) GO TO 1
```

```
CALL RST(PR,N,1)
CALL RST(PI,N,1)
1 CCCONTINUE
```

```
CHECK FOR FIRST STEP OF FIXED-STEP -- APPLY SPLIT-STEP
```

```
IF(KR.NE.0) GO TO 3
IF(NR.NE.1) GO TO 60
RSTEP = DR
DR = 0.0
GO TO 40
3 CCCONTINUE
```

```
COMPUTE VARIABLE RANGE STEP SIZE BY FIRST
SEARCHING FOR K-SPACE PEAK
```

```
FL = 0.
AL2 = 0.
ARMS = 0.
APEAK = 0.
```

```
DC 5 I = 1,NPTS
AMP = PR(I)*PR(I) + FI(I)*PI(I)
AL2 = AL2 + AMP
APEAK = AMAX1(APEAK,AMP)
```

```
FI = FL + EK
ARMS = ARMS + FL*FL*AMP
5 CCCONTINUE
ARMS = ARMS/(AL2+ARMS)
```

```
FIND LAST POINT IN K-SPACE AT LEAST
```

```

C      50 DB DCKN FROM PEAK
C
N50 = NPTS
DC 8 I = 1, NPTS
AMP = PR(N50)*PI(N50) + PI(N50)*PI(N50)
RATIO = AME/APEAK
IF(RATIO.GE.1.E-5) GO TO 10
N50 = N50 - 1
8 CONTINUE

C      DETERMINE RANGE STEP USING 50 DB DOWN ANGLE
C      (COMPUTE RMS ANGLE FOR COMPARISON)
C
10 SINA = FLOAT(N50-1)*HK
SINA2 = SINA*SINA
IF(SINA2.GT.1.0) SINA2 = 1.0
CCSA = SQRT(1.-SINA2)
RSTEP = AMIN1(WL/(1.-CCSA), DRMAX)

C      CHECK FOR DIAGNOSTIC STEP
C
IF(NR-KR) 30,15,60

C      DIAGNOSTIC STEP - EXECUTE ALIASING TEST, EXAMINE
C      ENERGY DISTRIBUTION IN K-SPACE
C
15 KR = KR + 5
HL2 = 0.
HL4 = 0.
N116 = NPTS - NPTS/16
AL16 = 0.
DC 20 I = 1, NPTS
AMP = PR(I)*PI(I) + PI(I)*PI(I)
IF(I.LE.N2) HL2 = HL2 + AMP
IF(I.LE.N4) HL4 = HL4 + AMP
IF(I.GE.N116) AL16 = AL16 + AMP
20 CCNTINUE

C      SET ALIASING FLAG IF TOLERANCE EXCEEDED
C
AL16 = AL16/AL2
IF(AL16.GT.ALIAS) FLAG = 10.*ALOG10(AL16)

C      IF(DR.LE.0.) GO TO 40
C
C      TRUNCATE SPECTRUM IF POSSIBLE
C
IF((ARMS.LT.RMSMIN).OR.(NEOTA.NE.0)) GO TO 30

```

```

C      HI4 = {HL2-HL4}/HL4
C      HI2 = {AL2-HL2}/HL2
C      IF((HL2.GT.1.E-7).OR.(HL4.GT.1.E-6)) GO TO 30
C
C      TRUNCATE THE SPECTRUM.
C
N=N-1
NPTS=NPTS/2
N2=N2/2
N4=N4/2
NI4=NL4/2
IR=IB/2
NA=NA/2
NW=NW/2
NEFFW=NEFF/2
DZ=DZ+DZ
HALF=0.50*HALF
K=0
DC 25 I = 1,NPTS
K=K+2
FIL(I)=FIL(K)
FN(I)=FN(K)
UI(I)=UI(K)
AMP=SR(I)*EF(I)-SI(I)*PI(I)
PI(I)=SR(I)*PI(I)+SI(I)*PR(I)
EF(I)=AMP
SR(I)=SR(I)+SR(I)
SI(I)=SI(I)+SI(I)
25 SI(I)
GO TO 80
CHECK RELATIVE CHANGE IN STEP SIZE.
IF (ABS(DR-ISSTEP)/DR.LE.0.25) GO TO 60
PREPARE FOR NEW RANGE STEP.
FACTOR=0.25*(DR+RSTEP)/FK
DR=RSTEP
NFIL=(3./4.) * FLOAT(NPTS+1)
DC 50 I=1,NPTS
FI=H*FLOAT(I)
FI=FACTOR*FI*FL

```

```

50 TR=COS(FL)/HALF
C IF(I.GT.NFIL) TR=FI(I)*TR
C TI=-SIN(FL)/HALF
C IF(I.GT.NFIL) TI=FI(I)*TI
C AMP=TR*PR(I)-TI*PI(I)
C PI(I)=TR*PI(I)+TI*PR(I)
C PR(I)=AMP
C
C CONSTRUCT TABLES FOR NEW RANGE STEP.
C
C CALL SET
C
C GO TO 80
C
C THE RANGE STEP HAS NOT BEEN CHANGED.
C MULTIPLY BY STORED SECOND DERIVATIVE TRANSFORM TABLE.
C
C DC 70 I=1,NETS
C AMP=SR(I)*FE(I)-SI(I)*PI(I)
C PI(I)=SR(I)*PI(I)+SI(I)*PR(I)
C PR(I)=AMP
C
C FCURIER TRANSFORM.
C
C CONTINUE
C CALL RST(PF,N,1)
C CALL RST(PI,N,1)
C
C CHECK FOR FIAT BOTTOM.
C
C IF(IB-NW) 82,82,100
C
C THE BOTTOM IS NOT FLAT.
C SMOOTH THE TRANSITION INTO AN ISOVELOCITY REGION AND
C PUT IN THE ARTIFICIAL ABSORBING LAYER.
C
C K=1
C I=IB + 1
C
C L = 1
C
C I IS ABSCLUTE DEPTH INDEX
C K IS MESH-POINTS-IN-BOTTOM INDEX
C INX IS INDEX INTO TABULATED ATTENUATION FUNCTION
C L IS INDEX INTO FILTER FUNCTION ROLL-CFF FACTOR
C IB IS INDEX OF BOTTOM AT THIS RANGE
C NOTE THAT VOLUME ATTENUATION IS CARRIED INTO THE AUD

```

```

C      GET VOLUME ATTENUATION FACTOR
C
C      ALPHA = 1.0
C      IF (VABSF.EQ. 0.0) ALPHA = EXP(-ATTEN*DR)
C
C      GET N**2 - 1.0 AT THE WATER-MUD INTERFACE
C
C      NMD=0
C      FNO=FN(IB)
C      T=FACTOR*FNC
C      UREAL=ALPHA*COS(T)*FIL(NW)
C      UIMAG=ALPHA*SIN(T)*FIL(NW)
C      IF(NBOTM.EQ.0) GO TO 92
C
C      F1=3.1415926535/HORRAN
C      IF(NBOTM.LT.0) FN1=FN*MUD(FNO,0)
C      ZM=0.0
C      NMD=NEFFW-NW
C
C      START LCCP
C
C      85 CONTINUE
C      IF(NBOTM.LT.0) GO TO 86
C
C      USE ANALYTIC PROFILE IN MUD
C
C      ZP=ZM+DZ
C      CCR=ZM*F1
C      ANMUD=FNO-CCR*CORR
C      GO TO 90
C
C      USE USER-SPECIFIED PROFILE
C
C      86 FN2=FN*MUD(FN1,1)
C      ANMUD=0.25*(FNO+FN1+FN1+FN2)
C      FNO=ANMUD
C      FN1=FN2
C
C      90 T=FACTOR*ANMUD
C      NNN=NW+K
C      XYZ=ALPHA*FIL(NNN)
C      UREAL=XYZ*CCS(T)
C      UIMAG=XYZ*SIN(T)
C
C      CONSTRUCT U*P      (REMEMBER, THEY ARE BOTH COMPLEX)
C
C      92 CONTINUE

```

```

F3 = PR(I)*UREAL - PI(I)*UIMAG
FI(I) = PR(I)*UIMAG + PI(I)*UREAL
PR(I) = F3
I = I + 1
K = K + 1
IF (I.GT.NPTS) GO TO 94
CHECK TC SEE IF WE SHOULD ENTER ISOSPEED EXTENSION MODE
IF (K.LE.NMD) GO TO 85
NOW IN FULLY ABSORBING BOTTOM ... CUT IN FILTER FACTOR
JJJ=NEFFW+I
IF (JJJ.GT.NETS) GO TO 93
PR(I)=PR(I)*FI(JJJ)
FI(I)=PI(I)*FI(JJJ)
L=L+1
GC TO 92
93 PR(I)=0.0
PI(I)=0.0
GC TO 92
94 CONTINUE

MULTIPLY BY STORED INDEX OF REFRACTION TABLE.

DC 110 I=1,IB
AMP=UR(I)*PR(I)-UI(I)*PI(I)
FI(I)=UR(I)*PI(L)+UI(I)*PR(I)
PR(I)=AMP

RETURN
END

SET CONSTRUCTS ALL TABLES THAT ARE A FUNCTION OF THE
RANGE STEP (DR).

INPUT - DR THE CURRENT RANGE STEP
OUTPUT - A THE ARTIFICIAL ATTENUATION TABLE
S THE SECOND DERIVATIVE TRANSFORM TABLE
S = EXP(-I * DR * L**2 / (2 * K)) / (NETS / 2)
(RETURNED AS SA = REAL PART, SI = IMAGINARY PART)
U THE INDEX OF REFRACTION TABLE
U = EXP(I * K * DR * (N**2 - 1) / 2)
(RETURNED AS UR = REAL PART, UI = IMAGINARY PART)
FACTOR = DR * K / 2

```

```

WHERE I = SORT(-1)
K = AVERAGE WAVE NUMBER
N = INDEX OF REFRACTION

TEMPORARY VARIABLE - FL

SUBROUTINES - INDEX (CONSTRUCTS INDEX OF REFRACTION TABLE)

SUBROUTINE SET
COMMON /HERTZ/ CO,H,HK,F,FK,FACTOR,WL

COMMON /MESH/ R, DR, NR, KR, DZ, ZMAX, IB, N, NPTS, N2,
A NL4, NA, NW, ZW, HALF, NEFFW

COMMON/FIL1/FIL(2047)
COMMON/TABLE/ SR(2047), SI(2047), DR(2047), DI(2047), FN(2047)

GENERATE THE TABLE USED TO ATTENUATE THE ADDITIONAL DISCRETE
MODES INTRODUCED BY TRUNCATING THE FOURIER TRANSFORM. THE
PHYSICAL BOTTOM IS EXTENDED 1/4 THE MAXIMUM DEPTH AND
TERMINATED WITH A HIGHLY ABSORBING LAYER.

GENERATE L E/DZ2 TRANSFORM TABLE (INCLUDES FFT NORMALIZATION).

NFIL = (3./4.) * FLOAI(NPTS+1)
FACTOR=0.5C*DR/FK
DO 20 I=1,NPTS
FI=H*FLOAI(I)
PI=FACTOR*FI*FL
SR(I)=COS(FI)/HALF
SI(I)=-SIN(FI)/HALF
IF(I.LE.NFII) GO TO 20
SR(I)=FIL(I)*SR(I)
SI(I)=FIL(I)*SI(I)
20 CCNTINUE

GENERATE INDEX OF REFRACTION TABLE.

FACTOR=0.50*DR*FK
CALL INDEX

RETURN
END

INDEX GENERATES THE INDEX OF REFRACTION TABLE.

```

```

INPUT - FACTOR = DR * K / 2
      FN = N**2 - 1
      ATTN = VOLUME ATTENUATION

OUTPUT - U INDEX OF REFRACTION TABLE
        U=EXP(I*DR*(K*(N**2-1)/2+I*ATTN))
        (RETURNED AS UK = REAL PART, UI = IMAGINARY PART)

      WHERE I = SORT(-1)
      K = AVERAGE WAVE NUMBER
      N = INDEX OF REFRACTION

TEMPORARY VARIABLES - T, TR, TI

SUBROUTINE INDEX
  COMMON /PHASE/ NC,Z(100),C(100),MC,DM(20)
  COMMON /HERTZ/ CO,H,HK,F,FK,FACTOR,WL
  COMMON /LOSFEN/ THETI(50),LOSSI(50),NLTH,ZHAT(100),ALGEN(100),
1 HOKKAN,JA,NHAT,RAPRES,KANEXT,CHUD(30),ZHUD(30),NMUD,NBCIN
  REAL LOSSI
  COMMON /MESH/ R, DR, NR, KR, DZ, ZMAX, IB, N, NETS, N2,
A N4, NL4, NA, NW, ZW, HALF, NEFW
  COMMON /TAEIE/ SR(2047),SI(2047),UK(2047),UI(2047),FN(2047)
  COMMON /COSTR/ VABSF,ATTEN
  COMMON /FILT/FIL(2047)

  INTRODUCE VOLUME ABSORPTION/ATTENUATION

  IF(VABSF.GT.0.) ATTN=-1.
  ALPHA=1.
  IF(ATTN.GT.0.) ALPHA=EXP(-ATTN*DR)

  NWC=NPIS
  IF(NBOTM.NE.0) NWC=NW

  DC 10 I=1,NWC
  T=FACTOR*FN(I)
  AESLYR=FIL(I)
  UR(I)=ALPHA*COS(T)*AESLYR
  UI(I)=ALPHA*SIN(T)*AESLYR
10 CCNTINUE

  IF(NBOTM.EQ.0) RETURN

  INTRODUCE ATTENUATING MUD SEGMENT TO PROFILE

```

```

C      FNO=FN(NW)
C      IF(NBOTH.LT.0) FN1=FN MUD(FNO,0)
C      F1=(3.1415926535/HORRAN)
C      ZM=0.0
C
C      NWP1=NW+1
C      DC 40 I=NWP1,NEFFW
C      IF(NBOTH.LT.0) GO TO 20
C
C      USE ANALYTIC PROFILE IN MUD
C
C      ZM=ZM+DZ
C      CCRR=ZM*F1
C      ANMUD=FNO-CCRR*CORR
C      GC TO 30
C
C      USE USER-SPECIFIED PROFILE IN MUD - BUT FIRST SMOOTH PROFILE
C      WITH 1-2-1 FILTER TO TRY AND KEEP THE BUGGIES OUT
C
C      20 FN2=FN MUD(FN1,1)
C      ANMUD=0.25*(FNO+FN1+FN1+FN2)
C      FNO=ANMUD
C      FN1=FN2
C
C      BUILD INDEX TABLE WITH MUD
C
C      30 T=FACTOR*ANMUD
C      UR(I)=ALPHA*COS(T)*FIL(I)
C      UI(I)=ALPHA*SIN(T)*FIL(I)
C      40 CCNTINUE
C
C      COMPLETE TABLE BY USING COSTANT SOUND SPEED GRADIENT
C
C      NEFP1=NEFFW+1
C      UFC=UR(NEFFW)
C      UIC=UI(NEFFW)
C
C      DC 50 I=NEFP1,NPTS
C      UR(I)=URC*FIL(I)
C      UI(I)=UIC*FIL(I)
C      50 CCNTINUE
C      RETURN
C      END
C
C      SUBROUTINE FILLOS
C      INTRODUCE BOTTCM LOSS TO THE FILTER FUNCTION

```

```

C      COMMON /FIL/ FIL(2047)
C      COMMON /MESH/ R,DR,NR,KR,DZ,ZMAX,IB,N,NPTS,N2,N4,NL4,NA,NW,ZW,
1      HALF,NEFFW
C      COMMON /LOSFCN/ THETI(50),LOSSI(50),NLTH,ZHAT(100),ALGEN(100),
1      HORRAN,JA,NHAT,RAPRES,RANEXT,CMUD(30),ZMUD(30),NM6D,NBOTM
C      REAL LCSSI
C
C      INITIALLY DETERMINE LOSS
C
C      Z1 = 0.0
C      K = 1
C      NWP1 = NW + 1
C      FIL(NW) = 0.0
C      DC 30 I = NWP1,NEFFW
C      Z1 = Z1 + DZ
10  IF(K.GE.NHAT) GO TO 25
C      IF( (Z1.GE.ZHAT(K)) .AND. (Z1.LE.ZHAT(K+1))) GC TO 20
C      K = K + 1
C      GO TO 10
C
20  SLOPE = (ALGEN(K+1)-ALGEN(K))/(ZHAT(K+1)-ZHAT(K))
C      AIZ = ALGEN(K) + SLOPE*(Z1-ZHAT(K))
C      FIL(I) = AIZ*Z1
C      GC TO 30
25  FIL(I) = FIL(I-1)
30  CCNTINUE
C
C      MAKE SURE LOSS IS A RELATIVELY SMOOTH FUNCTION OF DEPTH
C      BY APPLYING A 1-2-1 FILTER
C
C      NEWM1 = NEFFW - 1
C
C      DC 40 I = NWP1,NEWM1
C      FIL(I) = 0.25*(FIL(I-1) + FIL(I) + FIL(I) + FIL(I+1))
40  CCNTINUE
C
C      CONVERT TO ATTENUATION IN PRESSURE
C
C      DC 50 I = NW,NEFFW
C      FIL(I) = EXP(-(2.302585/20.0)*FIL(I))
50  CCNTINUE
C
C      RETURN
C      END
C
C      FUNCTION FMUD(FNE,K)

```

```

C      DETERMINE MODIFIED INDEX (N**2-1) IN BOTTOM ...
C      GIVEN USER-SPECIFIED PROFILE
C
COMMON /MESH/ R,DR,NR,KR,DZ,ZMAX,IB,N,NPTS,N2,N4,NL4,NA,NW,ZW,
1 HALF,NEFFW
COMMON /LOSSCN/ THETI(50), LOSSI(50), NLTH,ZHAT(100), ALGEN(100),
1 HOKRAN,JA,NHAT,RAPRES,RANEXT,CHUD(30),ZMUD(30),NMUD,NBOTM
REAL LCSSI
COMMON /HEATZ/ CO,H,HK,F,FK,FACTOR,WL
COMMON /JUNK/ M,COLD,Z1
C
IF (K.GT.0) GO TO 10
C
M=1
Z1=0.0
CCLD = CO / SQRT(FNP+1.0)
C
10 Z1=Z1+DZ
20 IF((Z1-GE.ZMUD(M)).AND.Z1.LE.ZMUD(M+1)) GC TC 30
CCLD=CCLD+(CMUD(M+1)-CMUD(M))
M=M+1
IF(M.LT.NMUD) GO TO 20
CNEW=CCLD
GO TO 40
C
30 DELZ = (Z1-ZMUD(M)) / (ZMUD(M+1)-ZMUD(M))
CNEW = COLL + DELZ*(CMUD(M+1)-CMUD(M))
C
40 FNN=CO/CNEW
FNMUD=FNN*FNN-1.0
RETURN
END
C
C      FUNCTION TICSS INTERPOLATES THE FIELD AT DEPTH Z AND
C      RETURNS THE TRANSMISSION LOSS.
C
C      INPUT - RF RECIPIROCAL RANGE
C              Z DEPTH
C              DZ MESH INCREMENT
C              PR REAL PART OF FIELD MESH
C              PI IMAGINARY PART OF FIELD MESH
C
C      OUTPUT - TICSS TRANSMISSION LOSS AT DEPTH Z
C      FUNCTION TICSS(R,Z)
C
COMMON /FIELD/ PR(2049),PI(2049)
C
C

```

```

C
C
A COMMON /MESH/ R , DR , NH , KR , DZ , ZMAX , IB , N , NETS , N2 ,
  N4 , NL4 , HA , NW , ZW , HALF , NEFFW
C
C DATA PMIN/1.0E-18/
C
C ZM=Z/DZ
C M=ZM
C PZ=0.
C IF (M.GT.0) PZ=SQRT(PR(M)*PR(M)+PI(M)*PI(M))
C PZ=PZ+(SQRT(PR(N+1)*PR(N+1)+PI(N+1)*PI(N+1)))-EZ)*(ZM-FLCAT(M))
C TLOSS=-10.*ALOG10(RR*PZ*PZ+PMIN)
C RETURN
C END
C FUNCTION SPEED(D)
C
C SPEED RETURNS THE SOUND SPEED AT DEPTH = D.
C
C INPUT - NC NUMBER OF POINTS IN SOUND VELOCITY PROFILE TABLE.
C         Z DEPTH ARRAY.
C         C SOUND SPEED ARRAY.
C         I INPUT DEPTH.
C
C OUTPUT - SPEED SOUND SPEED AT DEPTH D.
C
C COMMON /PHASE/ NC,Z(100),C(100),MC,DM(20)
C
C K=2
C
C IF (D.LT.Z(K)) GO TO 20
C K=K+1
C IF (K.LT.NC) GO TO 10
C SPEED=C(K-1)+(C(K)-C(K-1))*(D-Z(K-1))/(Z(K)-Z(K-1))
C
C RETURN
C END
C
C FILTER EVALUATES THE INDEX OF REFRACTION ON THE FIELD MESH AND
C TRANSFORMS THE ENVIRONMENT TO REDUCE THE PARAOLOIC PHASE
C VELOCITY ERROR.
C
C INPUT - NC NUMBER OF POINTS ON SOUND VELOCITY PROFILE
C         Z DEPTH ARRAY (NC DEPTHS)
C         C SOUND SPEED ARRAY (NC SOUND SPEEDS)
C         CC REFERENCE SOUND SPEED
C         MC PHASE VELOCITY CORRECTION FLAG
C         I OUTPUT DEPTH ARRAY

```

```

C      OUTPUT - FN      SMOOTHED INDEX OF REFRACTION ARRAY
C      DM      TRANSFORMED OUTPUT DEPTH ARRAY
C      CD      TRANSFORMED FIELD PLOT DEPTH ARRAY
C
C      LOCAL VARIABLES - ZM  DEPTH (FT) AT FIELD MESH POINT
C                      CM  INDEX OF REFRACTION AT FIELD MESH POINT
C                      G   SOUND SPEED GRADIENT
C
C      CCNSTANTS - FT  CCVERSION FACTOR METERS/FT
C
C      SUBROUTINE FILTER (ND,D)
C      DIMENSION F(20)
C      COMMON /UNITS/ LC,LP,LT
C      COMMON /HEIGHT/ CO,H,HK,F,FK,FACTOR,WL
C      COMMON /PHASE/ NC,Z(100),C(100),MC,DM(20)
C      COMMON /PLT/TITLE(20),NPLT,LCR,CLMIN,DCL,CD1,CD2,DCD,CD(120)
C
C      COMMON /MESH/ R,DR,NE,KK,DZ,ZMAX,IB,N,NPTS,N2,N4,NL4,NA,NW,ZW,
C      1 HALF,NEFFK
C
C      COMMON /TABLE/ SR(2047),SI(2047),UR(2047),UI(2047),FN(2047)
C      DATA FT/0.3048/
C
C      IF (Z(1).EQ.0.) GO TO 10
C
C      WRITE (LP,9CC)
C      FCMAT(32H01INVALID SOUND VELOCITY PROFILE./
C      40H SOUND SPEED NOT DEFINED AT THE SURFACE.)
C      1 STOP
C
C      CHECK INPUT PROFILE UNITS.
C
C      IF (C(1).GT.3000.) GO TO 30
C
C      CONVERT UNITS TO FT AND FT/SEC.
C
C      DC 20 I=1,NC
C      Z(I)=Z(I)/FT
C      C(I)=C(I)/FT
C
C      IF (NC.GE.2) GO TO 40
C      NC=2
C      Z(2)=DZ*FLCAT(NW)
C      C(2)=C(1)
C
C      INTERPOLATE N**2 - 1 ON THE FIELD MESH.

```

```

40      K=0
      L=NC-1
      ZF=0.
      ZK=DZ*FLOAT(NW)
      IF (MC.EQ.1) ZM=ZM*SQRT(CO/SPEED(ZM))
      ASSIGN 45 TC LOOP
      C
45      I=1
      ZM=DZ*FLOAT(I)
      IF (ZM.LE.2W) GO TO 50
      ASSIGN 70 TC LOOP
      GO TO 70
50      IF (ZM.LT.2L.OR.K.EQ.L) GO TO (1,2),NC
      K=K+1
      G=(C(K+1)-C(K))/(Z(K+1)-Z(K))
      ZP=Z(K+1)
      IF (MC.EQ.1) ZP=ZP*SQRT(CO/C(K+1))
      GC TO 50
      CM=G*2W*ZM
      ZM=(CM+ZM*SQRT(G*CM+4.*CO*(C(K)-G*Z(K))))/(CO+CO)
      CM=CO/(C(K)+G*ZM-Z(K))
      IF (MC.EQ.1) GO TO 60
      CP=CM*CM-1.6
      GO TO 70
      CM=CM+CM-2.0
      FN(I)=CM
      I=I+1
      IF (J.LE.NPTS) GO TO LOOP,(45,70)
      C
      C
      C
      SMOOTH THE MESH WITH A 1-2-1 FILTER.
      C
      L=NPTS-1
      FN(1)=0.25*(FN(1)+FN(1)+FN(1)+FN(2))
      DO 90 I=2,L
      FN(I)=0.25*(FN(I-1)+FN(I)+FN(I)+FN(I+1))
      IF (MC.EQ.2) GO TO 100
      C
      C
      C
      TRANSFORM THE OUTPUT AND FIELD PLOT DEPTHS.
      DO 90 I=1,NL
      CM(I)=D(I)*SQRT(CO/SPEED(D(I)))
      IF (NPLT.LE.0) GO TO 100
      C
      C
      ZM=CD1
      DO 95 I=1,NPLT
      CD(I)=ZM*SQRT(CO/SPEED(ZM))

```



```

CCCONTINUE
C
C
C
C
RETURN
END
C
FID IS A LINE PRINTER FIELD PLOTTING ROUTINE. SYMBOLS
CCRRESPONDING TO LCL DB TRANSMISSION LOSS BINS AT UP
TO 120 DEPTH MESH POINTS FROM DEPTH CD1 TO DEPTH CD2 ARE PRINTED
AT APPROXIMATELY A 1 NAUTICAL MILE RANGE INCREMENT. A DEPTH
SCALE AND A TRANSMISSION LOSS SCALE ARE PRINTED THE FIRST TIME
FID IS CALLED.
C
INPUT - NPIT NUMBER OF PLOT DEPTHS (.LE. 120)
CD1 MINIMUM PLOT DEPTH
DCD PLOT DEPTH INCREMENT
CLMIN MINIMUM LOSS
DCI LOSS INCREMENT
CD PLOT DEPTH ARRAY
ZW WATER DEPTH
R CURRENT RANGE STEP
DR CURRENT RANGE
RR RECEIVED RANGE
LOCAL VARIABLES - KCR CURRENT RANGE
NUM NUM SYMBOLS FOR PLOTTING DEPTHS
LEVEL LEVEL PLOTTING SYMBOL ARRAY
PL PL TRANSMISSION LOSS AT DEPTH CD(I)
CL CL CURRENT PLOT LOSS BIN
SUBROUTINE - FUNCTION TLOSS (INTERPOLATE FIELD AND RETURN
TRANSMISSION LOSS)
C
SUBROUTINE FLD(RR)
INTEGER TITLE
DIMENSION ILEV(6), NUM(10), LOSS(120)
COMMON /FIELD/ PR(2049), PI(2049)
COMMON /UNITS/ LC LP IT
COMMON /OUTEUF/ NCUT,RNM,TL(20)
COMMON/PLT/TITLE(20),NPLT,LCR,CLMIN,DCL,CD1,CD2,DCD,CD(120)
COMMON /MESH/ N4 , DR , NR , NA , NW , ZW , HALF , NEFFW , N2 ,
A
DATA FNM,LEVEL/0070,1,1H*,1H/,1H*,1H0,1HB/
DATA NUM/1EC,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/

```

81

```

C      KCR=(R+0.50*DR)/FNM
C      IF (KCR.EQ.ICR) RETURN
C      CHECK FOR FIRST CALL.
C      IF (LCR.GT.C) GO TO 40
C      WRITE (LP,9CG) TITLE
C      FCRMAT (1H1,20A4)
C      PRINT TRANSMISSION LOSS SCALE.
C      K=4
C      DC 2 I=1,5
C      DO 1 J=1,1C
C      K=K+1
C      LOSS(K)=LEVEL(I)
C      CCNT INUE
C      CI=CLMIN
C      DC 3 I=1,4
C      LOSS(I)=CL+C.50
C      IF (LOSS(I).LT.100) LOSS(10*I+5)=LEVEL(I)
C      CI=CI+DCL
C      WRITE (LP,910) (LOSS(I),I=1,K)
C      FCRMAT (1H0,21X,8HTL SCALE/11X,4(I3,3H DB,4X)/1X,50A1)
C      PRINT DEPTH SCALE.
C      WRITE (LP,915)
C      FCRMAT (1H0,22X,10HDEPTH (FT))
C      L=10000
C      DO 30 I=1,5
C      K=NPLT+1
C      DC 20 J=1,NFLT
C      F=K-1
C      IZ=CD1+DCD*FLOAT(J-1)
C      IF (L.EQ.1-CR.IZ-GE.L) GO TO 10
C      LOSS(K)=LEVEL(3)
C      GO TO 20
C      KZ=1+IZ/L-10*(IZ/(10*L))
C      LOSS(K)=NUM(KZ)

```

```

20 CONTINUE
30 I=L/10
40 WHITE(LP,920) (LOSS(K),K=1,NPLT)
50 FCRMAT(9X,120A1)
60 C
70 WRITE(LP,930)
80 FCRMAT(2X,5HRANGE,2X,120(1H-))
90 C
100 PRINT TRANSMISSION LOSS SYMBOLS AT PLOT DEPTHS.
110 C
120 ICK=KCR
130 K=NPLT+1
140 C
150 C
160 C
170 C
180 C
190 C
200 C
210 C
220 C
230 C
240 C
250 C
260 C
270 C
280 C
290 C
300 C
310 C
320 C
330 C
340 C
350 C
360 C
370 C
380 C
390 C
400 C
410 C
420 C
430 C
440 C
450 C
460 C
470 C
480 C
490 C
500 C
510 C
520 C
530 C
540 C
550 C
560 C
570 C
580 C
590 C
600 C
610 C
620 C
630 C
640 C
650 C
660 C
670 C
680 C
690 C
700 C
710 C
720 C
730 C
740 C
750 C
760 C
770 C
780 C
790 C
800 C
810 C
820 C
830 C
840 C
850 C
860 C
870 C
880 C
890 C
900 C
910 C
920 C
930 C
940 C
950 C
960 C
970 C
980 C
990 C

```



```

C C C C
      SET UP THE ARRAY FOR THE COSINE TRANSFORM
      B(1) = X(1)
      B(2) = X(NP+1)
      J1=0
      DC 20 J=3, NEM1,2
      J1=J1+1
      J2=JI(J1)
      J3=NP-J2
      B(J) = X(J2+1)
      B(J+1) = X(J3+2) - X(J3)
      GC TO 45
20
C C C C
      SET UP ARRAY FOR THE SINE TRANSFORM
      CCNTINUE
      E(1) = -X(1) - X(1)
      B(2) = X(NPM1) + X(NPM1)
      J1=0
      DC 40 J=3, NEM1,2
      J1=J1+1
      J2=JI(J1)
      J3=NP-J2
      B(J) = X(J2-1) - X(J2+1)
      B(J+1) = X(J3)
      CCNTINUE
40
45
C C C C
      BEGIN FAST FOURIER SYNTHESIS
      IF (N8.EQ.0) GO TO 60
      RADIX 8 ITEFATIONS
      INT=1
      NT=NP/16
      DC 50 J=1, N8
      J1= 1+INT
      J2=J1+INT
      J3=J2+INT
      J4=J3+INT
      J5=J4+INT
      J6=J5+INT
      J7=J6+INT
      CALL R8SYN(INT,B,E(J1),B(J2),B(J3),B(J4),B(J5),B(J6),B(J7))
      NT=NT/8

```

```

50 INT=8*INT
60 CCNTINUE
70 IF (N4) 90,80,70
80 RADIX 4 ITERATION
90 INT=NPD4
100 J1=1+INT
110 J2=J1+INT
120 J3=J2+INT
130 CALL K4SYN(INT,B,E(J1),B(J2),B(J3))
140 GC TO 90
150 RADIX 2 ITERATION
160 INT=NPD2
170 J1=1+INT
180 CALL R2TR(INT,B,B(J1))
190 CONTINUE
200 J1=NP
210 IF (IFLAG.GT.0) GO TO 95
220 FORM THE CCSINE TRANSFORM
230 DO 94 J=1,NM1
240 X(J)=.25*((E(J1)-B(J+1))*ST(J)+B(J+1)+B(J1))
250 J1=J1-1
260 CCNTINUE
270 RETURN
280 FORM SINE TRANSFORM
290 CCNTINUE
300 DO 100 J=1,NPM1
310 X(J)=0.25*((B(J+1)+B(J1))*ST(J)-B(J+1)+B(J1))
320 J1=J1-1
330 RETURN
340 COMPUTE CONSTANTS AND CONSTRUCT TABLES
350 N2=N
360 N8=N2/3
370 N4=N2-3*N8-1

```

```

C
NF=2**N2
NPD2=NF/2
NPD4=NF/4
NPD16=NF/16
NPM1=NF-1
DT=PI/FLOAT(NP)
DO 210 J=1,NPM1
T=DT*FLOAT(J)
ST(J)=0.50/SIN(T)
C
C
C
CONSTRUCT THE BIT REVERSED SUBSCRIPT TABLE.
J1=0
NT=NPD2-1
DC 240 J=1,NT
J2=NPD2
IF (AND(J1,J2).EQ.0) GO TO 230
J1=IABS(J1-J2)
J2=J2/2
GC TO 220
J1=J1+J2
JI(J)=J1
C
C
C
IF (N8.EQ.C) GO TO 10
CONSTRUCT THE TRIGONOMETRIC TABLES FOR THE RALIX 8 PASSES.
THE TABLES ARE STORED IN BIT REVERSED ORDER.
J1=0
N1=NPD16-1
DC 270 J=1,NT
J2=NPD16
IF (AND(J1,J2).EQ.0) GO TO 260
J1=IABS(J1-J2)
J2=J2/2
GC TO 250
J1=J1+J2
T=DT*FLOAT(J1)
CS(J)=COS(T)
SS(J)=-SIN(T)
C
GC TO 10
C
END
C
SUBROUTINE R8SYN(INT,B0,B1,B2,B3,B4,B5,B6,E7)
C
C
C

```

```
C C C
RADIO 8 SYNTHESIS SUBROUTINE
CALLED BY RST, THE SINE TRANSFORM DRIVER.

DIMENSION EC(2), B1(2), B2(2), B3(2), B4(2), B5(2), B6(2), B7(2)
COMMON /WTS/, NT, CS{128}, SS{128}
DATA R2,CPI4/1.414213562373106, 70710678118655/
DATA CPI8,SPI8/0.92387953251129, 0.38268343236509/

JT=0
JI=2
JR=2
JL=3
INT8=8*INT

DO 72 K=1, INT
T0=B0 {K} +B1 {K}
T1=B0 {K} -B1 {K}
T2=B2 {K} +B2 {K}
T3=B3 {K} +B3 {K}
T4=B4 {K} +B6 {K}
T5=B4 {K} -B6 {K}
T6=B7 {K} -B5 {K}
T7=B7 {K} +B5 {K}
T8=R2*(T7-T5)
T5=R2*(T7+T5)
T10=T0+T2
T2=T0-T2
T11=T1+T3
T3=T1-T3
T4=T4+T4
T6=T6+T6
B0 {K} =T0+T4
B4 {K} =T0-T4
B1 {K} =T1+T5
B5 {K} =T1-T5
B2 {K} =T2+T6
B6 {K} =T2-T6
B3 {K} =T3+T8
B7 {K} =T3-T8
C CNTINUE
72 CCNTINUE

IF (NT.EQ.0) GO TO 70
K0=INT8+1
KLAST=K0+INT-1
DO 75 K=K0,KLAST
T1=D0 {K} +B0 {K}
```

```

T3=B0 (K) -B6 (K)
T2=B7 (K) -B1 (K)
T4=B7 (K) +B1 (K)
T5=B2 (K) +B4 (K)
T7=B2 (K) -B3 (K)
T6=B5 (K) +B3 (K)
T8=B5 (K) +B3 (K)
B0 (K) = (T1+T5) + (T1+T5)
B4 (K) = (T2+T6) + (T2+T6)
T5=T1-T5
T6=T2-T6
B2 (K) = R2* (T6+T5)
B6 (K) = R2* (T6-T5)
T1=T3*CP18-T4*SPI8
T2=T4*CP18-T3*SPI8
T3=T8*CP18-T7*SPI8
T4=-T7*CP18-T8*SPI8
B1 (K) = (T1+T3) + (T1+T3)
B5 (K) = (T2+T4) + (T2+T4)
T3=T1-T3
T4=T2-T4
B3 (K) = R2* (T4+T3)
B7 (K) = R2* (T4-T3)
75 CONTINUE

```

89

C

76

GC TO 70

```

C1=CS (JT)
S1=SS (JT)
C2=C1*CI-S1*S1
S2=C1*S1+CI*S1
C3=C1*C2-S1*S2
S3=C1*S2+CI*S2
C4=C2*CI-S2*S2
S4=C2*S2+CI*S2
C5=C2*C3-S2*S3
S5=C2*S3+CI*S3
C6=C3*CI-S3*S3
S6=C3*S3+CI*S3
C7=C3*C4-S3*S4
S7=C3*S4+CI*S4

```

C

```

K=JI*INT8
J0=JR*INT8+1
JLAST=J0+INT-1
DO 77 J=J0,JLAST
K=K+1

```

C

90

```

B6(J)=C6*{TR2-TTR6}-S6*{TI2-TI6}
P6(K)=C6*{TI2-TI6}+S6*{TR2-TTR6}
B7(J)=C7*{TR3-TTR7}-S7*{TI3-TI7}
E7(K)=C7*{TI3-TI7}+S7*{TR3-TTR7}
77 CCNTINUE

JK=JR+2
JI=JI-2
IF (JI-GT-JI) GO TO 70
JI=JR+JR-1
JI=JR
JT=JT+1
IF (JT.LT.NT) GO TO 76

RETURN
END

SUBROUTINE K4SYN(INT,B0,B1,B2,B3)
RADIX 4 SYNTHESIS SUBROUTINE
CALLED BY EST, THE SINE TRANSFORM DRIVER.
DIMENSION E0(2),B1(2),B2(2),B3(2)

DC 200 K=1,INT
T0=B0(K)+B1(K)
T1=B0(K)-B1(K)
T2=B2(K)+B3(K)
T3=B2(K)-B3(K)
E0(K)=T0+T2
E2(K)=T0-T2
B1(K)=T1+T3
E3(K)=T1-T3
CCNTINUE

RETURN
END

SUBROUTINE K2TR(INT,B0,B1)
RADIX 2 TRANSFORM SUBROUTINE
CALLED BY EST, THE SINE TRANSFORM DRIVER.
DIMENSION E0(2),B1(2)

DC 100 K=1,INT
T=B0(K)+B1(K)
E1(K)=E0(K)-B1(K)

```

```

100      BO(K)=T
C      CONTINUE
C
C      RETURN
C      END
C
C      FUNCTION ZE(R)
C
C      ZE RETURNS THE BOTTOM DEPTH AT RANGE R.
C
C      COMMON /BATHY/ RE,KE,NB,BR(101),BZ(101)
C
C      IF (R.LT. ER(KB+1)) GC TO 20
C      KP=KB+1
C      ES={BZ(KB+1)-BZ(KB)}/(BR(KB+1)-BR(KB))
C      GC TO 10
C
C      ZE=BZ(KB)+ES*(R-BR(KE))
C
C      RETURN
C      END
C
C      SUBROUTINE SVP(NC,Z,C,RNEXT)
C
C      COMMON/UNITS/LC,LP,LT
C
C      DIMENSION Z(2),C(2)
C
C      READ(2)NC,Z(1),C(1),I=1,NC
C      READ(2)RNEXT
C
C      RETURN
C      END
C
C*****
C      THIS ROUTINE GENERATES THE ATTENUATION FUNCTION ALPHA(Z)
C      ON A MESH ZHAT GIVEN THE FOLLOWING
C      A TABULATED I(THETA) CURVE AS GIVEN IN LOSSI(), THETI()
C      THE HORIZONTAL RANGE PERIOD HORRAN
C      A SOUND SPEED PROFILE OF INVERSE PARABOLIC FORM DERIVED
C      SUCH THAT THE RAY PERIOD IN THE MEDIUM IS
C      INDEPENDANT OF RAY GRAZING ANGLE.
C
C      THE FUNCTION ALPHA IS CONSTRUCTED AT MESH POINTS GIVEN
C      BY ZHAT(). ALPHA IS REPRESENTED BY A PIECEWISE
C      CONTINUOUS FUNCTION IN DEPTH, WITH THE COEFFICIENTS

```

```

C*****
C      IN EACH LINEAR SEGMENT GIVEN IN CI() AND DI().
C*****
C      SUBROUTINE LOSGEN
COMMON /LOSFCN/ THETI(50), LOSSI(50), NLTH, ZHAT(100), ALGEN(100),
A HORRAN, JA, NHAT, RAPRES, KANEXT, CAUD(30), ZMUD(30), NMUD, NBCIM
REAL LCSSI

DIMENSION T1(100), T2(100), CI(100), DI(100)
REAL LOFZ, INTLG1, INIEG2
COMMON /HERTZ/ COH, HK, F, FK, FACTOR, WL
COMMON /UNITS/ LC, LP, IT
CCNA = HORRAN / 3.1415926
GMAX = -1000.0
IF(LOSSI(1).GT.0.0) GMAX=AMAX1(GMAX, LOSSI(1)/(180.0*HK/3.1415926))
DC 5 J = 2, NLTH
GFAD = (LOSSI(J)-LOSSI(J-1)) / (THETI(J)-THETI(J-1))
GMAX = AMAX1(GMAX, GFAD)
5 CCONTINUE
IF(GMAX.GT. 1.0) WRITE(LP,6)
IF(GMAX.GT. 1.0) STOP
6 FCIMAT(80) MAXIMUM LOSS GRADIENT HAS EXCEEDED 1.0 DE/DEGREE. FE CO
XDE IS UNABLE TO HANDLE
MAXIMUM ZHAT AND INCREMENT
NHAT = 1
ZHAT(1) = 0.0
ZINC = 250.0
10 CONTINUE
NHAT = NHAT + 1
ZHAT(NHAT) = ZHAT(NHAT-1) + ZINC
IF(ZHAT(NHAT).LT. (HORRAN/5.0)) GO TO 10
IF(ZHAT(NHAT).LT. (HORRAN/5.0)) GO TO 10
GO TO 10
DC 12 I = 1, 100
CI(I) = 0.0
DI(I) = 0.0
IT(I) = 0.0
T2(I) = 0.0
12 CCONTINUE
INSERT ZHAT POINTS CORRESPONDING TO TABULATED L(THETA)
DC 130 I = 1, NLTH
THETI(I) = THETI(I)
ZTHI = HORRAN*SIN(3.1415926*THETI/180.) / 3.1415926
NM1 = NHAT-1
LC 110 J = 1, NM1
IF(ZTHI.GT. ZHAT(J)) ZHAT(J) = ZTHI
GO TO 110
105 REMOVE = NHAT - (J+1) + 1

```

```

DC 106 L = 1 , LMCVE
M = NHAT + 2 - L
ZTHAT(M) = ZTHAT(M-1)
106 CCNTINUE
ZTHAT(M-1) = ZTHI
NHAT = NHAT + 1
GC TO 130
110 CCNTINUE
130 CCNTINUE
C START LCC2 OVER TABULATED POINTS IN FINAL MESH
J=2
300 CCNTINUE
IFLAG = 0
301 CCNTINUE
ZTHJ = ZTHAT(J)
ALOSZJ = LCC2 (THE T1, LOSSI, NLTH, ZHJ, HORRAN) - LOSSI(1)
C START LCC2 ON PARTIAL INTEGRALS
JM1 = J - 1
DO 20 K = 1, JM1
T1(K+1) = INTEG1 { ZTHAT(K), ZTHAT(K+1), ZHJ, CONA }
T2(K+1) = INTEG2 { ZTHAT(K), ZTHAT(K+1), ZHJ, CONA }
20 CCNTINUE
AIHS = ALOSZJ
IF (J .LE. 2) GO TO 26
C SUM OVER ALL SEGMENTS BUT THE LAST ONE
LIMU = J - 1
DC 24 I = 2 , LIMU
ALHS = ALHS - DI(I) * T1(I) - CI(I) * T2(I)
24 CCNTINUE
ALHS = ALHS - (CI(J-1) + DI(J-1) * ZTHAT(J-1)) * T2(J)
26 CCNTINUE
DI(J) = ALHS / (T1(J) - ZTHAT(J-1) * T2(J))
CI(J) = CI(J-1) + DI(J-1) * ZTHAT(J-1)
C COMPUTE ALPHA ON INTERVAL BOUNDING POINTS
BLO = DI(J) * ZTHAT(J-1) + CI(J)
BHI = DI(J) * ZTHAT(J) + CI(J)
IF (BLO .GE. C.C.0 .AND. BHI .GE. 0.0) GO TO 303
C ALPHA WENT NEGATIVE ON THIS INTERVAL... INSERT A POINT INTO
C ZTHAT BY BISECTING THE CURRENT INTERVAL. IFLAG IS A BISECTION
C COUNTER FOR THIS INTERVAL
ZHN = (ZTHAT(J) + ZTHAT(J-1)) / 2.0
IF (IFLAG .GT. 0) GO TO 304
NHAT = NHAT + 1
K = NHAT
NEOVE = NHAT - J
DC 300 M = 1 , RMCVE
ZTHAT(K) = ZTHAT(K-1)

```

```

306 K = K - 1
307 CCNTINUE
308 ZHAT(J) = ZLN + 1
309 IFLAG = IFLAG + 1
310 CHECK FCF EXCESSIVE BISECTION
311 IF (IFLAG.LT.6) GO TO 301
312 WRITE(LP,307)
313 STOP
314 FORMAT(85HCEOTOM ATTENUATION CALCULATION TERMINATED .... UNABLE TO
315 X REPRESENT ALPHA AS LINEAR
316 CCNTINUE
317 IFLAG = 0
318 J = J + 1
319 IF (J.LE.NHAT) GO TO 300
320
321 COMPUTE ALPHA(Z) AT THE TABULATED POINTS ZHAT(I)
322 JA WILL BE INDEX OF MAXIMUM ALPHA
323 DC 310 J = 2.0 NHAT
324 ALGEN(J) = 0.5 * (CI(J) + DI(J)*ZHAT(J) )
325 CCNTINUE
326 IF (LOSSI(1).LE.0.0) RETURN
327 DC 311 I=1,NHAT
328 ALGEN(I)=ALGEN(I) + (LOSSI(1)/HORRAN)
329 CCNTINUE
330 RETURN
331 END
332
333 REAL FUNCTION INTEG1(ALPHA,BETA,ZHAT,A)
334
335 F(X) = (SIN(X)/COS(X)**2) + ALOG((1.0+SIN(X))/CCS(X))
336
337 EVALUATE THE INTEGRAL RESULTING FROM SOLUTION OF EQUATION
338 RELATING LOSS(THETA) AND ATTENUATION ALPHA(Z)
339
340 F1 = ALPHA**2
341 F2 = BETA**2
342 F3 = ZHAT**2
343 F4 = SQRT(F3-F1)/A
344 F5 = SQRT(F3-F2)/A
345 X10 = ATAN(F4)
346 X11 = ATAN(F5)
347 INTEG1 = A*(F(X11)-F(X10))
348 RETURN
349 END
350
351 REAL FUNCTION INTEG2(ALPHA,BETA,ZHAT,A)

```

~~AD-A143 079~~

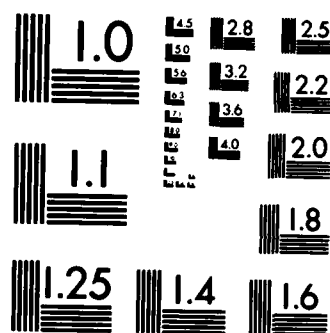
A COMPARISON OF TWO ACOUSTIC PARABOLIC TRANSMISSION
LOSS MODELS FOR COMP. (U) NAVAL POSTGRADUATE SCHOOL
MONTEREY CA J L BLANCHARD MAR 84

2/2

UNCLASSIFIED

F/G 20/1

NL



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```

C C C C C
EVALUATE THE INTEGRAL RESULTING FROM SOLUTION OF EQUATION
RELATING LOSS(THETA) AND ATTENUATION ALPHA(Z)

F1 = ZHAT**2
F2 = BETA**2
F3 = ALPHA**2
F4 = A**A
F5 = F1 + F4
F6 = SQRT(F5)
DEN1 = AMAX1(1.0E-10, SQRT(F1-F2))
DEN2 = SQRT(F1-F3)
CALL ELI2(RESU, BETA/DEN1, A/F6, 1.0, F4/F5)
CALL ELI2(RESL, ALPHA/DEN2, A/F6, 1.0, F4/F5)
INTEG2 = 2.0 * F6 * (RESU - RESL)
RETURN
END

C
REAL FUNCTION LOFZ(IHETI, LOSSI, NI, ZHAT, HOREAN)
REAL THETI(NI), LOSSI(NI)

EVALUATE THE TABULATED LOSS FUNCTION L(THETA) AT AN INCIDENT
ANGLE THETA CORRESPONDING TO THE TURNING DEPTH ZHAT
COMPUTE THETA(Z) BY USING FUNCTIONAL FORM CF PROFILE AND SNELLS
LAW.

WE GET A RELATION FOR THE INCIDENT ANGLE THETA(PARABOLIC)
BY COMBINING THE FUNCTIONAL FORM OF THE PROFILE AND THE SNELLS
LAW BOTH IN THE PARABOLIC APPROXIMATION. HOWEVER, THE LOSS
FUNCTION L(THETA) IS GIVEN AS A FUNCTION OF THETA(ELLIPTIC) ...
THE PHYSICAL ANGLE. CONVERT VIA THE FOLLOWING

$$\sin(\text{THETA-ELLIPTIC}) = \tan(\text{THETA-PARABOLIC})$$


THETZ = (180.0/3.1415926) * ARCSIN(3.1415926*ZHAT/HCRAN)
USE TABLE LOOKUP AND LINEAR INTERPOLATION IN THE GIVEN
LOSS FUNCTION. INDEPENDANT VARIABLE THETA IS INCIDENT ANGLE
DC 10 I = 2, NI
IF ( THETZ - GT. THETI(I) ) GO TO 10
I = LOSSI(I-1) + (LOSSI(I) - LOSSI(I-1)) * (THETZ - THETI(I-1))
A = THETI(I) - THETI(I-1)
RETURN
10 CC CONTINUE
ERROR EXIT ... ANGLE IS NOT IN TABULATED ANGLE RANGE
WRITE(6,11) THETZ, THETI(1), THETI(NI)
11 FORMAT(41HLOSS FUNCTION ANGLE NOT IN TABLE RANGE ,3F10.3)
STOP 007
END
C C C C C

```

```

*****
PURPOSE ... COMPUTE THE GENERALIZED ELLIPTIC INTEGRAL OF THE
SECOND KIND

USAGE ... CALL ELI2(R,X,CK,A,B)

DESCRIPTION OF PARAMETERS
R      UPPER INTEGRATION BOUND
X      UPPER INTEGRATION BOUND (1.0-K*K)
CK     COMPLEMENTARY MODULUS
A,E    CONSTANTS IN THE INTEGRAND
SPECIAL CASE .. E(AIAN(X),K) OBTAINED WITH A = 1.0, B = CK*CK

SUBROUTINES REQUIRED ... NONE

METHOD   SERIES SUMMATION WITH LANDEN TRANSFORMATION

SEE IBM LIBRARY SSP
*****
TEST ARGUMENT
SUBROUTINE ELI2(R,X,CK,A,B)
IF(X) 2,1,2
1 R = 0.0
2 C = 0.0
3 D = 0.5
4 IF(CK) 7,3,7
5 R = (A-B)*ABS(X)/R+B*ALOG(ABS(X)+h)
6 TEST SIGN OF ARGUMENT
7 R = R+C*(A-E)
8 IF(X) 5,6,6
9 R = R
10 RETURN
11 TIALIZATION
12 AN = (E+A)*0.5
13 AA = A
14 RANG = ABS(1.0/X)
15 RANG = 0.0
16 RANG = 0
17 RANG = 1.0
18 GEO = ABS(CK)
19 LANDEN TRANSFORMATION
C

```

```

8 R = AA * GEC + R
  SGEO = ARI * GEO
  AA = AN
  AARI = ARI
  ARI = ARITHMETIC MEAN
  ARI = GEO + ARI
  SUM OF SINE VALUES
  AN = (R/ARI+AA)*0.5
  AANG = ABS(ANG)
  ANG = -SGEC / ANG + ANG
  PIM = PIM
  PIMF(ANG) 10 9 11
  IF ANG = -1.0E-8*AANG
10 FIM = FIM + 3.14159265
  ISI = ISI + 1
11 AANG = ARI*ARI + ANG*ANG
  P = D / SQRT(AANG)
  IF (ISI-4) 13,12,12
  ISI = ISI - 4
12 IF (ISI-2) 15,14,14
13 F = C + P
14 C = C + P
15 D = D * (AARI - GEO) * 0.5 / ARI
  IF (ABS(AARI-GEO) - 1.0E-6*AARI) 17,17,16
16 SGEO = SORT(SGEO)
  GEOMETRIC MEAN
  GEO = SGEO * SGEO
  FIM = FIM + PIM
  ISI = ISI + ISI
  GC TO 8
  C ACCURACY WAS SUFFICIENT
17 R = (ATAN(ARI/ANG) + PIM) * AN / ARI
  C = C + D * ANG / AANG
  GC TO 4
  END
  SUBROUTINE BDPROF (NPROF)
  COMMON/EARTH/ISPH
  COMMON/UNITS/ LC, LP, LT
  DIMENSION Z(50), C(50), Z1(50), C1(50)
  C*****
  C THIS ROUTINE READS PROFILES FROM CAR
  C*****
  C IF (ISPH.LE.0) WRITE (LP,12)
  C 12 FORMAT (40#SPHERICAL EARTH CORRECTION

```

THIS ROUTINE READS PROFILES FROM CARDS AND MAKES A FIELD TAPE

12 IF (ISPH.LE. 0) WRITE (LP,12)
12 FORMAT(40HOSPHERICAL EARTH CORRECTION APPLIED
//)

```

C      REWIND 2
CVP=1.0E10

C      DC 100 I = 1 NPTSF
C      READ(LC 1) RANGE,NPTS
C      RANGE1=RANGE
C      CONVERT INPUT RANGE (IN MILES) TO FEET
C      RANGE = RANGE * 6076.1
C      IF(I.NE.1) WRITE(2) RANGE
C      1 FORMAT(F10.2,I5)
C      READ(LC 2) (Z(L), C(L), L=1,NPTS)
C      2 FORMAT(8F10.2)
C      IF(C(1).GT.3000.0) GO TO 6
C      CONVERT INPUT TO ENGLISH UNITS
C      DO 3 J=1,NPTS
C      Z1(J)=Z(J)
C      Z(J)=Z(J)/.3048
C      C1(J)=C(J)
C      C(J)=C(J)/.3048
C      CCNTINUE
C      3 WRITE(LP,4) RANGE1
C      4 FORMAT(//,22HPROFILE AT RANGE = ,F10.2,3H NM
C      X 10H DEPTH(FT),5X,10HC(Z),5X,10H ,DEPTH(M),5X,10H /C(Z) (M/S))
C      WRITE(LP,5) (Z(J),C(J),Z1(J),C1(J),J=1,NPTS)
C      GC TO 90
C      5 FORMAT(F10.0,F15.2,F15.0,F15.2)
C      CCNTINUE
C      6 WRITE(LP,41) RANGE1
C      41 FORMAT(//,22HPROFILE AT RANGE
C      X 10H DEPTH(FT),5X,10HC(Z),5X,10H /C(Z) (FT/S)
C      WRITE(LP,7) (Z(J),C(J),J=1,NPTS)
C      7 FORMAT(F10.0,F15.2)
C      90 CCNTINUE
C      IF(ISP.H.GT.0) GO TO 10
C      DO 11 J = 1, NPTS
C      ZTEMP = Z(J)
C      Z(J) = Z(J) * (.0+Z(J)/4.1807E7)
C      C(J) = C(J) * 2.09035E7 / (2.09037E7 - ZTEMP)
C      11 CCNTINUE
C      10 CCNTINUE

C      FIND MIN
C      IF(I.GT.1) GO TO 9
C      DC 8 J = 1, NPTS
C      CVP = AMIN1(CVP,C(J))
C      8 CCNTINUE
C      WRITE(2) CVP

```

```

C      WRITE(2) CVF
C      9 CCNTINUE
C      DUMP TC TAPE
C      WRITE(2) NETS, (Z(I), C(L), L=1, NPTS)
C      100 CONTINUE
C      RANGE=1.0E10
C      WRITE(2) RANGE
C      REWIND 2
C      RETURN
C      END
C      SUBROUTINE GETBOT (IFIAT, DMAX)
C      READ AND PRINT BATHYMETRY. CONVERT RANGE, DEPTH TO FEET
C      CCOMMON /UNITS/ LC, LP, IT
C      CCOMMON /BATHY/ RE, KB, NB, BR(101), BZ(101)
C      DATA FNM, FT, RAD/6076.1, 6.3048, 6.17453292519943E-01/
C      NE=IABS (IFIAT)
C      IF (NB.LE.100) GO TO 10
C      WRITE (LP, 9CC)
C      900 FCRMAT (41H0**BOTTOM DATA EXCEED AVAILABLE STORAGE.)
C      STOP
C      IF (NB.EQ.C) GO TO 30
C      READ (LC, 91C) (BR(I), BZ(I), I=1, NB)
C      910 FCRMAT (8F10.2)
C      WRITE (LP, 920)
C      920 FCRMAT (1H0.7X, 10HEATHYMETRY/
C      1 24H POINT RANGE DEPTH)
C      DO 20 I=1, NB
C      WRITE (LP, 930) I, BR(I), BZ(I)
C      930 FORMAT (2X, I3, F8.1, 3X, F8.1)
C      BF(I)=FNM*BR(I)
C      IF (IFIAT.EQ.0) BZ(I)=BZ(I)/FT
C      20 DMAX=AMAX1 (BZ(I), DMAX)

```

```
C      EZ (NB+1) = EZ (NB)
C
C      30 CCNTINUE
C      PR (NB+1) = 1. CE16
C      RETURN
C      END
```

APPENDIX B: WT SOURCE CODE

THE PURPOSE OF THIS PROGRAM IS TO TRANSFORM THE ACOUSTIC
SSSFFT PARAFELIC EQUATION MODEL "PRESSURE DATA" INTO THE
WAVENUMBER DOMAIN FOR FURTHER ANALYSIS BY THE WAVENUMBER
TECHNIQUE AS DESCRIBED BY RICHARD LAUER OF NORDA

```

INTEGER N
INTEGER TITLE(16)
INTEGER Q(2050), P(2050), QMOD(2050)
REAL*4 FREFFHA(2050), IWK(13200), WK(13200)
DIMENSION FF(2050), PI(2050), RANGE(2050)
DIMENSION D(21), BUFO(21)
DCCMPLEX PREFHA
PIE=3.141592654
ATC=0.0015
NNTOT = 2050

```

ALL IS THE MISC. DATA INPUT FILE FROM THE PEMODEL

$$IT = 3$$

LT11 IS THE RANGE AND PRESSURE DATA INPUT FILE FROM THE PEMODEL

IT1 = 4

LO IS THE DATA OUTPUT FILE OF WAVENUMBER DATA

LO = 7

LD ID THE DIAGNOSTIC OUTPUT FILE

```
LD = 8      LT1
REWIND      LT1
```

READ INPUT

```

C$$$ (TITLE(I),I=1,16)
      READ(LT,800)
      800 FORMAT(16A4)

```

FREQ	INUT FREQUENCY
ZSS	SOURCE DEPTH
NFT	NUMBER OF POINTS IN THE "PRESSURE" ARRAY
NNF	NUMBER OF RECEIVER DEPTHS
CCLMIN	{ NCT USED } SEE PE PROGRAM
DCCL	{ NCI USED } SEE PE PROGRAM
FACT	{ NCI USED } SEE PE PROGRAM


```

111 CCNTINUE
113 CONTINUE
C
C
C      READ-IN RANGE AND COMPLEX "PRESSURE" DATA
      DO 860 IWT2=1, NPT
      READ (LT1,850) RANGE(IWT2), PK(IWT2), PI(IWT2)
850  FORMAT(3(2X,E15.7))
860  CCNTINUE
      WRITE(LD,901)
901  FORMAT('0', 'PASSED POINT 2')
      *****
      LET RANGE INCREMENT EQUAL THE FIRST RANGE STEE
      RINC=RANGE(2) - RANGE(1)
      CONVERT NAUTICAL MILES TO FEET
      DELR=RINC*6076.1
      ZERO OUT FUTURE ARRAYS
      CMOD WILL BE BETA
      DC 60 I=1, NTOT
      PREPHA(I)=CMPLX(0.0,0.0)
      F(I)=0.0
      Q(I)=0.0
      QMOD(I)=0.0
60  CCNTINUE
      WRITE(LD,9C3)
903  FORMAT('0', 'PASSED POINT 3')
      DO 75 I=1, NET
      C
      C
      C      INSERT THE HANKEL APPROXIMATION TO OBTAIN TRUE PRESSURE FROM
      INPUT "PRESSURE" AND BACK OUT ATTENUATION
      F(I)=(PR(I)*COS(FK*RANGE(I))-PI(I)*SIN(FK*RANGE(I)))
      1*EXP(ATC*RANGE(I))
      Q(I)=(PR(I)*SIN(FK*RANGE(I))+PI(I)*COS(FK*RANGE(I)))
      1*EXP(ATC*RANGE(I))
75  CCNTINUE
      CXXXXXXXXXXXXXXXXXXXXXXXXX
      WRITE(LD,9C4)
904  FORMAT('0', 'PASSED POINT 4')
      C
      C      LCAD REAL AND IMAGINARY PARTS INTO COMPLEX ARRAY

```

```

C      DO 100 I=1,NN
      PREPHA(I)={MELX(P(I),Q)
100    CONTINUE
      WRITE(LD,905)
905    FORMAT('0',P,5)
C      FAST FOURIER TRANSFORM PRESSURE DATA
C
C      CALL FFTCC(PREPHA,NN,IWK,WK)
      WRITE(LD,906)
906    FORMAT('0',P,6)
C      XI      IS INITIAL HORIZONTAL WAVENUMBER
      DZ      IS WAVENUMBER INCREMENT
C      MULTIPLICATION BY 1000 FOR COMPUTER USAGE ONLY
C
      XI=(2.0*PI*FREQ/VELC-2.0*PI*DELX)*1000.0
      DZ={2.0*PI*DELX/(NN*DELX)}*1000.0
      WRITE(LD,957)
957    FORMAT('0',P,6)
C      Q WILL BECOME THE SPECTRUM HORIZONTAL WAVENUMBER (HORIZ AXIS)
      P AT END OF LOOP WILL BE SPECTRAL INTENSITY (VERT AXIS)
C
C125  I=1,NN
      RN=FLOAT(I)
      A=XI+RN*DZ
      P(I)=AIMAG(PREPHA(I))
      Q(I)=REAL(PREPHA(I))
      F(I)=(P(I)**2+Q(I)**2)**(2)
C      1000 FOR COMPUTER USAGE ONLY, IS REMOVED
C
      Q(I)=A/1000.
C125  CONTINUE
      WRITE(LD,907)
907    FORMAT('0',P,7)
      FMAX=P(I)
C      DETERMINE MAXIMUM P FOR NORMALIZATION IN VERT AXIS
C
C127  I=1,NN
      IF(P(I).GT.FMAX) PMAX=P(I)
127    CONTINUE
C      NORMALIZE VERT AXIS

```



```
C##### START OF EXECUTABLE CODE #####C#####  
INTEGER TITLE(15),BUFO(21)  
DIMENSION L(21),QMOD(2050),P(2050)  
DIMENSION RANGE(2050)  
DATA IV,Y,N,N/  
  
LT IS THE MISC. DATA INPUT FILE FROM THE PEMODEL  
IT = 3  
  
LT1 IS THE RANGE DATA INPUT FILE FROM THE PEMODEL  
LT1 = 4  
  
LC IS THE DATA INPUT FILE OF WAVENUMBER DATA FROM THE FFT  
LC = 7  
  
LI ID THE LIAGNOSTIC OUTPUT FILE  
  
LD = 8  
CCNTINUE  
REWIND LT  
REWIND LT1  
REWIND LO  
  
READ INPUT  
  
10 READ(LT,100) (TITLE(I),I=1,15)  
100 FORMAT(15A4) FREQ,ZS,NPT,ND,CLMIN,DCL,FACT,IPLOT  
110 FORMAT(2(2X,F10.2),2(2X,I4),3(2X,F7.2),2X,I3)  
WRITE(LD,150)  
120 READ(LT,120) D(I),I=1,ND)  
FORMAT(5E15.7)  
IF(NPT.GT.2050) NPT = 2050  
DO 140 I=1,NPT  
READ(LT1,120) RANGE(I)  
FORMAT(2X,E15.7)  
130 CCNTINUE  
140  
150 FORMAT(' PASSED POINT 1')  
160 READ(LT,160) DDDD,FK,VELC,ATTEN,DMAX  
FORMAT(5(1X,E15.7))  
WRITE(LD,170) DDD
```



```

C
C
C
      READ(5,*) XX1,XX3,X2
      IF(XX1.NE.0.0) QMIN = XX1
      IF(XX3.NE.0.0) QMAX = XX3

      CLEAR SCREEN AT THE TERMINAL

      CALL F7C7MS('CLRS CRN',)
      WRITE(6,901) QMIN,QMAX
901  FORMAT(1,'IS',E9.2,'X-AXIS VALUE IS',E9.2,'ENTER NEW',
      1, 'VALUE IS',E9.2,'X-AXIS VALUE IS',E9.2,'ENTER NEW',
      2, 'MINIMUM AND MAXIMUM VALUES OR A ZERO FOR EACH IF YOU',
      3, 'WOULD LIKE TO ACCEPT THE CURRENT VALUES',
      4, 'ENTER DESIRED X-AXIS INCREMENT',
      5, ' (OR 0 FOR SELF SCALING)',)
      READ(5,*) XX1,XX3,XX2
      IF(XX1.NE.0.0) QMIN = XX1
      IF(XX3.NE.0.0) QMAX = XX3

      CLEAR SCREEN AT THE TERMINAL

      CALL F7C7MS('CLRS CRN',)
      CALL GET INCREMENT OF Y-AXIS
      WRITE(6,903) PMIN,PMAX
903  FORMAT(1,'IS',E9.2,'Y-AXIS VALUE IS',E9.2,'ENTER NEW',
      1, 'VALUE IS',E9.2,'Y-AXIS VALUE IS',E9.2,'ENTER NEW',
      2, 'MINIMUM AND MAXIMUM VALUES OR A ZERO FOR EACH IF YOU',
      3, 'WOULD LIKE TO ACCEPT THE CURRENT VALUES',
      4, 'ENTER DESIRED Y-AXIS INCREMENT',
      5, ' (OR 0 FOR SELF SCALING)',)
      READ(5,*) Y1,Y3,Y2
      IF(Y1.NE.0.0) PMIN = Y1
      IF(Y3.NE.0.0) PMAX = Y3

      CALL PAGE(11,8.5)
      CALL INTACS
      CALL SETDEV(1,0)
      CALL GRACE(0.6)
      CALL NCBRRD
      CALL AREA2D(9,0.6,0)
      CALL XNAME('SCALED WAVENUMBER BETA (1/FT)',29)
      CALL YNAME('NORMALIZED SPECTRAL INTENSITY',30)
      CALL YTICKS(5)
      CALL XTICKS(5)
      CALL HEADIN(1,5,1)
      IF((X2.NE.0.0).AND.(Y2.NE.0.0))
      *CALL GRAF(QMAX,XX2,QMIN,PMIN,Y2,PMAX)
      IF((X2.EQ.0.0).AND.(Y2.EQ.0.0))

```

```

*CALL GRAF(QMAX,'SCALE',QMIN,PMIN,'SCALE',PMAX)
IF((X2.EQ.0.0).AND.(Y2.NE.0.0))
*CALL GRAF(QMAX,'SCALE',QMIN,PMIN,Y2,PMAX)
IF((X2.NE.0.0).AND.(Y2.EQ.0.0))
*CALL GRAF(QMAX,X2,QMIN,PMIN,'SCALE',PMAX)
CALL BLSYN
CALL SPLINE
CALL CURVE(CMOD,PN,0)
CALL MESSAGE(SOURCE,DEPTH$,100,-.35,6.7)
CALL REALNC(ZS,2,'ABUT','ABUT')
CALL MESSAGE(ZS,2,'RECEIVER DEPTH $',100,'ABUT','ABUT')
CALL REALNC(D(1),2,'ABUT','ABUT')
CALL MESSAGE(D(1),2,'FREQUENCY $',100,'ABUT','ABUT')
CALL REALNC(FREQ,2,'ABUT','ABUT')
CALL MESSAGE(FREQ,2,'WATER DEPTH $',100,'ABUT','ABUT')
CALL REALNC(HZ$,2,'ABUT','ABUT')
CALL MESSAGE(HZ$,2,'WATER DEPTH $',100,'ABUT','ABUT')
CALL REALNC(DMAX,2,'ABUT','ABUT')
CALL MESSAGE(DMAX,2,'FIELD DEPTH $',100,'ABUT','ABUT')
CALL REALNC(DDDD,2,'ABUT','ABUT')
CALL MESSAGE(DDDD,2,'ATTENUATION COEF $',100,'ABUT','ABUT')
CALL REALNC(ATEN,-3,'ABUT','ABUT')
CALL MESSAGE(ATEN,-3,'AVERAGE WAVE NUMBER $',100,-.35,6.3)
CALL REALNC(AVER,-3,'ABUT','ABUT')
CALL MESSAGE(AVER,-3,'REFERENCE SOUND SPEED $',100,'ABUT','ABUT')
CALL REALNC(VELC,2,'ABUT','ABUT')
CALL MESSAGE(VELC,2,'FT./SEC$',100,'ABUT','ABUT')
CALL DCT
CALL GRID(1,1)
CALL ENDPL(0)

```

```

CALL PAGE(11,8.5)
CALL INTAXS
CALL SETDEV(1,0)
CALL GRACE(0,0)
CALL NOBRDE
CALL AREA2L(9,0,6,0)
CALL YNAME('HORIZONTAL WAVELENGTH',100)
CALL YTICKS(5)
CALL XTICKS(5)
CALL RESET('DOT')
CALL HEADIN('TITLE',60,1.5,1)
IF((X2.NE.0.0).AND.(Y2.NE.0.0))
*CALL GRAF(QMAX,X2,QMIN,PMIN,Y2,PMAX)
IF((X2.EQ.0.0).AND.(Y2.EQ.0.0))

```

C *****

```

*CALL GRAF(CMAX,SCALE,QMIN,PMIN,SCALE,PMAX)
IF((XX2-NE.0.0).AND.(Y2-NE.0.0))
*CALL GRAF(CMAX,SCALE,QMIN,PMIN,Y2,PMAX)
IF((XX2-NE.0.0).AND.(Y2-NE.0.0))
*CALL GRAF(CMAX,XX2,QMIN,PMIN,SCALE,PMAX)
CALL BLSYNE
CALL SPLINE
CALL CURVE(CP,NN,0)
CALL MESSAGE(SOURCE,DEPTH$,100,-.35,6.7)
CALL REALNC(ZS2,ABUT,ABUT)
CALL REALNC(ZS2,RECEIVED,DEPTH$,100,ABUT,ABUT)
CALL REALNC(D(1),ABUT,ABUT)
CALL REALNC(D(1),FREQUENCY$,100,ABUT,ABUT)
CALL REALNC(FREQ,2,ABUT,ABUT)
CALL MESSAGE(HZ$,100,ABUT,ABUT)
CALL REALNC(WATER,DEPTH$,100,-.35,6.5)
CALL REALNC(DMAX,2,ABUT,ABUT)
CALL MESSAGE(FD,2,FIELD,DEPTH$,100,ABUT,ABUT)
CALL REALNC(DDDD,2,ABUT,ABUT)
CALL MESSAGE(ATTEN,-3,ABUT,ABUT)
CALL REALNC(ATTEN,-3,ABUT,ABUT)
CALL MESSAGE(AVERAGE,WAVE NUMBER$,100,-.35,6.3)
CALL REALNC(FK,-3,ABUT,ABUT)
CALL MESSAGE(REF,REFERENCE,SOUND,SPEED$,100,ABUT,ABUT)
CALL REALNC(VELC,3,ABUT,ABUT)
CALL MESSAGE(FT./SEC$,100,ABUT,ABUT)
CALL DOT
CALL GRID(1,1)
CALL ENDPL(6)

CLEAR SCREEN AT THE TERMINAL

CALL FRTCHS('CLRSCHN ')
** WOULD YOU LIKE TO MAKE ANOTHER SET OF PLOTS?
WRITE(6,1900)
FORMAT(/,' WOULD YOU LIKE TO MAKE ANOTHER SET OF PLOTS',/,'
: ANSWER (Y/N)')
READ(5,505) IANS
905 FCRMAT(1A1)
IF((IANS-NE.IY).AND.(IANS-NE.NO)) GO TO 1800
IF(IANS-EQ.IY) GO TO 10
CALL DCNEPI
STOP
END

```

THIS SUBROUTINE IS USED TO OUTPUT DATA IN A FORMAT WHICH IS
REDUCED IN VOLUME AND COMPATIBLE FOR THE WAVENUMBER TECHNIQUE
(WT)

```

*****
** IMPLICIT FINITE-DIFFERENCE PROGRAM FOR SOLVING THE PARABOLIC EQUATION **
*****
** LT LARRY JAEGER POSTGRADUATE SCHOOL U.S. NAVAL CA 93943 MONTEREY, CA *****
** *****
** ALPHABETICAL LIST OF PROGRAM VARIABLES FOLLOWS: *****
** A - ARRAY - COEFFICIENT A IN PARABOLIC EQUATION (IN WATER) *****
** A2 - COEFFICIENT A IN PARABOLIC EQUATION (IN SEDIMENT) *****
** ALPHA - VOLUME ATTENUATION - DB/METER *****
** ATT - ARRAY - ATTENUATION COEFFICIENT FOR ARTIFICIAL ATTENUATION LAYER *****
** BEDA1 - BETA 1 AS DEFINED IN LEE AND MCDANIEL {1983} *****
** BEDA2 - BETA 2 AS DEFINED IN LEE AND MCDANIEL {1983} *****
** BETA1 - ATTENUATION IN WATER - DB/WAVELENGTH *****
** BETA2 - ATTENUATION IN SEDIMENT - DB/WAVELENGTH *****
** BR - ARRAY - RANGE FOR BOTTOM PROFILE - METERS *****
** BZ - ARRAY - DEPTH FOR BOTTOM PROFILE - METERS *****
** C - ARRAY - CONTAINS TRIANGONAL MATRIX SYSTEM THAT NEEDS TO BE SOLVED (SEE SUBROUTINE TRIDG) *****
** C0 - REFERENCE SOUND SPEED - METERS/SEC *****
** C2 - SOUND SPEED IN SEDIMENT *****
** COSE - COS (THETA) *****
** CR - ARRAY - STORAGE SPACE USED IN SUBROUTINES TRIDG AND TRIDL *****
** CSVP - ARRAY - SOUND SPEED IN SOUND SFEEL PROFILE - METERS/SEC *****
** CTWO - ARRAY - STORAGE SPACE USED IN SUBROUTINES TRIDG AND TRIDL *****
** CWATER - ARRAY - SOUND SPEED AT GRID POINTS IN WATER COLUMN *****
** DELTA - DELTA AS DEFINED IN LEE AND MCDANIEL (1983) *****
** DELIN - 1 / DELTA *****
** DR - RANGE STEP - METERS *****
** DRIVL - RANGE STEP ALONG LEVEL INTERFACE - METERS *****

```

```

*** DRMAX - MAXIMUM ALLOWABLE RANGE STEP - METERS
*** DZ - DEPTH INCREMENT OF SOLUTION - METERS
*** EYE - COMPLEX "I"
*** FREQ - FREQUENCY - HZ
*** GAMMA1 - GAMMA 1 AS DEFINED IN LEE AND MCDANIEL (1983)
*** GAMMA2 - GAMMA 2 AS DEFINED IN LEE AND MCDANIEL (1983)
*** IBOT1 - POINTER THAT POINTS TO BOTTOM ERFCE FILE PCINT AT START
          OF BOTTOM SEGMENT
** IBOT2 - POINTER THAT POINTS TO BOTTOM ERFCE FILE PCINT AT END
          OF BOTTOM SEGMENT
*** ID - ARRAY - RUN IDENTIFICATION
*** IFACE1 - POINTER THAT POINTS TO INTERFACE AT RANGE RA1
*** IFACE2 - POINTER THAT POINTS TO INTERFACE AT RANGE RA2
*** IFACEH - IFACE - 1
*** IFACEP - IFACE + 1
*** IP2 - EVERY IP2*TH VALUE IN DEPTH IS PRINTED
*** ISLOPE - SLOPE FLAG:
          ISLOPE = 1 - BOTTOM SLOPES DCWN
          ISLOPE = 2 - BOTTOM LEVEL
          ISLOPE = 3 - BOTTOM SLOPES UP
          ISLOPE = 4 - BOTTOM SLOPES DCWN, BOTTOM MODIFIED
          ISLOPE = 5 - BOTTOM SLOPES UP, BOTTOM MODIFIED
          TEMPORARY VARIABLE
*** ITEMP - GRID POINT CORRESPONDING TO RECEIVER DEPTH
*** IMZ - NUMBER OF EQUI-SPACED GRID POINTS IN U
*** N - INCLUDES BOTTOM POINT - DOES NOT INCLUDE SURFACE POINT
          INCLUDES POINTS IN ARTIFICIAL ATTENUATION LAYER
*** NA - NUMBER OF POINTS IN BOTTOM PRCFILE (BR AND EZ)
*** NBOT - NUMBER OF POINTS IN BOTTOM PRCFILE (BR AND EZ)
*** NIU - UNIT NUMBER FOR INPUT DATA
*** NM1 - N - 1
*** NOU - UNIT NUMBER FOR OUTPUT PLOTTER FILE
*** NPOT - UNIT NUMBER FOR OUTPUT PRINTER FILE
*** NSTEP - NUMBER OF RANGE STEPS ALONG A BOTTOM SEGMENT
*** NSTEP1 - NUMBER OF RANGE STEPS CORRESPONDING TO ONE VERTICAL
          GRID STEP FOR MODIFIED BOTTOM
*** NSVP - NUMBER OF POINTS IN CSVP AND ZSVP
*** NMAX - NUMBER OF GRID POINTS IN WATER AT MAX DEPTH
*** NXLFS - NUMBER OF NEXT LEVEL SECTION FOLLOWING A SLOPING
          SECTION FOR A MODIFIED BOTTOM
*** PDR - RANGE INCREMENT AT WHICH SOLUTION IS PASSED TO
          CUTPUT PRINTER FILE - METERS
*** PDZ - DEPTH INCREMENT AT WHICH SOLUTION IS PRINTED - METERS
*** PI - THE VALUE OF PI
*** PL - PROPAGATION LOSS - DB
*** PRIN - INITIAL RANGE FOR PLOT AND PRINT FILE - METERS
*** PRTOT - FINAL RANGE FOR PLOT AND PRINT FILE - METERS
*** R1 - RANGE AT WHICH BOTTOM DEPTH IS AVAILABLE - METERS
*** R2 - NEXT RANGE AT WHICH BOTTOM DEPTH IS AVAILABLE - METERS

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

```

- INCREMENTED AS SOLUTION IS MARCHED CUT IN RANGE.
- RANGE AT WHICH SOLUTION IS KNOWN - METERS
- ( RA2 = RA1 + DR ) - GM/CM**3
- DENSITY IN WATER
- DENSITY IN SEDIMENT - GM/CM**3
- MAXIMUM RANGE OF SOLUTION - METERS
- SIN ( THETA )
- TEMPERATURE VARIABLE
- SLOPE CF BOTTOM - RADIANS
- THETA = 0 - LEVEL INTERFACE DCWN
- THETA > 0 - INTERFACE SLOPES UP
- THETA < 0 - INTERFACE SLOPES UP
- ARRAY - COMPLEX ACOUSTIC PRESSURE FIELD
- RANGE STEP AT WHICH SOLUTION IS WRITTEN TO OUTPUT
- PLOTTER FILE
- REFERENCE WAVE NUMBER
- MATRIX ELEMENT, X MATRIX, LOWER DIAGONAL, ON INTERFACE
- XLI FOR SLOPING, BOTH
- REFERENCE WAVELENGTH - METERS
- MATRIX ELEMENT, X MATRIX, OFF-DIAGONAL, IN WATER AND
  SEDIMENT
- MATRIX ELEMENT, X MATRIX, MAIN DIAGONAL, ON INTERFACE
- MATRIX ELEMENT, X MATRIX, MAIN DIAGONAL, IN SEDIMENT
- ARRAY - MATRIX ELEMENT, X MATRIX, MAIN DIAGONAL, IN
  WATER
- REAL INDEX OF REFRACTION
- COMPLEX INDEX OF REFRACTION SQUARED
- RANGE AT WHICH SOLUTION IS PRINTED - METERS
- RANGE AT WHICH SOLUTION IS WRITTEN TO OUTPUT
- PLOTTER FILE
- VARIABLES THAT BEGIN WITH XX HAVE NO SPECIAL PHYSICAL
  SIGNIFICANCE BUT THEY CONTRIBUTE TO COMPUTATIONAL
  EFFICIENCY. ALL XX VARIABLES ARE CALCULATED IN
  SUBROUTINE INITIAL, ALL ARE INDEPENDENT OF RANGE STEP
  AND INTERFACE SLOPE, AND ALL ARE USED TO CALCULATE
  MATRIX ELEMENTS.
- MATRIX ELEMENT, Y MATRIX, LOWER DIAGONAL, ON INTERFACE
- YLI FOR LEVEL INTERFACE
- YLI FOR SLOPING INTERFACE
- MATRIX ELEMENT, Y MATRIX, OFF-DIAGONAL, IN WATER AND
  SEDIMENT
- MATRIX ELEMENT, Y MATRIX, MAIN DIAGONAL, ON INTERFACE
- MATRIX ELEMENT, Y MATRIX, MAIN DIAGONAL, IN SEDIMENT
- ARRAY - MATRIX ELEMENTS, Y MATRIX, MAIN DIAGONAL, IN
  WATER
- MATRIX ELEMENT, Y MATRIX, UPPER DIAGONAL, ON INTERFACE
- YRI FOR LEVEL INTERFACE

```

```

*** RA1
*** RA2
*** RHO1
*** RHO2
*** RMAX
*** SINE
*** TEMP
*** THETA
*** U
*** WDR
*** XKO
*** XLI
*** XLI2
*** XLAMDA
*** XLEWS
*** XMI
*** XMS
*** XMW
*** XN1
*** XN1
*** XPR
*** XPR
*** XX...
*** YLI
*** YLI2
*** YLI2
*** YLRWS
*** YMI
*** YMS
*** YMW
*** YRI
*** YRI

```



```

C 50 *** BOTTOM SLOPES DOWN SLOWLY, BOTTOM MODIFIED
C 60 CALL SSLOPE
C 70 GO TO 70
*** BOTTOM SLOPES UP SLOWLY, BOTTCM MODIFIED
CALL SSLOPE

CONTINUE
CALL ATTENU(U,ATT,IA,NA)
RA2F = RA2+0.5
*** TIME TO WRITE?
IF (RA2P-GE.XWR) CALL WRITE2
*** TIME TO PRINT?
IF (RA2P-GE.XPR) CALL PRWT2
*** TIME TC TERMINATE?
IF (RA2-GE.RMAX) GO TO 90
CONTINUE

80 *** GO BACK AND CONTINUE WITH NEXT LINEAR BOTTOM SEGMENT
GO TO 10

90 *** TIME TC TERMINATE
CCCONTINUE
CALL END (RA2)
STOP
END

SUBROUTINE READ
(1) THIS SUBROUTINE READS ALL INPUT DATA.
(2) THE DATA IS READ FROM INPUT UNIT NUMBER: NIU = 51
(3) INPUT FILENAME AND FILETYPE ARE: IFDIN DATAIN
(4) DATA IS READ IN FREE FORMAT.
(5) DATA IS TRANSFERRED BACK TO MAIN PROGRAM VIA COMMON BLOCK

COMMON /IN/ IA,IBOT1,IFACE,IPZ,ISLOPE,ISTEP,INZ,N,NA,NBCT,NM1,
* NSTEP,NSTEP1,NSVP,NWMAX,NXLFS
COMMON /REAL/ ALPHA,ATT(500),BETA1,BETA2,BR(101),BZ(101),C0
* CSVP(101),C2,CWATER(500),DR,DRLV1,DRMAX,DZ,FHQ,PCK,LDZ,
* R1,RA1,RA2,RHOC1,RHOC2,RHOC3,RHOC4,RHOC5,RHOC6,RHOC7,
* XX11,XWR,XWR1,ZLYR1,ZLYR2,ZR,ZS,ZSVP(101),ZABLYR,
* PRIN,PEI,TOT
CCCC DATA NIU/51, NPOUT/6/
C DATA NIU/51, NPOUT/55/
*** READ INPUT PARAMETERS

```

```

C C READ(NIU,*,END=100) FRQ, ZS, ZR, CO, N
C C READ(NIU,*,END=100) RMAX, DR, DRVL, DRMAX, WDR, PDR, PDZ, PRTIN
C C IF(PRTIN.EQ.0.0) PRTIN = WDR
C C PRTOT = RMAX
C C IF(PRTCT.LE. PRTIN) GO TO 90

C C *** READ BOTTOM PROFILE - RANGE, DEPTH
C C DO 10 I=1,101
C C READ(NIU,*,END=100) BR(I), BZ(I)
C C NBOT=I
C C *** END OF PROFILE?
C C IF(BR(I).LT.0.0) GO TO 20
C C *** NO
C C CONTINUE
C 10

C C CONTINUE
C C *** EXTEND LAST DEPTH BEYOND MAX RANGE
C C BR(NBOT) = 1.0E+10
C C BZ(NBOT) = BZ(NBOT-1)
C C 20

C C *** FIRST LAYER IS WATER. SECOND IS SEDIMENT.
C C *** READ MAX DEPTH, DENSITY AND ATTENUATION OF FIRST LAYER
C C READ(NIU,*,END=100) ZLYR1, RHO1, BETA1
C C

C C *** READ SOUND SPEED PROFILE IN FIRST LAYER
C C DO 25 I=1,101
C C NSVP=I
C C READ(NIU,*,END=100) ZSVP(I), CSVP(I)
C C *** READ ANOTHER PROFILE POINT?
C C IF(ZSVP(I).LT.ZLYR1) GO TO 25
C C *** NO
C C *** WAS THAT THE LAST PROFILE POINT?
C C IF(ZSVP(I).EQ.ZLYR1) GO TO 30
C C *** NO, THERE IS ERROR.
C C GO TO 101
C C 25

C C CONTINUE
C C *** DOES THE SOUND SPEED PROFILE START AT THE SURFACE?
C C IF ( ZSVP(1).NE.0.0 ) GO TO 102
C C *** YES
C C 30

C C *** READ DEPTH, DENSITY, ATTENUATION AND SPEED IN SECOND LAYER
C C READ(NIU,*,END=100) ZLYR2, RHO2, BETA2, C2
C C *** READ DEPTH OF UPPER EDGE OF ARTIFICIAL ATTENUATING LAYER
C C READ(NIU,*,END=100) ZADLYR
C C RETURN
C C *** ERROR EXISTS

```

```

90 WRITE(6,895)
   WRITE(NPOUT,899)
   STOP
100 WRITE(6,900)
   WRITE(NPOUT,900)
   STOP
101 WRITE(6,901)
   WRITE(NPOUT,901)
   STOP
102 WRITE(6,902)
   WRITE(NPOUT,902)
   STOP
C
899 FORMAT('// 1X,ERROR: THE INITIAL PRINTER/PLOTTER RANGE IS',//,
          * 9X,'GREATER THAN OR EQUAL TO THE FINAL RANGE' '//,
          * 9X,'EXECUTION TERMINATED' '//)
900 FCRMAT('// 1X,ERROR: EXPECTING MORE INPUT DATA. '//, 9X,
          * 'EXECUTION TERMINATED' '//)
901 FORMAT('// 9X, ERROR: FINAL DEPTH IN SOUND SPEED PROFILE DOES NOT ',
          * 'EQUAL MAXIMUM DEPTH OF WATER COLUMN.' '//, 9X,
          * 'EXECUTION TERMINATED' '//)
902 FCRMAT('// 9X, ERROR: FIRST DEPTH IN SOUND SPEED PROFILE',
          * 'NOT EQUAL ZERO.' '//, 9X,
          * 'EXECUTION TERMINATED' '//)
END
C
SUBROUTINE SVPW
C
      THIS SUBROUTINE CALCULATES THE VERTICAL STEP SIZE: DZ
      THIS SUBROUTINE ALSO CALCULATES THE SPEED OF SOUND AT
      EACH OF THE VERTICAL GRID POINTS.
      SOUND SPEED VALUES ARE DETERMINED BY LINEAR INTERPOLATION.
      SOUND SPEEDS ARE STORED IN CWATER(I).
      {A} THE INDEX I RANGES FROM 1 TO NWMAX.
      {B} CWATER(I) CORRESPONDS TO THE GRID POINT DZ BELOW
           THE SURFACE.
C
COMMON /IN/ IA,IBOT1,IFACE,IPZ,ISLOPE,ISTEP,IWZ,N,NA,NBCT,NM1,
* NSTEP,NSTEP1,NSVP,NWMAX,NXLFS
COMMON /REAL/ ALPHA,ATT(500),BETA1,BETA2,BR(101),BZ(101),CO
* CSVF(101),C2,CWATER(500),DR,DRLVL,DRMAX,DZ,FRO,PDR,FDZ,
* R1,RAA1,RAA2,RHO1,RHO2,RNAX,THETA,XK0,XIAMB,A,XPR,XX4,XX10,
* XX11,XWR,WDR,ZLYR1,ZLYR2,ZR,ZS,ZSVF(101),ZABLYR,
* PRTN,PROT

```



```

10      *      0.5 *(CSVP(I)-CSVP(I-1)))
11      CONTINUE
C      CO = CC/ZSVP(NSVP)
C      CONTINUE
C      *** INITIALIZE RANGE
C      RAI = 0.0
C      *** INITIALIZE POINTER THAT POINTS TO BOTTCM PROFILE POINT
C      IBCT1 = 0
C      *** COMPUTE REFERENCE WAVE NUMBER
C      XKO = 2.0*PI*FRQ/CO
C      *** COMPUTE REFERENCE WAVELENGTH
C      XLAMDA = CO/FRQ
C      *** IF DRLVL=0 SET DRLVL EQUAL TO 1/2 REFERENCE WAVELENGTH
C      IF ( DRLVL.EQ.0.0 ) DRLVL = 0.5 * XLAMDA
C      *** IF DRMAX=0 SET DRMAX EQUAL TO REFERENCE WAVELENGTH
C      IF ( DRMAX.EQ.0.0 ) DRMAX = XLAMDA
C      *** IF DRLVL GREATER THAN DRMAX SET DRLVL EQUAL TO DRMAX
C      IF (DRLVL.GT.DRMAX) DRLVL = DRMAX
C      *** COMPUTE ATTENUATION - SACLANT MEMO SM-121 (JENSEN + FERLA)
C      MODIFIED AS FOLLOWS:
C      *** IF INPUTTED BETA IS LT 0.0, ALPHA IS COMPUTED IN DB/METER
C      *** AND USED FOR BETA
C      ALPHA=FRQ*FRQ*(-.007+ (.155*1.7)/(1.7*1.7+FRQ*FRQ*.000001))
C      *      *1.0E-09
C      *** INITIALIZE POINTER THAT POINTS TO INTERFACE GRID POINT
C      IFACE = INT ( BZ(1)/DZ + 0.5 )
C      RETURN
C      END
C      SUBROUTINE MATCON
C      THIS SUBROUTINE CALCULATES VARIOUS VARIABLES NEEDED TO COMPUTE
C      TRIDIAGONAL MATRIX ELEMENTS. VARIABLES BEGINNING WITH XX HAVE
C      NO SPECIAL PHYSICAL SIGNIFICANCE BUT THEY CONTRIBUTE TO
C      COMPUTATIONAL EFFICIENCY.

```

[illegible]

123


```

*      WRITE(NFOUT,900) (ID(I), I=1,15),FRQ,ZS,ZR,CO,RMAX,ZLYR1,
      BETA1,XK0,XLAMDA,DZ,WDR,ZABLYR,N
900    FORMAT(T5,15A4,/,8(1X,F9.3),/,4(1X,F9.3),1X,I10)
      IPZ = INT(PDZ/(DZ+0.5))
      IF(IPZ.EQ.0) IPZ = 1
      XFR = RA1 + PDR
      RETURN
      END

```

SUBROUTINE NEWSEG

```

THIS SUBROUTINE IS CALLED AT THE START OF EACH NEW BOTTC1
SEGMENT. THE SUBROUTINE DOES THE FOLLOWING TASKS FOR EACH
EOTTCM SEGMENT:
{1} UPDATES BOTTOM PROFILE POINTERS: IBCT1 & IPOT2
{2} COMPUTES SLOPE: THETA
{3} COMPUTES NUMBER OF RANGE2 STEPS IN SEGMENT: NSTEP
{4} COMPUTES RANGE STEP: DR
{5} SETS SLOPE FLAG: ISLOPE
    {A} ISLOPE = 1 - BOTTOM SLOPES DOWN
    {B} ISLOPE = 2 - BOTTOM LEVEL
    {C} ISLOPE = 3 - BOTTOM SLOPES UP
    {D} ISLOPE = 4 - BOTTOM SLOPES DOWN, MODIFY BOTTOM
    {E} ISLOPE = 5 - BOTTOM SLOPES UP, MODIFY BOTTOM
{6} INITIALIZES RANGES: RA1 & RA2
{7} CHECKS THAT RANGE STEP IS LESS THAN DRMAX
{8} ISSUES ERROR OR WARNING MESSAGES AS APPROPRIATE

```

```

COMMON /IN/ IA,IBOT1,IFACE,IPZ,ISLOPE,ISTEP,IWZ,N,NA,NBCT,NM1,
*      NSTEP,NSTEP1,NSVP,NMAX,NX1FS
COMMON /REAL/ ALPHA,ATT(5000),BETA1,BETA2,ER(101),BZ(101),CO,
*      CSVE(101),C2,CHATER(5000),DR,DR1VL,DRMAX,DZ,FRQ,PDR,PDZ,
*      R1,RA1,RA2,RHO2,RMAX,THETA,XK0,XLAMDA,XPR,XX4,XX10,
*      XX11,XX12,XWR,WDR,ZLYR1,ZLYR2,ZR,ZS,ZSVE(101),ZABLYR,
*      PRTIN,PRTOT
DATA NFOUT/55/

```

```

** UPDATE EOTTCM PROFILE POINTER
IBCT1 = IBOT1 + 1
IBCT2 = IBOT1 + 1
** GET STARTING AND ENDING RANGES AND DEPTHS FOR THIS SEGMENT
R1 = BR(IBOT1)
Z1 = BZ(IBOT1)
R2 = BR(IBOT2)

```

```

C      Z2 = B2(1BOT2)
C      *** ERFCR CHECK
C      IF (R2.LE.R1) GO TO 100
C      *** PUT Z1 AND Z2 ON NEAREST GRID POINTS
C      ITEMP = INT (Z1/DZ + 0.5)
C      Z1 = DZ * FLOAT(ITEMP)
C      ITEMP = INT (Z2/DZ + 0.5)
C      Z2 = DZ * FLOAT(ITEMP)
C      *** COMPUTE SLOPE
C      THETA = ATAN2 (Z2-Z1,R2-R1)
C      *** DOES BOTTOM SLOPE DOWN, LEVEL OR UP?
C      IF (THETA.GT.0.0) GO TO 10
C      IF (THETA.LT.0.0) GO TO 20
C      *** BOTTOM IS LEVEL
C      *** DETERMINE NUMBER OF RANGE STEPS FOR SEGMENT
C      NSTEP = INT ((R2-R1)/DRIVL + 0.99999)
C      *** DETERMINE RANGE STEP
C      LR = (R2-R1) / FLOAT(NSTEP)
C      *** SET ISLOPE
C      ISLOPE = 2
C      GO TC 80
C
C      *** BOTTOM SLOPES DOWN
C      *** DETERMINE NUMBER OF RANGE STEPS
C      NSTEP = INT ((Z2-Z1+0.05)/DZ)
C      *** DETERMINE RANGE STEP
C      LR = (R2-R1)/FLOAT(NSTEP)
C      *** SET ISLOPE
C      ISLOPE = 1
C      GO TC 30
C
C      *** BOTTOM SLOPES UP
C      *** DETERMINE NUMBER OF RANGE STEPS
C      NSTEP = INT ((Z1-Z2+0.05)/DZ)
C      *** DETERMINE RANGE STEP
C      LR = (R2-R1)/FLOAT(NSTEP)
C      *** SET ISLOPE
C      ISLOPE = 3
C
C      CCNTINUE
C      *** IS RANGE STEP TOO LARGE?
C      IF ( DR.LE.DRMAX ) GO TO 80
C      *** YES, BOTTOM MUST BE MODIFIED
C      *** SET ISLOPE
C      ISICPE = 4
C      IF ( THETA.LT.0.0 ) ISLOPE = 5
C      *** DETERMINE NUMBER OF RANGE STEPS REQUIRED TO MOVE UP

```

```

C      *** OR ICWN ONE GRID POINT
C      NSTEP1 = INT ( DR/DRMAX + 0.99999 )
C      *** DETERMINE RANGE STEP
C      DR = DR / FLCAI(NSTEP1)
C      *** REDETERMINE NUMBER OF RANGE STEPS
C      NSTEP = NSTEP * NSTEP1
C      *** COMPUTE SLOPE OF SLOPING SECTION
C      THETA = ATAN2(DZ, DR)
C      *** SLOPING SECTION
C      NYLIS = NSTEP1/2 + 2
C      *** INDICATE TO USER THAT BOTTOM HAS BEEN MODIFIED
C      TEMF = 0.5 * DZ
C      WRITE(6,903) R1, R2, TEMP
C      WRITE(NPOUT,903) R1, R2, TEMP

C 80  CONTINUE
C      *** INITIALIZE RA1 & RA2
C      RA1 = R1
C      RA2 = RA1 + DR
C
C      *** INDICATE TO USER HOW FAR SOLUTION FIELD HAS PROGRESSED
C      WRITE(6,902) R1
C
C      *** IF RANGE STEP GREATER THAN 1 (?) WAVELENGTH WRITE WARNING
C      IF ( DR.LE.XLAMDA ) GO TO 90
C      WRITE(6,901) R1, R2, DR, XLAMDA
C      WRITE(NPOUT,901) R1, R2, DR, XLAMDA
C
C 90  RETURN
C
C 100 *** ERROR EXIT
C      WRITE(6,900) IBOT2, IBOT1
C      WRITE(NPOUT,900) IBOT2, IBOT1
C      STOP
C
C 900 FORMAT(//, 1X, 'ERROR: THE RANGE AT BOTTOM PROFILE PCINT NUMBER',
*      I2, ' IS LESS', 9X, ' THAN THE RANGE AT BOTTOM PROFILE POINT',
*      I2, ' NUMBER', I2, ' ', 1X, 'EXECUTION TERMINATED.', ///)
C
C 901 FORMAT(//, 'WARNING: THE HORIZONTAL RANGE STEP BETWEEN RANGE R =', F8.1, ' METERS',
*      ' AND RANGE R =', F8.1, ' (METERS) IS', F5.1, ' METERS',
*      ' THE REFERENCE WAVELENGTH IS', F5.1, ' METERS',
*      ' THE PROGRAM HAS REACHED RANGE R =', F8.1, ' METERS',
*      ' FCRMAT(//, ' NOTE: THE BOTTOM MODIFIED BETWEEN RANGE', F8.1, ' AND RANGE',
*      F8.1, ' HAS BEEN MODIFIED BECAUSE OF ITS VERY SMALL',
*      ' SLOPE', 8X, ' THE DIFFERENCE BETWEEN THE MODIFIED',

```

'BOTTOM AND YOUR'//8X,'INPUT BOTTOM IS NEVER GREATER',
'THAN',F5.2,' METERS.'//

END

SUBROUTINE NEWMAT

```

THIS SUBROUTINE IS CALLED AT THE START OF EACH NEW BOTTCM
SEGMENT. THE SUBROUTINE DOES THE FOLLOWING TASKS:
(1) COMPUTES TRIANGONAL MATRIX ELEMENTS FOR MATRIX Y
      {A} MATRIX Y IS AT RANGE WHERE FIELD IS KNOWN: RANGE=RA1
      {B} MATRIX ON RHS OF EQUATION
      {C} Y MATRIX ELEMENTS AND KNOWN FIELD AT RANGE RA1 TO
(2) USES Y MATRIX ELEMENTS AND KNOWN FIELD AT RANGE RA1 TO
      COMPUTE RHS COLUMN VECTOR C(I,4). (SEE TRIDG AND TEXT)
(3) COMPUTES TRIANGONAL MATRIX ELEMENTS FOR MATRIX X
      {A} X MATRIX ON LHS OF EQUATION
      {B} X MATRIX VALUES STORED IN C(I,1), C(I,2) AND C(I,3)
      {C} (SEE TRIDG AND TEXT REFERENCED IN TRIDG)
           X MATRIX AT RANGE: RA2 = RA1 + DR

```

TRIDIAGONAL MATRIX ELEMENTS ARE REPRESENTED BY VARIABLES BEGINNING WITH X OR Y. FOR VARIABLES BEGINNING WITH X OR Y, THE LETTERS IN THE VARIABLE NAME HAVE THE FOLLOWING SIGNIFICANCE:

```

X Y M L R W I S V Z
- - - - -
ELEMENT IN ROW AT INTERFCE
ELEMENT IN ROW AT INTERFCE
ELEMENT IN ROW AT INTERFCE
ELEMENT TO LEFT OF MAIN DIAGONAL
ELEMENT TO RIGHT OF MAIN DIAGONAL
X MATRIX
Y MATRIX
MAIN DIAGONAL
MAIN DIAGONAL
WATER
SLOPING INTERFACE

```

[illegible]

```

**      CSVF(101) , C2,CWATER(5000),DR,DRLVL,DRMAX,DZ,FRO,PCR,PDZ,
**      R1,FA1,RA2,RHO1,RHO2,RMAX,THETA,XK6,XLAMDA,XPR,XX4,XX10,
**      XX11,XMR,WDR,ZLYR1,ZLYR2,ZR,ZS,ZSVF(101),ZABLYE,
**      PRJIN,PRFOT
**      COMMON /CPIX/A(5000),A2,C(5000,4),CR(5000),CTWO(5000),
**      EYE,XLI,XLI2,XLRWS,XMI,XMS,XRI,XRI2,
**      XX1,XX2,XX3,XX4,XX5,XX6,XX7,XX8,XX9,XX12,XX1M(5000),
**      YLI,YLIV,YLI2,YLRWS,YMI,YMS,YMW(5000),YRI,YRIV,YRI2,
**      U(5000),Z25,Z26,Z27,Z28,Z29,ZZ10
**
**      COMPUTE MAIN DIAGONAL ELEMENT, Y MATRIX, IN SEDIMENT
**      YMS = 1.0 + DR*XX3
**      COMPUTE OFF-DIAGONAL ELEMENTS, Y MATRIX, IN WATER & SEDIMENT
**      YLRWS = CR*XX1
**      COMPUTE MAIN DIAGONAL ELEMENT, X MATRIX, IN SEDIMENT
**      XMS = 2.0 - YMS
**      COMPUTE OFF-DIAGONAL ELEMENTS, X MATRIX, IN WATER & SEDIMENT
**      XLRWS = -YLRWS
**      COMPUTE FIRST ELEMENT IN RHS COLUMN VECTOR
**      YMW(1) = 1.0 + DR*XX1M(1)
**      C(1,4) = U(1)*YMW(1) + U(2)*YLRWS
**      COMPUTE TWO ELEMENTS IN FIRST ROW ON LHS
**      C(1,2) = 2.0 - YMW(1)
**      C(1,3) = XLRWS
**      COMPUTE REMAINING ELEMENTS ON BOTH RHS & LHS FOR ROWS IN WATER
**      IFACEM = IFACE - 1
**      DO 10 I=2,IFACEM
**          FIRST WORK WITH RHS
**          YMW(I) = 1.0 + DR*XX1M(I)
**          C(I,4) = U(I)*YMW(I) + (U(I-1)+U(I+1))*YLRWS
**          WORK WITH LHS
**          C(I,1) = XLRWS
**          C(I,2) = 2.0 - YMW(I)
**          C(I,3) = XLRWS
10      CONTINUE
**      COMPUTE LHS & RHS ELEMENTS IN SEDIMENT
**      IFACEP = IFACE + 1
**      NM1 = N - 1
**      DO 20 I=IFACEP,NM1
**          RHS
**          C(I,4) = U(I)*YMS + (U(I-1)+U(I+1))*YLRWS
**          LHS
**          C(I,1) = XLRWS
**          C(I,2) = XMS
**          C(I,3) = XLRWS
20      CONTINUE
**      IF ENTIRE SEGMENT LEVEL GO TO 50

```



```

C (IFACE2,1) = XLI
C (IFACE2,2) = XMI
C (IFACE2,3) = XRI
*** COMPUTE X MATRIX ELEMENTS ONE ROW ABOVE INTERFACE
C (IFACE,1) = XLRWS
C (IFACE,2) = 1.0 - DR * XX1M(IFACE)
C (IFACE,3) = XLRWS
GO TO 60

*** BOTTOM SLOPES UP
IFACE2 = IFACEM
*** COMPUTE OFF-DIAGONAL, Y MATRIX ELEMENTS ON INTERFACE
YLI = 0.5 * DR * GAMMA2
YRI = 0.5 * DR * BEDA2
*** COMPUTE MAIN DIAGONAL, Y MATRIX ELEMENT ON INTERFACE
YMI = - A(IFACE) * ZZ1 + ZZ9
*** COMPUTE INTERFACE ELEMENT IN RHS COLUMN VECTOR
C (IFACE,4) = U(IFACEM)*YLI + U(IFACE)*YMI +
              U(IFACE)*YRI
*** COMPUTE X MATRIX ELEMENTS ON INTERFACE
XLI = -0.5 * DR * GAMMA1
XMI = - A(IFACE2) * ZZ5 + ZZ10
XRI = -0.5 * DR * BEDA1
*** IF MODIFIED BOTTOM THEN NO NEED TO ADJUST LHS
IF ( ISLOPE.EQ.5 ) GO TO 45
C (IFACE2,1) = XLI
C (IFACE2,2) = XMI
C (IFACE2,3) = XRI
*** COMPUTE X MATRIX ELEMENTS ONE ROW BELOW INTERFACE
C (IFACE,1) = XLRWS
C (IFACE,2) = XMS
C (IFACE,3) = XLRWS
GO TO 60

*** SAVE INTERFACE VALUES ON SLOPING SECTION
YLI2 = YLI
YRI2 = YRI
XLI2 = XLI
XRI2 = XRI

*** SEGMENT LEVEL 1
IFACE2 = IFACE
YLI = DR * XX6
YMI = 1.0 + DR * ( XX1M(IFACE)/XX4 + XX5 )
YRI = DR * XX7
C (IFACE,4) = U(IFACEM)*YLI + U(IFACE)*YMI + U(IFACE)*YRI
C (IFACE,1) = -YLI
C (IFACE,2) = 2.0 - YMI

```

```

C(IFACE,3) = -YRI
*** SAVE INTERFACE VALUES ON LEVEL SECTION
    YLIV = YII
    YRIV = YRI

CONTINUE

60 C
C *** COMPUTE ARTIFICIAL ATTENUATION MATRIX
C *** SEE AESL PE MODEL BY BROCK - NORDA TECH NOTE 12 - JAN 78
C *** CALCULATE GRID POINT AT TOP OF ART ATTENUATION LAYER
C ** IA = INT ( ZABLYR/DZ + 0.01 )
C *** CALCULATE NUMBER OF GRID POINTS IN ART ATTENUATION LAYER
C ** NA = N - IA
C *** CALCULATE ATTENUATION MATRIX
DO 70 I=1,NA
    TEMP = 3.0 * (I-NA) / NA
    ATT(I) = EXP (-0.01*DR*EXP(-(TEMP*TEMP)))
CONTINUE

70 C
C *** SOLVE FOR SOLUTION FIELD AT RANGE RA2
CALL TRIDG (C,U,N,Cr,CTWO)

C *** APPLY ARTIFICIAL ATTENUATION
CALL ATTENU (U,ATT,IA,NA)

RA2P = RA2 + 0.5
*** TIME TC WRITE?
IF ( RA2P.GE.XWR ) CALL WRITE2
*** TIME TC PRINT?
IF ( RA2P.GE.XPR ) CALL PRTWT2

C *** UPDATE INTERFACE POINTER
IFACE = IFACE2

RETURN
END

SUBROUTINE TRIDG (C,U,N,Cr,CTWO)

*** THIS SUBROUTINE SOLVES A SET OF N - 1 (NM1) LINEAR
*** SIMULTANEOUS EQUATIONS HAVING A TRIANGULAR COEFFICIENT
*** MATRIX. MATRIX ELEMENTS IN THE LOWER DIAGONAL, MAIN DIAGONAL
*** AND UPPER DIAGONAL ARE STORED IN C(I,1), C(I,2) AND C(I,3)
*** RESPECTIVELY. THE RHS COLUMN VECTOR IS STORED IN C(I,4).
*** THE SOLUTION FIELD IS STORED IN U(I).
```

```

(1)  THE INDEX I REFERS TO ROW NUMBER.
(2)  WE NEED ONLY SOLVE AN NM1 X NM1 SYSTEM (RATHER THAN
(3)  AN N X N SYSTEM) BECAUSE U(N) IS KNOWN: U(N)=0.0
(4)  THE SUBROUTINE IS A MODIFIED VERSION OF SUBROUTINE
    TRIDG FROM:
    "APPLIED NUMERICAL ANALYSIS" ( SECOND EDITION )
    BY:  CURTIS F. GERALD
    PUBLISHED BY ADDISON-WESLEY PUBLISHING CO. 1980
    THE MAIN MODIFICATIONS TO THE ROUTINE IN THE TEXT
    INVOLVED:
    (A)  INTRODUCING ARRAYS CTWO AND CF TO PRESERVE THE
        ORIGINAL VALUES IN C(I,2) AND TC MAKE THE ROUTINE
        MORE EFFICIENT. THIS RESULTS IN A CONSIDERABLE
        SAVINGS IN EXECUTION TIME FOR THE CASE OF A
        HORIZONTAL BOTTOM. (SEE SUBROUTINE TRIDL)
    (B)  MODIFYING THE ROUTINE TO SOLVE AN NM1 X NM1
        SYSTEM.
(5)  SEE PAGES 129 AND 133 IN THE TEXT FOR FURTHER INFO.

```

```

CCOMPLEX C(5000,4), U(5000), CR(5000), CTWO(5000)

```

```

NM1 = N - 1
NM2 = N - 2
CTWO(1) = C(1,2)
DO 10 I=2,NM1
  CR(I) = C(I,1) / CTWO(I-1) * C(I-1,3)
  CTWC(I) = C(I,2) - CR(I) * C(I-1,4)
  C(I,4) = C(I,4)
CCONTINUE

```

```

U(N) = 0.0

```

```

** NOW PERFORM BACK SUBSTITUTION

```

```

U(NM1) = C(NM1,4) / CTWO(NM1)
DO 20 I=1,NM2
  U(I) = NM1 - I
  U(I) = ( C(M,4) - C(M,3) * U(M+1) ) / CTWO(M)
CONTINUE
RETURN
END

```

```

SUBROUTINE TRIDL (C,U,N,CR,CTWO)

```

```

** THIS SUBROUTINE IS A MODIFIED VERSION OF SUBROUTINE TRIDG
** FROM THE IFD PROGRAM. SUBROUTINE TRIDG IS IN TURN A MODIFIED
** VERSION OF TRIDG AS PER THE REFERENCE BELOW. LINEAR
** THE SUBROUTINE SOLVES A SET OF N-1 (NM1) COEFFICIENT
** SIMULTANEOUS EQUATIONS HAVING A TRIANGULAR MAIN DIAGONAL
** MATRIX. MATRIX ELEMENTS IN THE LOWER TRIANGULAR, MAIN DIAGONAL
** AND UPPER DIAGONAL ARE STORED IN C(I,1), C(I,2), AND C(I,3)
** RESPECTIVELY. THE RHS COLUMN VECTOR IS STORED IN C(I,4).
** THE SOLUTION FIELD IS STORED IN U(I).
(1) THE INDEX I REFERS TO ROW NUMBER.
(2) WE NEED ONLY SOLVE AN NM1 X NM1 SYSTEM (RATHER THAN
(3) AN NM1 X NM1 SYSTEM) BECAUSE U(N) IS KNOWN: U(N)=0.0
(4) THE SUBROUTINE IS A MODIFIED VERSION OF IFD SUB-
ROUTINE TRIDG WHICH IN TURN IS A MODIFIED VERSION
OF SUBROUTINE TRIDG AS PER:
"APPLIED NUMERICAL ANALYSIS" ( SECOND EDITION )
BY: CURTIS F. GERALD
PUBLISHED BY ADDISON-WESLEY PUBLISHING CO., 1980
THE ONLY MODIFICATION TO IFD SUBROUTINE TRIDG IS
THAT TRIDG DOES NOT RECALCULATE CTWO AND CR BUT
BUT RATHER TRIDG USES THE ARRAY VALUES CALCULATED
BY TRIDG. THIS RESULTS IN A CONSIDERABLE SAVINGS
IN EXECUTION TIME FOR THE CASE OF A HORIZONTAL
BOTTOM.
**
** COMPLEX C(5000,4), U(5000), CR(5000), CTWO(5000)
NM1 = N - 1
NM2 = N - 2
DO 10 I=2,NM1
  C(I,4) = C(I,4) - CR(I) * C(I-1,4)
CONTINUE
U(N) = 0.0
** NOW PERFORM BACK SUBSTITUTION
U(NM1) = C(NM1,4) / CTWO(NM1)
DO 20 I=1,NM2
  U(I) = ( C(I,4) - C(I,3)*U(I+1) ) / CTWO(I)
CONTINUE
RETURN
END

```

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

10

CCCCCCCC

20

CCCCCCCC

```

SUBROUTINE DOWN
THIS SUBROUTINE UPDATES THE RHS & LHS OF THE EQUATION AND
SOLVES FOR THE SOLUTION FIELD AT RA2.
(1) THE RHS COLUMN VECTOR VALUES ARE STORED IN C(I,4)
(2) THE INTERFACE AT RANGE RA1 IS AT GRIDPOINT IFACE1
(3) THE INTERFACE AT RANGE RA2 IS AT GRIDPOINT IFACE2
( WHERE IFACE2 = IFACE + 1 )

COMPLEX A,A2,C,CR,CTWO,EYE,XMS,XRI,XRIZ,
* XLI,XLIZ,XLRWS,XMI,
* XX1,XX2,XX3,XX4,XX5,XX6,XX7,XX8,XX9,XX12,XX1M
* YLI,YLIV,YLIZ,YLRWS,YMI,YMS,YMW,YRI,YRIV,YRIZ,
* U,ZZ5,ZZ6,ZZ7,ZZ8,ZZ9,ZZ10
COMMON /IN/IA,IBOT1,IFACE,IP2,ISLOPE,ISTEP,IPZ,N,NA,NBOT,NM1,
* NSTEP,NSTEP1,NSVP,NWMAX,NXLFS
COMMON /REAL/ALPHA,ATT(5000),BETA1,BETA2,ER(101),BZ(101),CO
* CSVE(101),C2,CWATER(5000),DR,DRLVL,DRMAX,DZ,FRQ,PDR,PDZ,
* R1,RA1,RA2,RHO2,RMAX,THETA,XK6,XIAMD,XPK,XX4,XX10,
* XX11,XPR,WDR,ZLIR1,ZLVR2,ZR,ZS,ZSVE(101),ZAEIYR,
* PRTN,PRFOT
COMMON /CPIX/ A(5000),A2,C(5000,4),CR(5000),CTWO(5000),
* EYE,XLI,XLIZ,XLRWS,XMI,XMS,XRI,XRIZ
* XX1,XX2,XX3,XX4,XX5,XX6,XX7,XX8,XX9,XX12,XX1M(5000)
* YLI,YLIV,YLIZ,YLRWS,YMI,YMS,YMW(5000),YRI,YRIV,YRIZ,
* U(5000),ZZ5,ZZ6,ZZ7,ZZ8,ZZ9,ZZ10

*** UPDATE IFACE2
IFACE2 = IFACE + 1
*** UPDATE Y MATRIX* MAIN DIAGONAL, INTERFACE ELEMENT
YMI = A(IFACE), ZZ5 + ZZ6
*** UPDATE Y MATRIX*, MAIN DIAGONAL, WATER ELEMENT, CHE ROW
*** ABCVE INTERFACE
YMW(IFACE-1) = 1.0 + DR * XX1M(IFACE-1)

*** UPDATE RHS
CALL RHS
*** UPDATE IHS
*** ** UPDATE X MATRIX ELEMENTS ONE ROW ABCVE INTERFACE
C{IFACE,1} = XLRWS
C{IFACE,2} = 1.0 - DR*XX1M(IFACE)
C{IFACE,3} = XLRWS
*** ** UPDATE X MATRIX ELEMENTS ON INTERFACE
C{IFACE2,1} = XLI
C{IFACE2,2} = A(IFACE2) * ZZ7 + ZZ8
C{IFACE2,3} = XRI

```



```

*      YLI YLIV YLIZ YLRWS YMI YMS YMW(5000) YRI YRIV YRIZ,
*      U(5000), Z25, Z26, Z27, Z28, Z29, Z30
C      *** IS THIS A LEVEL SECTION FOLLOWING A SLOPING SECTION?
C      IF (ISTEP.EQ.NXLFS) GO TO 20
C      *** NO
C      *** IS THIS A SLOPING SECTION?
C      ITEST = NXLFS - ISTEP
C      IF (ITEST.EQ.1) GO TO 30
C      *** NO
C      *** LEVEL SECTION FOLLOWS A LEVEL SECTION
C      CALL LEVEL
C      GO TO 50
C      *** LEVEL SECTION FOLLOWS A SLOPING SECTION
C      *** UPDATE NXLFS + NSTEP1
C      NXLFS = NXLFS + NSTEP1
C      *** IF LAST SECTION SLOPED DOWN, UPDATE Y MATRIX ELEMENT,
C      *** MAIN DIAGONAL IN WATER ONE ROW ABOVE INTERFACE
C      IF (ISLOPE.EQ.5) GO TO 25
C      YMW(IFACE-1) = 1.0 + DR * XX1M(IFACE-1)
C      *** UPDATE RHS INTERFACE ELEMENTS
C      YLI = YLIV + DR * (XX1M(IFACE)/XX4 + XX5)
C      YRI = YRIV
C      *** UPDATE LHS
C      C(IFACE,1) = -YLI
C      C(IFACE,2) = 2.0 - YMI
C      C(IFACE,3) = -YRI
C      *** SOLVE SYSTEM
C      CALL RHS
C      CALL TRIDG(C,U,N,CR,CTWO)
C      GO TO 50
C      *** SLOPING SECTION
C      *** UPDATE INTERFACE ELEMENTS
C      YLI = YLIZ
C      YRI = YRIZ
C      XLI = XRI
C      *** SOLVE SYSTEM AS APPROPRIATE
C      IF (ISLOPE.EQ.4) CALL DOWN
C      IF (ISLOPE.EQ.5) CALL UP
C      CCNTINUE
C      RETURN

```

```

END
SUBROUTINE FHS
THIS SUBROUTINE MULTIPLIES TRIDIAGONAL MATRIX Y TIMES SOLUTION
FIELD U TO OBTAIN AN UPDATED RHS.
(1) THE RHS COLUMN VECTOR VALUES ARE STORED IN C(I,4) .

CCOMPLEX A, A2, C, CR, CTWO, EYE,
* XLI, XLIZ, XLRWS, XMI, XMS, XRI, XRIZ,
* XX1, XX2, XX3, XX5, XX6, XX7, XX8, XX9, XX12, XX1M,
* YLI, YLIV, YLIZ, YLRWS, YMI, YMS, YMW, YRI, YRIV, YRIZ,
* U, Z25, Z26, Z27, Z28, Z29, Z30
CCOMMON /IM/ IA, IBOT1, IFACE, IP2, ISLOPE, ISTEP, IWZ, N, NA, NBOT, NM1,
* NSTEP, NSTEP1, NSVP, NMAX, NYLFS
CCOMMON /REAL/ ALPHA, ATT(5000), BETA1, BETA2, BR(101), BZ(101), CO
* CSVF(101), C2, CHWATER(5000), DR, DRLVL, DRMAX, DZ, FRQ, PBR, PDZ,
* RI, RA1, RA2, RHO1, RHO2, RMAX, THETA, XKO, XIAMBA, XPR, XX4, XX10,
* XX11, XGR, WDR, ZLYR1, ZLYR2, ZR, ZS, ZSVF(101), ZABLYR,
* PRIN, PRIOT
CCOMMON /CPLX/ A(5000), A2, C(5000,4), CR(5000), CTWO(5000),
* EYE, XLI, XLIZ, XLRWS, XMI, XMS, XRI, XRIZ,
* XX1, XX2, XX3, XX5, XX6, XX7, XX8, XX9, XX12, XX1M(5000)
* YLI, YLIV, YLIZ, YLRWS, YMI, YMS, YMW(5000), YRI, YRIV, YRIZ,
* U(5000), Z25, Z26, Z27, Z28, Z29, Z30

*** UPDATE IFACEM & IFACEP
IFACEM = IFACE - 1
IFACEP = IFACE + 1
*** UPDATE RHS
C(1,4) = U(1) * YMW(1) + U(2) * YLRWS
DO 10 I=2, IFACEM
C(I,4) = U(I) * YMW(I) + (U(I-1)+U(I+1)) * YLRWS
CONTINUE
C(IFACE,4) = U(IFACEM)*YLI + U(IFACE)*YMI + U(IFACEP)*YRI
DO 20 I=IFACEP,NM1
C(I,4) = U(I)*YMS + (U(I-1)+U(I+1))*YLRWS
CONTINUE

RETURN
END

```

```

SUBROUTINE LEVEL
THIS SUBROUTINE UPDATES THE RHS OF THE EQUATION AND SOLVES
FOR THE SOLUTION FIELD AT RANGE RA2.
  (1) THE RHS COLUMN VECTOR VALUES ARE STORED IN C(1,4).
  (2) FOR THE LEVEL INTERFACE THE LHS TRIDIAGONAL MATRIX
      ELEMENTS NEED NOT BE UPDATED.

CCOMPLEX A,A2,C,CR,CTWO,EYE,XMS,XRI,XRIZ,
* XLI,XLI2,XLRHS,XMI,XMS,XRI,XRIZ,
* XX1,XX2,XX3,XX5,XRHS,XMI,XMS,XRI,XRIZ,
* YLI,YLI2,YLI3,YLRHS,YMI,YMS,YMW,YRI,YRIV,YRIZ,
* U,ZZ5,ZZ6,ZZ7,ZZ8,ZZ9,ZZ10
COMMON /IN/IA,IBOT1,NSVP,NSTEP,ISLOPE,ISTEP,IFZ,N,NA,NBCT,NM1,
* NSTEP,NSTEP1,NSVP,NMAX,NXLFS
COMMON /REAL/ALPHA,ATT(5000),BETA1,BETA2,BR(101),BZ(101),CO,EDZ,
* CSVP(101),C2,CWATER(5000),DR,DRLVL,DHMAX,DZ,FRQ,PRQ,PDZ,
* R1,RA1,RA1WDR,ZLH1,RHO2,RMAX,THTA,XK0,XIAMDA,XPR,XX4,XX10,
* XX11,XMR,WDR,ZLH1,ZLYR2,ZR,ZS,ZSVP(101),ZAEIYR,
* PRIN,PRTOT
COMMON /CPIX/ A(5000),A2,C(5000,4),CR(5000),CTWO(5000),
* EYE,XLI,XLI2,XLRHS,XMI,XMS,XRI,XRIZ,
* XX1,XX2,XX3,XX5,XX6,XX7,XX8,XX9,XX12,XX1M(5000)
* YLI,YLI2,YLI3,YLRHS,YMI,YMS,YMW(5000),YRI,YRIV,YRIZ,
* U(5000),ZZ5,ZZ6,ZZ7,ZZ8,ZZ9,ZZ10

*** UPDATE RHS
CALL RHS

*** SOLVE THE TRIDIAGONAL SYSTEM
CALL TRIDL (C,U,N,CR,CTWO)

RETURN
END

SUBROUTINE ERTWT2

THIS SUBROUTINE IS USED TO OUTPUT DATA IN A FORMAT WHICH IS
REDUCED IN VOLUME AND COMPATIBLE FOR THE WAVENUMBER TECHNIQUE
(FT)

(1) THIS SUBROUTINE IS EFFECTIVELY THE CONTINUATION OF
    SUBROUTINE PRWT1.
(2) THE FILE CREATED CORRESPONDS TO UNIT FILE NUMBER:

```

```

CCCCCCCCC
(3)      NPCUT = 55
          THE FILENAME AND FILETYPE FOR THIS FILE ARE:
          IFDOU1 PRINTER
(4)      NPCUT1 = 60
          THE FILENAME AND FILETYPE FOR THIS FILE ARE:
          IFDOU1A PRINTER

* COMPLEX A, A2, C, CR, CIWO, EYE,
* X11, X12, X1RWS, XMI, XMS, XRI, XRIZ,
* XX1, XX2, XX3, XX5, XX6, XX7, XX8, XX9, XX12, XX1M,
* Y11, Y12, Y1RWS, YMI, YMS, YMW, YRI, YRIV, YRIZ,
* U, Z25, Z26, Z27, Z28, Z29, Z30
* COMMON /IN, IA, IB, OT1, IFACE, IPZ, ISLOPE, ISTEP, IWZ, N, NA, NBOT, NM1,
* NSIPE, NSTEP1, NSVP, NMAX, NALFS
* COMMON /REAL, ALPHA, ATT(5000), BETA1, BETA2, ER(101), EZ(101), CO
* CSVE(101), C2, CMATER(5000), DR, DRVLV, DRMAX, DZ, FRC, PDR, PDZ,
* R1, RA1, RA2, RHO1, RHO2, RMAX, THETA, XK6, XLAMDA, XPR, XX4, XX10,
* XX11, XWR, WDR, ZL1R1, ZL1R2, ZR, ZS, ZSVE(101), ZAEIYR,
* PRTIN, PRTOT
* COMMON /CPIX, A(5000), A2, C(5000, 4), CR(5000), CIWO(5000),
* EYE, Y11, Y12, Y1RWS, XMI, XMS, XRI, XRIZ,
* XX1, XX2, XX3, XX5, XX6, XX7, XX8, XX9, XX12, XX1M(5000)
* Y11, Y12, Y1RWS, YMI, YMS, YMW(5000), YRI, YRIV, YRIZ,
* U(5000), Z25, Z26, Z27, Z28, Z29, Z30
* DATA NPOUT/55, NPOUT1/60,
* ZWHOLD = 0.0
* ZLHOLD = 0.0

*** INDICATE TO USER HOW FAR SOLUTION FIELD HAS PROGRESSED
BY INDICATING 1 KM STEPS
LPRTCK = MOD(INT(RA2), 1000)
IF(LPRTCK.EQ.0) WRITE(6,10) RA2
10 FCRMAT/, THE PROGRAM HAS REACHED RANGE R = , F10.3, METERS.

*** PRINT RANGE, U (REAL) AND U (IMAGINARY)

DO 20 I=IPZ, N, IPZ
ZI = FICAT(1), IPZ
ZIH = ZI + (FLOAT(I+IPZ)*DZ)
IF(ZI.EQ.ZR)LCHK = I
IF(ZI.IT.ZR).AND.(ZIH.GT.ZR))
LCHK = I+1
IF(ZI.EQ.ZLYR1)ZWHOLD = CABS(U(I))
IF(ZI.IT.ZLYR1).AND.(ZIH.GT.ZLYR1))
ZWHOLD = CABS(U(I+1))
IF(ZI.EQ.ZABLYR)ZLHOLD = CABS(U(I))
IF(ZI.IT.ZABLYR).AND.(ZIH.GT.ZABLYR))
ZLHOLD = CABS(U(I+1))

```

[illegible]

```

*      YLI, YLIV, YLI2, YLRWS, YMI, YMS, YMW(5000), YEI, YRIV, YRIZ,
*      U(5000), Z25, Z26, Z27, Z28, Z29, Z210
DATA NOU/52/

```

```

*** WRITE RANGE, DEPTH AND U(I)
IF((RA2.LT. PRTIN).OR. (RA2 .GT. PRTCT)) GO TO 50
WRITE(NCU,*) RA2, ZK, U(IWZ)
50 CCNTINUE

```

```

*** DETERMINE NEXT RANGE AT WHICH TO WRITE SOLUTION
XWR = XWR+WDR

```

```

RETURN
END

```

```

SUBROUTINE ATTENU(U, ATT, IA, NA)

```

```

THIS SUBROUTINE APPLIES ARTIFICIAL ATTENUATION TO THE BOTTOM-
MOST NA GRID POINTS AS PER AESD PE MODEL BY BROCK - NCHDA
TECH NOTE 12 - JAN 78
(1) ATTENUATION MATRIX ATT IS CALCULATED IN SUBROUTINE
    NEWMAT

```

```

COMPLEX U(5000)
DIMENSION ATT(5000)

```

```

DO 10 I=1, NA
  U(IA+I) = U(IA+I) * ATT(I)
CCNTINUE

```

```

RETURN
END

```

```

SUBROUTINE END (RA2)

```

```

THIS SUBROUTINE IS CALLED WHEN THE SOLUTION FIELD HAS REACHED
THE MAXIMUM RANGE SPECIFIED (RMAX). THE SUBROUTINE SENDS
APPROPRIATE MESSAGES TO THE TERMINAL AND STOPS EXECUTION.

```

```

WRITE(6,895) RA2
WRITE(6,900)
WRITE(6,901)

```

```

C
899 STOP
900 FCRMAT (1) THE PROGRAM HAS REACHED RANGE R = 'P8.1,' METERS.')
* 10X (1) RUN, PRINTER FILE: ' ' OUTPUT HAS BEEN GENERATED. ' '
* 15X (1) A FILE WITH FORMAT TO BE SENT TO THE PRINTER. ' '
* 15X (1) THIS FILE IS READY TO BE SENT TO THE PRINTER. ' '
* 15X (1) THE FILENAME AND FILETYPE FOR THIS FILE ARE: ' '
* 15X (1) IPDOUT PRINTER ' '
* 10X (2) OUTPUT PLCTTR FILE: ' ' OUTPUT HAS BEEN GENERATED. ' '
* 15X (2) A FILE WITH UNFORMATTED INPUT DATA FOR THE PLCT ROUTINE. ' '
* 15X (2) THIS FILE CONTAINS INPUT DATA FOR THE PLCT ROUTINE. ' '
* 15X (2) THE FILENAME AND FILETYPE FOR THIS FILE ARE: ' '
* 15X (2) IPDOUT PLOTTER ' '
* 15X (2) THE FILENAME AND FILETYPE FOR THE PLCT ROUTINE ARE: ' '
* 15X (2) PLCTIF FORTRAN ' '
* 15X (E) TO PLOT THE DATA GO TO A GRAPHICS TERMINAL AND ' '
* 15X (E) ENTER COMMANDS AS FOLLOWS: ' '
* 15X (E) DEF STOR 1M ' '
* 15X (E) I CMS ' '
* 15X (E) FORTGI PLOTIFD ' '
* 15X (E) PLOTIFD ' '
* 15X (E) THEN FOLLOW PROGRAM PROMPTS.')
* 15X (E)
C
RETURN
END

```

LIST OF REFERENCES

1. Naval Underwater Systems Center Technical Report 4103, Fast Field Program for the Multilayered Media, by P.R. DiNapoli, 26 June 1977.
2. Naval Underwater Systems Center Technical Memorandum, Signal Transmission in the Wavenumber Domain, by R. Lauer, 5 June 1979.
3. Stamey, B.B., Preliminary Investigation of the Environmental Sensitivity of Acoustic Signal Transmission in the Wavenumber Domain with Respect to Source Depth Determination, Naval Postgraduate School, December 1982.
4. Naval Ocean Research and Development Activity Technical Note 12, The AESD Parabolic Equation Model, H.K. Brock, January, 1978.
5. Officer, C.B., Introduction to the Theory of Sound Transmission with Application to the Ocean, McGraw-Hill, 1958.
6. Buckner, H.P., "Use of calculated sound fields and matched-field detection to locate sound sources in shallow water," J. Acoust. Soc. Am., v. 59, p. 368-373, February 1976.
7. Brekhovskikh, L.M., Fundamentals of Ocean Acoustics, Springer-Verlag, New York, 1982.
8. DeSanto, J.A., "Relation between the Solutions of the Helmholtz and Parabolic Equations for Sound Propagation," J. Acoust. Soc. Am., v. 62, p. 295-297, August 1977.
9. DeSanto, J.A., Topics in Current Physics: Ocean Acoustics, Springer-Verlag, 1979.
10. Coppens, A.B., An Introduction to the Parabolic Equation for Acoustic Propagation, Naval Postgraduate School Technical Report, NPS57-33-002, November 1982.
11. Kinsler, L.E., Frey, A.R., Coppens, A.B., and Sanders, J.V., Fundamentals of Acoustics Third Edition, John Wiley and Sons, 1982.
12. Naval Underwater Systems Center Technical Report 6659, IFD: An Implicit Finite-Difference Computer Model for Solving the Parabolic Equation, by J. Lee and G. Botseas, 27 May 1982.

13. Jaeger, L.E., A Computer Program for Solving the Parabolic Equation Using an Implicit Finite-Difference Solution Method Incorporating Exact Boundary Conditions, Naval Postgraduate School, September 1963.

BIBLIOGRAPHY

Bannister, R.W., "Low-frequency surface interference effects in long-range sound propagation," J. Acoust. Soc. Am., v. 69, p. 76-83, January 1981.

Brigham, E.O., The Fast Fourier Transform, Prentice-Hall Inc., 1974.

Brock, H.K., Buchal, R.N., and Spofford, C.W., "Modifying the Sound Speed Profile to Improve the Accuracy of the Parabolic Equation Technique", J. Acoust. Soc. Am., v. 62, p. 543-552, September, 1977.

Clay, C.S. and H. Medwin, Acoustic Oceanography: Principles and Applications, John Wiley and Sons, 1977.

DiNapoli, F.R. and R.L. Deavenport, "Theoretical and Numerical Green's function field solution in a plane multi-layered medium", J. Acoust. Soc. Am., v. 67, p. 92-105, January 1980.

Hamming, R.W., Digital Filters Second Edition, Prentice-Hall Inc., 1977.

Naval Undersea Center Technical Report TP 488, Low-Frequency Propagation Effects for Sources or Receivers Near the Ocean Surface, by V.A. Pedersen, D.F. Gordon, and D. White, September 1975.

Ross Donald, Mechanics of Underwater Noise, Pergamon Press, 1976.

Urick, R.J., Principles of Underwater Sound, Second Edition, McGraw-Hill, 1975.

Urick, R.J., Sound Propagation in the Sea, Defense Advanced Research Projects Agency, 1979.

INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, VA 22314	2
2. Library, Code 0142 Naval Postgraduate School Monterey, CA 93943	2
3. Chairman (Code 68Mr) Department of Oceanography Naval Postgraduate School Monterey, CA 93943	1
4. Chairman (Code 63Rd) Department of Meteorology Naval Postgraduate School Monterey, CA 93943	1
5. Prof. A.B. Coppens (Code 61Cz) Department of Physics Naval Postgraduate School Monterey, CA 93943	4
6. Prof. C.R. Dunlap (Code 68Du) Department of Oceanography Naval Postgraduate School Monterey, CA 93943	3
7. Lt. J.L. Blanchard 598-B Michelson Rd. Monterey, CA 93940	3
8. Director Naval Oceanography Division Naval Observatory 34th and Massachusetts Avenue NW Washington, DC 20390	1
9. Commander Naval Oceanography Command NSTI Station Bay St. Louis, MS 39522	1
10. Commanding Officer Naval Oceanographic Office NSTI Station Bay St. Louis, MS 39522	1
11. Commanding Officer Fleet Numerical Oceanography Center Monterey, CA 93940	1
12. Commanding Officer Naval Ocean Research and Development Activity (Attn: Code 201) NSTI Station Bay St. Louis, MS 39522	1

13. Commanding Officer
Naval Environmental Prediction
Research Facility
Monterey, CA 93940 1
14. Chairman, Oceanography Department 1
U.S. Naval Academy
Annapolis, MD 21402
15. Chief of Naval Research 1
800 N. Quincy Street
Arlington, VA 22217
16. Office of Naval Research (Code 420) 1
Naval Ocean Research and Development
Activity
800 N. Quincy Street
Arlington, VA 22217
17. Scientific Liaison Office 1
Office of Naval Research
Scripps Institution of Oceanography
La Jolla, CA 92037
18. Library 1
CICESE
P.O. Box 4803
San Ysidro, CA 92073
19. Commander 1
Oceanographic Systems Pacific
Box 1390
Pearl Harbor, HI 96860
20. Commander (AIR-370) 1
Naval Air Systems Command
Washington, DC 20360
21. Commanding Officer 1
Attn: Mr R.B. Lauer, Code 323
Naval Ocean Research and Development
Activity
NSTL Station
Bay St. Louis, MS 39522

REPROD

FILMED

8

DOWNHILL