PROPAGATION AND ATTENUATION CHARACTERISTICS

OF MULTILAYERED MEDIA

by

STEPHEN PATRICK KOCH

B.S.E., Purdue University
(1974)

DATA LIBRARY & ARCHIVES
Woods Hole Oceanographic Institution

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF

OCEAN ENGINEER

at the

WOODS HOLE OCEANOGRAPHIC INSTITUTION

and at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and

MASTER OF SCIENCE IN OCEAN ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

1979

Signature of Author....Stephen P. Koch....
Joint Program in Oceanographic Engineering, Woods Hole Oceanographic Institution - Massachusetts Institute of Technology, and the Department of Ocean Engineering, Massachusetts Institute of Technology

Certified by..........
Thesis Supervisor

Certified by....George V. Frisk....
Thesis Supervisor

Accepted by..........
Chairman, Joint Committee for Oceanographic Engineering, Woods Hole Oceanographic Institution - Massachusetts Institute of Technology

MBL/WHOI
0 0301 0075599 7

PROPAGATION AND ATTENUATION CHARACTERISTICS

OF MULTILAYERED MEDIA

by

STEPHEN PATRICK KOCH

Submitted to the Department of Ocean Engineering in partial fulfillment of the requirements for the degrees of Master of Science and Ocean Engineer.

ABSTRACT

An algorithm is proposed to numerically integrate the inhomogeneous depth-separated wave equation using a state variable technique. The solution obtained for two simple shallow water models is shown to agree well with the known exact solutions. Integration grid density is discussed and a minimum required density specified.

The use of complex sound speeds to simulate bottom attenuation is reviewed. A numerical instability inherent to the technique that arises during the use of complex sound speeds is investigated.

The algorithm is also applied to a deep ocean profile, and the solution characteristics discussed. Sensitivity problems that arise when modelling the seafloor as a layered elastic medium are analyzed.

Thesis Supervisor: Arthur B. Baggeroer
Title: Associate Professor of Ocean Engineering

Thesis Supervisor: George V. Frisk
Title: Assistant Scientist, W.H.O.I.

ACKNOWLEDGEMENTS

Deep thanks to all my friends in Cambridge, Fort Wayne, and Woods Hole for the encouragement, interest, and alcohol that helped finish this work.

Special thanks to my mother for encouragement in getting it started.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

Since Pekeris's study of shallow water sound propagation(1), considerable effort has been directed toward developing a better understanding of the interaction of oceanic acoustic waves with the seabed. This effort has occurred in two disciplines, seismology and ocean acoustics.

In seismology, primary interest in wave propagation below the ocean-seabed interface. Such information as layer depths and crustal structure is desired, with the goal being to confirm seismic models through the generation of synthetic seismograms. Mathematical extensions of ray theory are usually used to construct the solution to the inhomogeneous wave equation(2,3,4,6,7). Ocean acoustics, on the other hand, is concerned with the seabed as a lossy and dispersive medium which influences the structure of the acoustic pressure field in the ocean. Construction of the solution via an expansion of the eigenfunctions associated with the homogeneous equation is most common(9,10, 12), although several recent methods obtain the Green's function solution to the inhomogeneous equation directly(13,14). The advantages and disadvantages of the most important analysis techniques that have evolved in each field will be discussed in the following.

SEISMOLOGY

The most basic approach to elastic wave propagation in the seismic literature is the Thomson-Haskell method(2,3). It considers a medium of parallel, isotropic layers which have constant elastic parameters and sound speeds. A plane elastic wave at specified wave number is propagated at oblique incidence through the stratified media. Matrices which express the boundary conditions for each interface are written. The properties of the wave in the N^{th} layer are obtained in terms of the same properties in the

1^{st} layer via a recurrence relation which involves the product of N matrices(2).

This matrix product is a major problem with the Thomson-Haskell method and prevents its more widespread use. Accurate modeling of the medium often requires a large number of layers(due to the restriction that each layer must have constant physical properties) and computation of the product of a large number of matrices is both time-consuming and inaccurate. The solution contains both positive and negative exponential terms and may suffer from numerical instability. This instability results from the loss of that arises as the exponentially growing term increases through the profile.

As an example, the Thomson-Haskell method has been applied to the special case of low-frequency sound propagation in the Arctic Ocean(22,23). In this application the upward-refracting sound speed profile required the division of the fluid into layers of constant speed overlying an infinite half-space. The half-space was taken to be either a high-speed fluid(22) or a solid(23). The calculation of the transmission matrices required the use of double precision arithmetic and limited depths to approximately 1 kilometer. These restrictions prevent the apllication of the method to more general deep ocean models.

The method of characteristics(4,5) can be used to solve the wave equation since it is a second order hyperbolic differential equation. The characteristics are the natural coordinates of the differential equation and are roughly analogous to the natural coordinates used in multidegree of freedom vibration systems. In application of the method to wave propagation problems, an initial wavefront is assumed on which the velocity potential and all first order derivatives are known. The characteristic equations, which represent propagation paths forward of the initial wavefront, can be derived from the inhomogeneous form of the original differential equation. A second set of equations, the conditionals, are then derived from the inhomogeneous equation. These two sets of equations are equivalent to the boundary value problem.

Numerical calculation of the solution is straightforward. The characteristics form a grid forward of the initial wavefront. By using the known initial conditions to begin integration of the conditionals, the solution is sequentially determined throughout the region.

Since the conditionals are usually integrated numerically, an advantage of this technique is that all elastic properties and sound speeds can vary. Discontinuities in the initial conditions are also acceptable and will propagate along the grid.

The main problem with the method of characteristics is similar to the instability problem of the Thomson-Haskell method. The propagation paths represent increasing and decreasing amplitude terms. As the solution is computed, the increasing term tends to swamp the contribution of the decreasing term, causing a loss of precision and numerical stability.

Finite difference techniques are most commonly implemented in both the integration and the approximation of the appropriate boundary conditions. The errors due to these procedures require high grid density and a resulting increase in computation time. The sequential nature of the solution is also a direct cause of long computation times.

Two Fourier transform techniques that have become important in the siesmic literature in recent years are the ray-theoretical and reflectivity methods. Both methods consider an earth model consisting of an oceanic liquid layer overlying a solid medium of plane, homogeneous, isotropic, elastic layers and an elastic half-space.

The ray-theoretical method requires the calculation of the step function response of the earth model for a point source and receiver in the ocean layer. The step-function response involves using generalized reflection and transmission coefficients to obtain an integral representation of the acoustic field. Next, a source transfer function is written which describes the nature of the pressure field due to the underwater explosion that acts as the acoustic source. The synthetic seismogram is then obtained by convolving the step response with the time derivative of the source transfer function(6). The initial step of the reflectivity method is the numerical integration in the horizontal wave number domain of the plane wave reflection coefficient(reflectivity). This procedure is repeated for each horizontal phase velocity and the seismogram is obtained by the inverse transformation of the product of the source spectrum and the reflectivity(7).

Since it includes multiple reflections and converted waves, the reflectivity method provides highly accurate synthetic seismograms. However, it suffers from long computation times if the reflection response has a long duration(7). The ray-theoretical method does not have a dependence on response duration but instead has a computation time which is directly related to the number of multiple reflections and conversions that are included(6). For any particular application, the choice must be made between the error resulting from omission of these waves and the computation time resulting from their inclusion. Common procedure is to use the ray-theoretical method to obtain an overall seismic picture , and the reflectivity method to study specific details of the profile under consideration(7).

OCEAN ACOUSTICS

In the ocean acoustic literature, emphasis has been on the application of various numerical techniques to the integration in the water column of the depth-separated wave equation. In addition, various computer programs have been developed which obtain solutions for those ocean models in which the sound speed can be written in terms of specified mathematical functions. Finally, Fourier transform theory has been used to study acoustic propagation for given source and receiver configurations.

The Naval Research Laboratory(NRL) programs FLUID and SOLID are examples of the direct application of finite difference techniques to the solution of the homogeneous wave equation(10). The primary goal in each is the determination of the eigenvalues and eigenfunctions of the normal mode form of the solution. The physical model considered is a fluid layer with a specified sound speed profile that overlies a half-space which is either fluid of solid. The elastic parameters of the half space are constants. The iteration procedure employed involves integration to obtain a trial eigenfunction for an estimated eigenvalue and adjustment of the estimate depending on the error obtained between the value of the eigenfunction at the surface and the surface boundary condition. This procedure is repeated until the error at the surface is within a prespecified bound.

Expression of the differential equation and the boundary conditions in finite difference form is straightforward. For multilayered work the particular form of the boundary condition approximation is important both with regard to the accuracy of the solution and with regard to the nature of the compressional-shear coupling in the elastic medium. The elastic multilayered model would require consideration of this topic. It is likely that the extension to the more complex model would also cause a substantial increase in computation time.

The Applied Research Laboratory at the University of Texas has developed a normal mode program which also obtains the eigenvalues and eigenfunctions of the homogeneous equation(12). It employs numerical integration as well as a parallel shooting technique for efficient computation of each mode and such of its properties as group velocity and attentuation. Parallel shooting involves parameterizing the velocity potential by introducing an independent variable, and dividing the depth coordinate intosubintervals(31). The differential equation is then written as a first order system where each member corresponds to a particular subinterval and the coupling of the equations is expressed by the appropriate continuity conditions. Using an assumed eigenvalue, the system is integrated towards the sound channel. The secant method is used to improve the estimate and reduce the error between the upward and downward integrated terms to within the desired bounde. Although reasonably efficient in its present application, the technique suffers from the same drawbacks that influenced the NRL programs, when applied to the more complex model.

The parabolic wave equation method(13) is a technique of solving the inhomogeneous wave equation that is particularly useful for studying long range propagation in the ocean. It allows both depth and range-dependent sound speed profiles. The two-dimensional parabolic wave equation is obtained from the elliptic wave equation by replacing the velocity potential with the product of a Hankel function and an envelope function, and assuming that:1) the receiver is in the far field, 2) only small angles to the horizontal are to be considered, and 3) motions are uncoupled in azimuthal directions. The integration is then performed by using a finite difference Fourier algorithm.

Since the parabolic equation is valid only in the far field, its solution must be matched with a near field solution, which is known for simple sources. The upper boundary condition is specified through the use of an image source out of phase with the actual source. The bottom outgoing wave boundary condition is obtained by introducing a large artificial absorption term which prevents the backscattering of waves into the ocean. The validity of this assumption is still in question(13).

Inaccuracies arise from the small angle approximation and at discontinuities in sound speed and density. Characteristically, the technique has long computation times, although recent versions have seen substantial improvement(13).

The Fast Field Program (14) is an application of Fourier transform techniques to the solution of the inhomogeneous wave equation. The acoustic potential is considered the output of a linear system and is the result of the transform of the product of the transfer function of the medium and the transform of the source waveform. The transfer function of the medium is itself a Fourier-Bessel transform and contains all information about the environment. These integrals are evaluated using the fast Fourier transform.

Computation time is the major problem with the FFP. The Green's function solution must be obtained either by numerical integration or by restricting the sound speed profile to situations for which known solutions are available. In the latter case, recurrence relations must be evaluated to obtain the Green's function. These recurrence relations suffer from numerical instability and are time-consuming to evaluate.

Green's functions obtained by numerical integration are usually sensitive to round-off errors and also substantial computer time. They are constructed by integrating the homogeneous equation for two solutions that independently satisfy the surface and bottom boundary conditions(a similar procedure is used in section 2.1 to obtain an exact solution for a simple shallow water ocean model). The emphasis of this thesis will be on obtaining the Green's function through a state variable representation of the wave equation. This solution technique will be shown to be numerically superior to these two procedures. In addition, when combined with a new technique of evaluating the Hankel transform(15), a substantial improvement in computational efficiency is expected.

A variety of other normal-mode related programs exist which do not fall in any of the above classifications. These include the works of Bucker(9), Stickler(16), and Mckisic and Hamm(17). The first two authors specify the sound speed profileas mathematical functions for which the solution is known. Mckisic and Hamm have applied a new method of solution of eigenvalue problems to the depth-dependent wave equation. The method assumes exponential solutions where the functions in the exponents are specified by Riccati equations. The exponent functions can be obtained by an iteration procedure on the Riccati equations. The additional complexity of the elastic multilayered model is such that the use of any of these techniques in a computer program would be time-consuming and inefficient.

This thesis will study a new technique for evaluating the Green's function solution to the inhomogeneous depth-separated wave equation. The technique involves the use of a state variable representation of the original differential equation that was originally proposed by Baggeroer(18) to obtain the eigenvalues of the homogeneous or normal mode program. Emphasis will be on the study of the technique and its potential in the solution of acoustic propagation problems. The numerical advantages of the representation will be apparent for the oceanic model that includes shear wave propagation in the bottom.

Chapter I will present the mechanics of the state variable system and introduce the continuity conditions required at the boundaries of each layer.

Chapter II will compare the state variable solution with known solutions for two simple shallow water models.

Chapter III will discuss the use of complex sound speeds to model bottom attentuation and investigate an associated numerical instability. It will conclude with a discussion of a deep ocean profile application and a sensitivity problem that arises when the bottom is modelled as a layered elastic medium.

1. DERIVATION OF STATE VARIABLE ALGORITHM

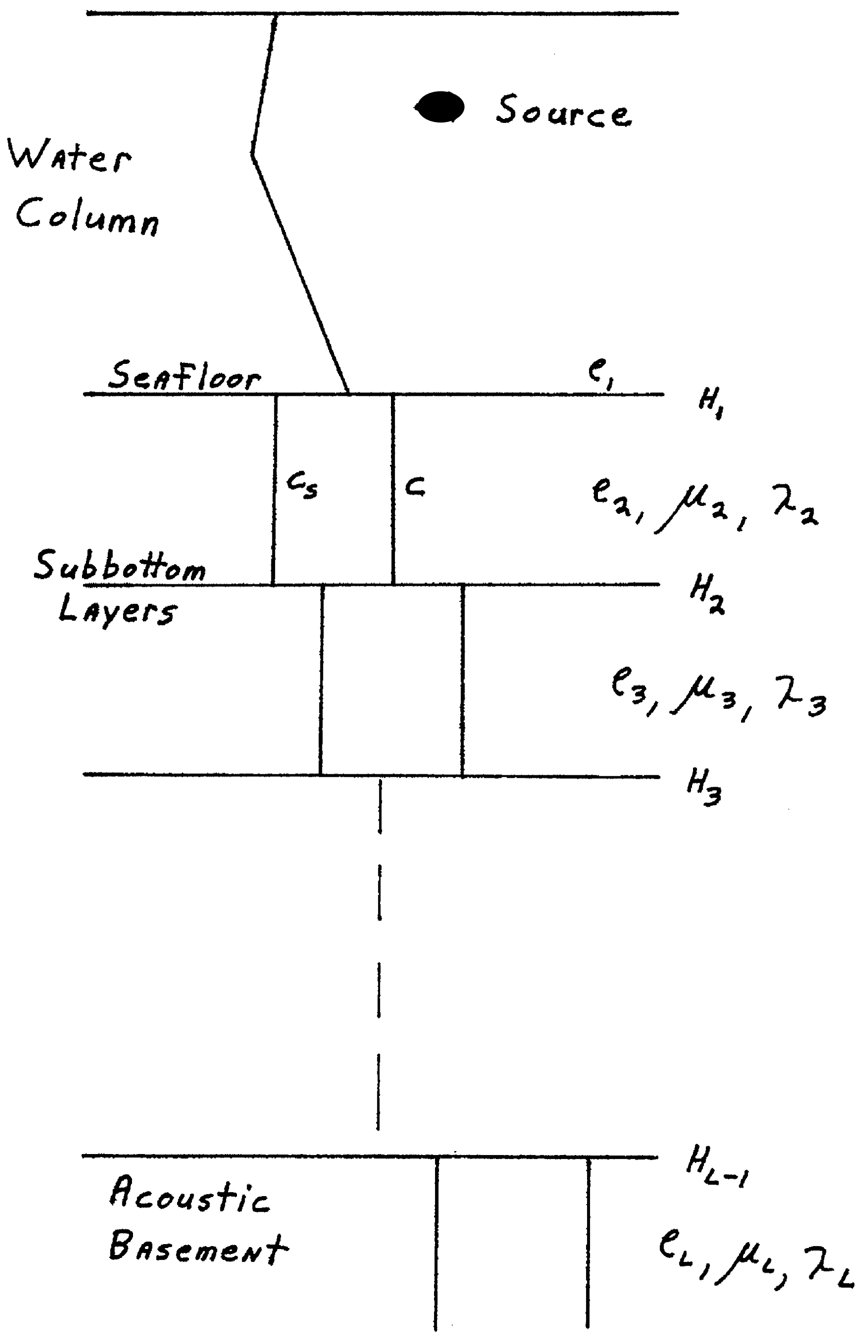
Certain assumptions and definitions are required to simplify the ocean model. These are presented in Section 1. The depth-separated wave equations are then derived and state variable theory used to obtain the state representations. Surface and basement boundary conditions are specified in Section 3, along with the continuity conditions required at the interface between any two layers. Finally, Section 4 introduces a magnitude and phase representation of the state equations that is numerically superior to the linear state equations.

1.1 SPECIFICATION OF THE OCEAN MODEL

For the purpose of this discussion, the ocean is assumed to be an compressible fluid media of constant depth. The seabed is modeled as an elastic medium consisting of a specified number of horizontal, homogeneous isotropic layers in which both compressional and shear waves can propagate. The sound speed of the ocean can vary in the vertical direction. All elastic layers are restricted to have constant sound speeds and elastic properties. Compressional and shear speeds C and C_s are defined in terms of Lame's constant λ, rigidity μ, and density ρ as follows:

$$\lambda + 2\mu = \rho C^2$$
$$\mu = \rho C_s^2$$

(all variables are defined in Appendix I). It should be noted that the technique to be described requires constant elastic parameters only in that the standard derivation of the wave equation for elastic media makes this assumption (19). The final requirement of the model is that coupling



The General Ocean Model

Figure 1.1.1

between compressional and shear waves occurs only at the interface between layers. This coupling is specified by the continuity of vertical and horizontal velocity and normal and shear stress. Figure 1.1.1 depicts the ocean-seabed model.

For convenience, a Cartesian coordinate system is employed. The depth-separated equation that results is identical to that resulting from the use of a cylindrical coordinate system (32). Therefore, the potential of the technique for use with the FFP or with the Hankel transform algorithm of reference (15) can still be evaluated. In addition, the problems associated with expressing the continuity conditions between elastic layers in cylindrical coordinates can be avoided.

1.2 DERIVATION OF THE STATE EQUATIONS

Consider the general form of the acoustic wave equation for a compressional velocity potential $\Phi(x, y, z, t)$

$$\nabla^2 \Phi - \frac{1}{c^2(z)} \frac{\partial^2 \Phi}{\partial t^2} = \delta(\vec{r} - \vec{r}_o) e^{-i\omega t} \qquad 1.2.1$$

Assuming harmonic time dependence and horizontal plane wave propagation (the far field) the substitution

$$\Phi(x, z, t) = e^{i2\pi(\nu x - ft)} \phi(z) \qquad 1.2.2$$

reduces 1.2.1 to

$$\frac{d^2 \phi(z)}{dz^2} + \left[\frac{(2\pi f)^2}{c^2(z)} - (2\pi\nu)^2 \right] \phi(z) = \delta(z - z_o) \qquad 1.2.3$$

Following the same procedure for a shear velocity potential gives the result

$$\frac{d^2\psi(z)}{dz^2} + \left[\frac{(2\pi f)^2}{c_s^2(z)} - (2\pi V_s)^2\right]\psi(z) = 0 \qquad 1.2.4$$

It is useful to define normalized sound speed and horizontal wave number functions as well as a depth parameter that is normalized by the longest wavelength of the sound speed profile (18). The resulting forms of the differential equations are

Compressional:

$$\frac{d^2\phi(\xi)}{d\xi^2} - \left[\gamma^2(\xi) - \tau^2\right]\phi(\xi) = \lambda_o \delta(\xi - \xi_o)$$

$$\xi = \frac{f}{c_o} z = \frac{z}{\lambda_o} \qquad 1.2.5$$

$$\gamma^2(\xi) = (2\pi)^2\left(1 - \frac{c_o^2}{c^2(\xi)}\right)$$

$$\tau^2 = (2\pi)^2(1 - \lambda_o^2 V^2)$$

Shear:

$$\frac{d^2\psi(\xi_s)}{d\xi_s^2} - \left[\gamma_s^2(\xi_s) - \tau_s^2\right]\psi(\xi_s) = 0$$

$$\xi_s = \frac{f}{c_{os}} z = \frac{z}{\lambda_{os}} \qquad 1.2.6$$

$$\gamma_s^2(\xi_s) = (2\pi)^2\left(1 - \frac{c_{os}^2}{c_s^2(\xi_s)}\right)$$

$$\tau_s^2 = (2\pi)^2(1 - \lambda_{os}^2 V_s^2)$$

Note that the depth parameter has been normalized by two different wavelengths corresponding to the compressional and shear maximum wavelengths, respectively. This is necessary because the normalization process requires that

$$\left(\frac{2\pi f}{c(\xi)}\right)^2 - (2\pi \nu)^2 = -\left(\gamma^2(\xi) - \Gamma^2\right)$$

$$\left(\frac{2\pi f}{c_s(\xi)}\right) - (2\pi \nu_s)^2 = -\left(\gamma_s^2(\xi_s) - \Gamma_s^2\right) \qquad 1.2.7$$

Had a single normalized depth parameter been used, both of equations 1.2.7 could not have been satisfied. This becomes clear upon the substitution of 1.2.5 c & d and 1.2.6 c & d into 1.2.7. Since it is preferable to scale 1.2.3 and 1.2.4 into a form which retains their mathematical similarity (i.e., 1.2.5a and 1.2.6a), the independent depth parameters are introduced. Further considerations regarding the use of the two depth parameters will be discussed in Section 1.4.

At this point, it is important to state the relationship between the horizontal wave numbers V and V_s. Consider the case of a plane compressional wave incident from a fluid onto an elastic solid as is depicted in Figure 1.2.1. Potential expressions for the various waves are

$$\Phi_i(x,z) = \exp\left[ik_1(x \sin\theta_i + z\cos\theta_i)\right]$$

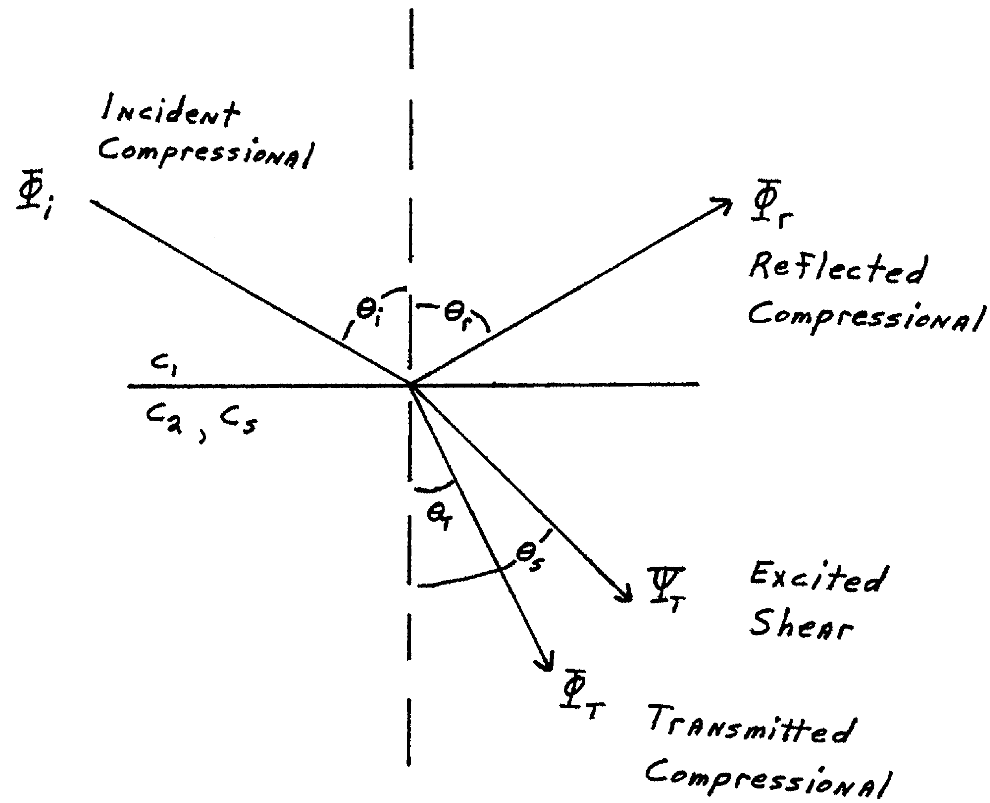
$$\Phi_r(x,z) = \exp\left[ik_1(x \sin\theta_r - z\cos\theta_r)\right]$$

$$\Phi_T(x,z) = \exp\left[ik_2(x \sin\theta_T + z\cos\theta_T)\right]$$

$$\Psi_T(x,z) = \exp\left[ik_s(x \sin\theta_s + z\cos\theta_s)\right]$$

By Snell's law, the wave number parallel to the boundary must be the same for all waves. Hence,

$$k_1 \sin\theta_i = k_1 \sin\theta_r = k_2 \sin\theta_T = k_s \sin\theta_s$$



Snell's Law Relationships

Figure 1.2.1

Since

$$2\pi \nu = k_1 \sin \theta_i$$
$$2\pi \nu_s = k_s \sin \theta_s$$

it follows that $V = V_s$.

Representation of systems in state form is common in the control theory literature (20,21). The general objective is to model the particular process by a system of first order ordinary differential equations of the form

$$\frac{d}{d\xi}\begin{bmatrix} X_1(\xi) \\ \vdots \\ X_n(\xi) \end{bmatrix} = \begin{bmatrix} A_{11}(\xi) & \cdots & A_{1n}(\xi) \\ \vdots & & \\ A_{n1}(\xi) & \cdots & A_{nn}(\xi) \end{bmatrix}\begin{bmatrix} X_1(\xi) \\ \vdots \\ X_n(\xi) \end{bmatrix} + \begin{bmatrix} U_1(\xi) \\ \vdots \\ U_n(\xi) \end{bmatrix} \qquad 1.2.8$$

where $\underline{X}(\xi)$ are the states of the process and $\underline{U}(\xi)$ the control inputs. Since control processes of this form have been well studied, representation of the acoustic wave equation in state form is straightforward.

The first step in the derivation is to define two state variables. The velocity potential, $\phi(\xi)$, being of prime importance here, is specified as the first variable. The second variable, $P(\xi)$, has no direct acoustic significance. The state system is

$$\begin{bmatrix} \dot{\phi}(\xi) \\ \dot{P}(\xi) \end{bmatrix} = \begin{bmatrix} A_{11}(\xi) & A_{12}(\xi) \\ A_{21}(\xi) & A_{22}(\xi) \end{bmatrix}\begin{bmatrix} \phi(\xi) \\ P(\xi) \end{bmatrix} + \begin{bmatrix} U_1(\xi) \\ U_2(\xi) \end{bmatrix} \qquad 1.2.9$$

where the dot refers to the derivative with respect to ξ .

The coefficient matrix $\underline{A}$ is determined by matching 1.2.9 to the original depth-separated wave equation 1.2.5. This is done by taking the derivative of 1.2.9a with respect to ξ and substituting in 1.2.9

for all first order derivatives. The second order equation that results is

$$\ddot{\phi}(\xi) = \left[\dot{A}_{11}(\xi) + A_{11}^2(\xi) + A_{12}(\xi)A_{21}(\xi)\right]\phi(\xi)$$

$$+\left[A_{11}(\xi)A_{21}(\xi) + A_{21}(\xi)A_{22}(\xi) + \dot{A}_{22}(\xi)\right]P(\xi) \quad 1.2.10$$

$$+ A_{12}(\xi)U_2(\xi) + \dot{U}_1(\xi)$$

Matching the coefficients of 1.2.10 with those of 1.2.5a, three equations are obtained for the six unknowns of 1.2.9.

$$\dot{A}_{11}(\xi) + A_{11}^2(\xi) + A_{12}(\xi)A_{21}(\xi) = g^2(\xi) - \tau^2$$

$$A_{11}(\xi)A_{21}(\xi) + A_{21}(\xi)A_{22}(\xi) + \dot{A}_{22}(\xi) = 0 \quad 1.2.11$$

$$A_{12}(\xi)U_2(\xi) + \dot{U}_1(\xi) = \lambda_0\,\delta(\xi-\xi_0)$$

Exact values of the coefficients $A_{ii}(\xi)$ and $U_i(\xi)$ can be chosen in any manner that satisfies all of equations 1.2.11. Particularly convenient is $A_{12}(\xi) = 1$ and $U_1(\xi) = 0$, which reduces 1.2.11 to

$$\dot{A}_{11}(\xi) + A_{11}^2(\xi) + A_{21}(\xi) = g^2(\xi) - \tau^2$$

$$A_{11}(\xi) = -A_{22}(\xi) \quad 1.2.12$$

$$U_2(\xi) = \lambda_0\,\delta(\xi-\xi_0)$$

The well-known Riccati equation can be obtained from 1.2.12a by the substitution $f(\xi) = -A_{11}(\xi)$.

$$\dot{f}(\xi) = f^2(\xi) + A_{21}(\xi) - g^2(\xi) + \tau^2 \quad 1.2.13$$

The above derivation is a repetition in greater detail of reference (18), which also discusses the choice of $A_{21}(\xi)$. That choice is

$$A_{21}(\xi) = -\nu^2 \qquad 1.2.14$$

The compressional state system is

$$\begin{bmatrix} \dot{\phi}(\xi) \\ \dot{p}(\xi) \end{bmatrix} = \begin{bmatrix} -f(\xi) & 1 \\ -\nu^2 & f(\xi) \end{bmatrix} \begin{bmatrix} \phi(\xi) \\ p(\xi) \end{bmatrix} + \begin{bmatrix} 0 \\ \lambda_0 \end{bmatrix} \delta(\xi-\xi_0) \qquad 1.2.15$$

$$\dot{f}(\xi) = f^2(\xi) - \gamma^2(\xi) \qquad 1.2.16$$

Notice that the Riccati equation is a function of the sound speed profile only and is independent of wavenumber. Riccati equations have been well studied and 1.2.16 can be numerically integrated. The coefficient matrix in 1.2.15 is determined by the Riccati solution and the equations can then be integrated to obtain the velocity potential.

The shear state system is

$$\begin{bmatrix} \dot{\psi}(\xi_s) \\ \dot{p}_s(\xi_s) \end{bmatrix} = \begin{bmatrix} -f_s(\xi_s) & 1 \\ -\nu_s^2 & f_s(\xi_s) \end{bmatrix} \begin{bmatrix} \psi(\xi_s) \\ p_s(\xi_s) \end{bmatrix} \qquad 1.2.17$$

$$\dot{f}(\xi_s) = f_s^2(\xi_s) - \gamma_s^2(\xi_s)$$

At this point, the advantages of the state representation are probably not obvious. They become more apparent as the state equations are implemented in a computer program. The simplification from a second to a first order system allows the use of numerical integration techniques which are easier to apply than those for the second order system and for which approximation errors are less crucial. The state systems also allow the analysis of sound speed profiles which are more representative of the real ocean. Finally, the analogy to the FFP is apparent, with the extension to long-range propagation applications theoretically straightforward.

1.3 CONTINUITY CONDITIONS

In general, continuity of nine quantities is required at the interface between two elastic layers. Velocity components comprise three of the quantities and stress components the remaining six. The velocity terms follow from the definition of compressional and shear velocity potentials, which is

$$\bar{V} = (u, v, w) = \nabla\Phi + \nabla \times \Psi \qquad 1.3.1$$

For a homogeneous media in a Cartesian coordinate system with wave propagation occurring only in two dimensions, the "V" velocity and all Y terms can be neglected. The remaining horizontal and vertical velocity terms, neglecting the exponential of equation 1.2.2, are

$$u = \frac{\partial\Phi}{\partial X} - \frac{\partial\Psi}{\partial Z} = i2\pi V\phi(\xi) - \frac{1}{Z_{os}}\frac{d\,\psi(\xi_s)}{d\,\xi_s}$$
$$w = \frac{\partial\Phi}{\partial Z} + \frac{\partial\Psi}{\partial X} = \frac{1}{Z_o}\frac{d\phi(\xi)}{d\xi} + i2\pi V\psi(\xi_s) \qquad 1.3.2$$

Since equations 1.2.5 and 1.2.6 were scaled using different normalized depth variables, notice that ϕ and ψ are functions of ξ and ξ_s, respectively. The integration size for ϕ and ψ will in general be different. However, the coupling at each interface requires that they be expressed in the continuity conditions with respect to the same coordinate. Since

$$\frac{d\phi}{dz} = \frac{d\phi}{d\xi}\frac{d\xi}{dz} = \frac{1}{\lambda_o}\frac{d\phi}{d\xi}$$

$$\frac{d\psi}{dz} = \frac{d\psi}{d\xi_s}\frac{d\xi_s}{dz} = \frac{1}{\lambda_{os}}\frac{d\psi}{d\xi_s}$$

this has in fact been done in 1.3.2.

For an isotropic media, the six stress components can be written in terms of two elastic parameters: the rigidity μ and the Lamé constant λ. The stress expressions, written in terms of the velocity components and the dilation θ , are

$$\begin{aligned}
\sigma_{XX} &= \lambda\theta + 2\mu\frac{du}{dX} \\
\sigma_{YY} &= \lambda\theta + 2\mu\frac{dv}{dY} \\
\sigma_{zz} &= \lambda\theta + 2\mu\frac{dw}{dz} \\
\sigma_{XY} &= \mu\left(\frac{du}{dY} + \frac{dv}{dX}\right) \\
\sigma_{Yz} &= \mu\left(\frac{dv}{dz} + \frac{dw}{dY}\right) \\
\sigma_{zX} &= \mu\left(\frac{dw}{dX} + \frac{du}{dz}\right) \\
\theta &= \frac{du}{dX} + \frac{dv}{dY} + \frac{dw}{dz}
\end{aligned} \qquad 1.3.3$$

The assumptions of the model reduce the above six expressions to the normal stress σ_{zz} and the tangential stress σ_{zX} . Expressed in potential form, they are

$$\sigma_{zz} = \lambda(i2\pi\nu)^2\phi(\xi) + \frac{\lambda+2\mu}{\lambda_o^2}\frac{d^2\phi(\xi)}{d\xi^2} + \frac{\mu i4\pi\nu}{\lambda_{os}}\frac{d\psi}{d\xi_s}$$

$$\sigma_{zX} = \mu\left[\frac{i4\pi\nu}{\lambda_o}\frac{d\phi(\xi)}{d\xi} + (i2\pi\nu)^2\psi(\xi_s) - \frac{1}{\lambda_{os}^2}\frac{d^2\psi(\xi_s)}{d\xi_s^2}\right] \qquad 1.3.4$$

It is important to note that the reduction of 1.3.1 to 1.3.2 and 1.3.3 to 1.3.4 is possible because of the use of an X, Y, Z geometry and the homogeneous, isotropic media assumption (33). Extension to a cylindrical geometry, as may be desirable for future applications, will require reconsideration of the velocity and stress equations.

The result of these simplifications is that there are four quantities that must be continuous across the interface between any two elastic layers. The formal expression of these continuity conditions, as shown in Figure 1.3.1, is

$$\begin{aligned} u_i(H_i) &= u_{i+1}(H_i) \\ w_i(H_i) &= w_{i+1}(H_i) \\ \sigma_{zz_i}(H_i) &= \sigma_{zz_{i+1}}(H_i) \\ \sigma_{zx_i}(H_i) &= \sigma_{zx_{i+1}}(H_i) \end{aligned} \qquad 1.3.5$$

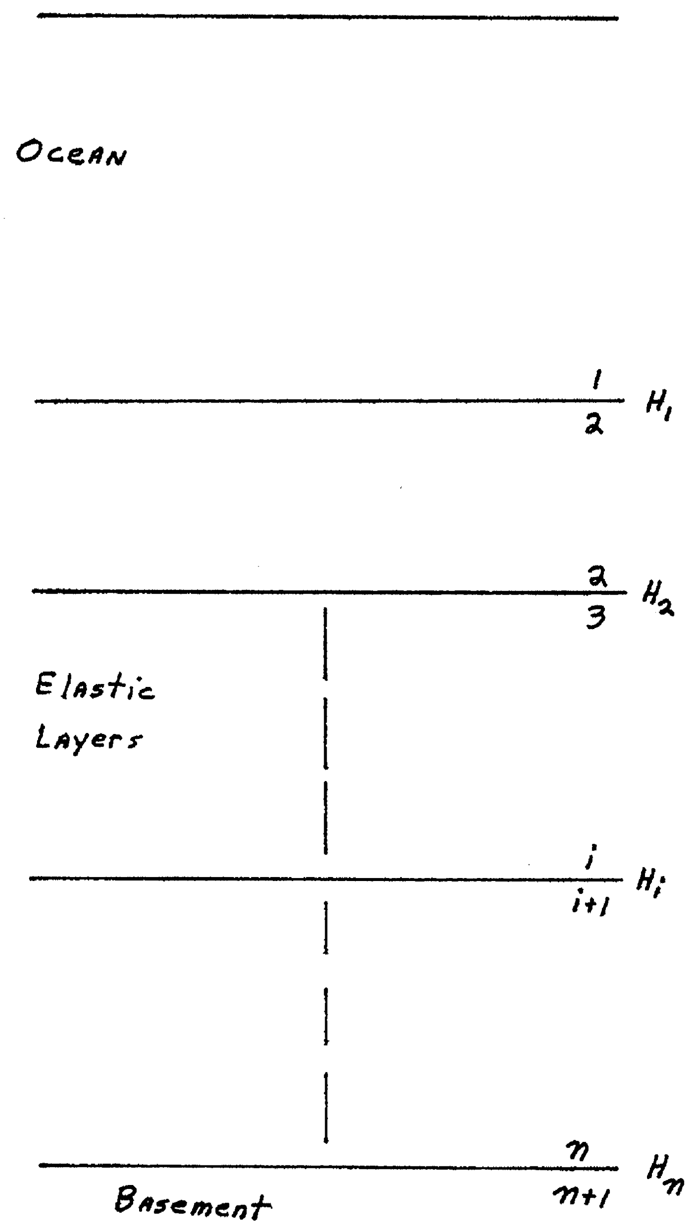
At the interface between the ocean and the uppermost elastic layer, a simplification of these conditions is in order. First, since the fluid is not allowed to hold shear, the tangential stress in the solid must vanish. Second, continuity of horizontal velocity is not required since slippage can occur. The formal expressions are

$$\begin{aligned} u_1(H_1) &= u_2(H_1) \\ \sigma_{zz_1}(H_1) &= \sigma_{zz_2}(H_1) \\ 0 &= \sigma_{zx_2}(H_1) \end{aligned} \qquad 1.3.6$$

The boundary condition at the ocean's surface is that the normal stress must vanish. From 1.3.4a and 1.2.5a, the expression for this condition is

$$\phi(0) = 0 \qquad 1.3.7$$

The bottom boundary condition can take either of two forms. For an N^{th} layer bounded by a rigid surface, both velocity components must vanish.



Continuity Conditions

Figure 1.3.1

$$i 2\pi \nu \phi\left(\frac{H_m}{\lambda_o}\right) - \frac{1}{\lambda_{os}} \frac{d \psi\left(\frac{H_m}{\lambda_{os}}\right)}{d \xi_s} = 0$$

$$\frac{1}{\lambda_o} \frac{d \phi\left(\frac{H_m}{\lambda_o}\right)}{d \xi} + i 2\pi \nu \psi\left(\frac{H_m}{\lambda_{os}}\right) = 0 \qquad 1.3.8$$

For an N^{th} layer bounded by a propagating, infinite elastic half space, constrained motions are required.

$$|\phi(\xi)| < M < \infty \qquad \xi \rightarrow \infty$$

$$|\psi(\xi_s)| < N < \infty \qquad \xi_s \rightarrow \infty \qquad 1.3.9$$

The boundary value problem has now been completely specified except for the exact nature of the solution. The homogeneous shear state equations are excited at the ocean bottom by the coupling of $\phi(\xi)$ and $\psi(\xi_s)$, expressed by 1.3.6, and the shear solution consists solely of the integration of 1.2.17 satisfying 1.3.6 and either 1.3.8 or 1.3.9. The compressional state equations are inhomogeneous and their solution must consist of some combination of homogeneous and particular terms which satisfy 1.3.7 and either 1.3.8 or 1.3.9. Formally, the solutions are

Compressional:

$$\phi(\xi) = \phi_P(\xi) + \phi_H(\xi)$$

$$P(\xi) = P_P(\xi) + P_H(\xi) \qquad 1.3.10$$

Shear:

$$\psi(\xi) = \psi_H(\xi_s)$$

$$P_s(\xi) = P_{sH}(\xi_s)$$

The boundary conditions for the particular solution depend on the integration direction. Baggeroer (18) discusses the convenience of integrating up the profile, and the resulting choice of

$$\phi_p\left(H_n/z_0\right) = 0$$
$$P_p\left(H_n/z_0\right) = 0 \qquad 1.3.12$$

These initial conditions result in a particular solution that is identically zero below the source.

Equations 1.3.12 allow expression of the bottom boundary conditions solely in terms of the homogeneous potentials. For the rigid bottom case

$$i2\pi\nu\,\phi_H\left(H_n/z_0\right) - \frac{1}{z_{0s}}\,\frac{\partial \psi_H\left(H_n/z_{0s}\right)}{\partial \xi_s} = 0$$
$$\frac{1}{z_0}\,\frac{\partial \phi_H\left(H_n/z_0\right)}{\partial \xi} + i2\pi\nu\,\psi_H\left(\frac{H_n}{z_{0s}}\right) = 0 \qquad 1.3.13$$

For the propagating basement model, the assumption of constant sound speed allows an analytic solution to be obtained. That solution is

$$\phi_H\left(\xi > \frac{H_n}{z_0}\right) = \frac{1}{2}\left[\phi_H\left(\frac{H_n}{z_0}\right) - \frac{1}{r}\,\frac{\partial \phi_H\left(\frac{H_n}{z_0}\right)}{\partial \xi}\right] e^{-r\left(\xi - \frac{H_n}{z_0}\right)}$$
$$+ \frac{1}{2}\left[\phi_H\left(\frac{H_n}{z_0}\right) + \frac{1}{r}\,\frac{\partial \phi_H\left(\frac{H_n}{z_0}\right)}{\partial \xi}\right] e^{r\left(\xi - \frac{H_n}{z_0}\right)}$$
$$\psi_H\left(\xi_s > \frac{H_n}{z_{0s}}\right) = \frac{1}{2}\left[\psi_H\left(\frac{H_n}{z_{0s}}\right) - \frac{1}{r}\,\frac{\partial \psi_H\left(\frac{H_n}{z_{0s}}\right)}{\partial \xi_s}\right] e^{-r\left(\xi_s - \frac{H_n}{z_{0s}}\right)} \qquad 1.3.14$$
$$+ \frac{1}{2}\left[\psi_H (\frac{H_n}{z_{0s}} + \frac{1}{r}\,\frac{\partial \psi_H\left(\frac{H_n}{z_{0s}}\right)}{\partial \xi_s}\right] e^{r\left(\xi_s - \frac{H_n}{z_{0s}}\right)}$$

From 1.3.9

$$\phi_H\left(\frac{H_n}{z_0}\right) + \frac{1}{r}\,\frac{\partial \phi_H\left(\frac{H_n}{z_0}\right)}{\partial \xi} = 0$$
$$\psi_H\left(\frac{H_n}{z_{0s}}\right) + \frac{1}{r_s}\,\frac{\partial \psi_H\left(\frac{H_n}{z_{0s}}\right)}{\partial \xi_s} = 0 \qquad 1.3.15$$

Both 1.3.13 and 1.3.15 contain four unknowns and therefore have two apparently arbitrary constants. As integration proceeds upward, however, these constants are specified by other requirements of the model. In particular, equation 1.3.6c, which must be satisfied at the ocean-sediment interface, determines the third unknown by expressing the required relationship between the compressional and shear fields. The final unknown is specified by the surface boundary condition. Recall that the particular solution can be integrated using 1.3.12. Therefore, from 1.3.7,

$$\phi_H(0) = -\phi_P(0) \qquad 1.3.16$$

and the boundary value problem is completely specified.

Since the boundary conditions 1.3.6c, 1.3.7, and 1.3.13 or 1.3.15 are split (that is, evaluated at different points in the depth profile), it is useful to consider the general step-by-step procedure to be employed in solving the boundary value problem. First, the particular solution is obtained. For the initial conditions of 1.3.12, $\phi_P(\zeta)$ and $P_P(\zeta)$ will be non-zero from the source point to the surface. Second, two unknowns are chosen from the four in 1.3.15 (for a propagating basement) and given unit amplitude. For example,

$$\begin{aligned} \phi_H\left(\frac{H_n}{\lambda_o}\right) &= 1 \\ \psi_H\left(\frac{H_n}{\lambda_{os}}\right) &= 1 \end{aligned} \qquad 1.3.17$$

For simplicity in satisfying 1.3.6c, the third step is to integrate upward using only the compressional field in the basement as the initial condition (assuming for the moment zero shear at H_n). Compressional and shear fields will be excited in all elastic layers and integration will proceed to the uppermost interface of the profile where 1.3.6c will not in general be satisfied. Next, the shear field is used as the initial condition at H_n (with a zero compressional field) and integration again will proceed upward to the uppermost interface.

At this point, two independent solutions will have been obtained for the elastic layers, neither of which satisfies 1.3.6c. The linearity of 1.2.15 and 1.2.17 allow superposition of these solutions in such a manner as to satisfy 1.3.6c. This superposition amounts to a scaling of the shear solution to insure that the tangential stress at H_1 vanishes. This procedure specifies the third constant as was discussed earlier.

Equations 1.3.6 a & b are used to obtain the appropriate compressional field quantities in the ocean and integration proceeds to the surface to obtain $\phi_H(0)$. The superposition used earlier determined the relationship between the compressional and shear fields but not the absolute magnitude of either (the fourth constant) and in general 1.3.16 will not be satisfied. Scaling of the entire compressional and shear solutions by the constant

$$C = -\frac{\phi_P(0)}{\phi_H(0)} \qquad 1.3.18$$

will insure that the surface boundary condition is satisfied.

The compressional and shear solutions are

$$\phi(\zeta) = \phi_P(\zeta) - \frac{\phi_P(0)}{\phi_H(0)}\,\phi_H(\zeta) \qquad 1.3.19$$

$$\psi(\xi) = -\frac{\phi_P(0)}{\phi_H(0)}\psi_H(\xi_s) \qquad 1.3.20$$

This completes the specification of the boundary value problem. The equations can now be programmed and solutions obtained for any ocean and seabed parameters. It has been found, however, that a representation independent of absolute magnitude is more convenient numerically. This representation will be presented in Section 1.4.

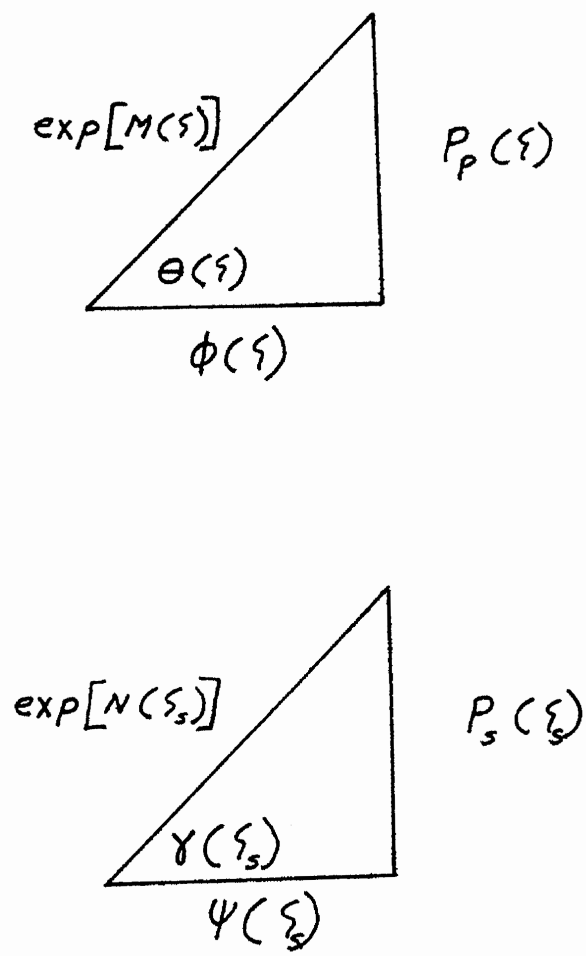
1.4 PHASE PLANE REPRESENTATION

The state equations of Section 1.3 can be integrated to obtain the solution for any given ocean model. However, the exponential nature of the solution causes large amplitude terms to arise, especially in deep ocean examples. Baggeroer (18) introduced a magnitude and phase representation of the state equations which is numerically preferable. This representation allows integration of differential equations which are independent of the absolute magnitude of $\phi(\xi)$ and $\psi(\xi_s)$. As shown in Figure 1.4.1, the respective variables are defined

$$\theta(\xi) = \mathrm{TAN}^{-1}\left(\frac{P(\xi)}{\phi(\xi)}\right) \qquad 1.4.1$$

$$M(\xi) = \frac{1}{2}\ln\left[\phi^2(\xi) + P^2(\xi)\right] \qquad 1.4.2$$

$$\gamma(\xi_s) = \mathrm{TAN}^{-1}\left(\frac{P_s(\xi_s)}{\psi(\xi_s)}\right) \qquad 1.4.3$$



Phase Plane Relationships

Figure 1.4.1

$$N(\xi_s) = \frac{1}{2} \ln \left[\psi^2(\xi_s) + P_s^2(\xi_s) \right] \qquad 1.4.4$$

Under this transformation, the state equations can be derived to be

$$\dot{\theta}(\xi) = 2f(\xi) \cos\theta(\xi) \sin\theta(\xi) - \sigma^2 \cos^2\theta(\xi) - \sin^2\theta(\xi) \qquad 1.4.5$$

$$\dot{M}(\xi) = f(\xi)\left[\sin^2\theta(\xi) - \cos^2\theta(\xi)\right] + (1-\sigma^2) \sin\theta(\xi) \cos\theta(\xi) \qquad 1.4.6$$

$$\dot{\gamma}(\xi_s) = 2f_s(\xi_s) \cos\gamma(\xi_s) \sin\gamma(\xi_s) - \sigma_s^2 \cos^2\gamma(\xi_s) - \sin^2\gamma(\xi_s) \qquad 1.4.7$$

$$\dot{N}(\xi_s) = f_s(\xi_s)\left[\sin^2\gamma(\xi_s) - \cos^2\gamma(\xi_s)\right] + (1-\sigma_s^2) \sin\gamma(\xi_s) \cos\gamma(\xi_s) \qquad 1.4.8$$

The surface and propagating basement boundary conditions become

$$\theta(0) = \frac{\pi}{2} \qquad 1.4.9$$

$$\theta\left(\frac{H_n}{\eta_o}\right) = \tan^{-1}\left[f\left(\frac{H_n}{\eta_o}\right) - f^2\left(\frac{H_n}{\eta_{os}}\right) - \sigma^2\right] \qquad 1.4.10$$

$$\gamma\left(\frac{H_n}{\eta_{os}}\right) = \tan^{-1}\left[f_s\left(\frac{H_n}{\eta_{os}}\right) - f^2\left(\frac{H_n}{\eta_{os}}\right) - \sigma_s^2\right] \qquad 1.4.11$$

The rigid bottom boundary conditions 1.3.13 and 1.3.5 and 1.3.6 do not transform into convenient magnitude and phase expressions. When they are required, the simplest procedure is to return to the linear plane, obtain the desired result and transform back to the phase plane.

Implementation of these equations is straightforward. The particular solution is integrated in the linear plane. The homogeneous solutions are obtained in the phase plane and the total solution is given by

$$\psi(\xi_s) = e^{N(\xi_s)} \cos \gamma(\xi_s) \qquad 1.4.12$$

$$\phi(\xi) = \phi_p(\xi) - \frac{\phi_p(0)}{e^{M(0)} \cos \theta(0)} e^{M(\xi)} \cos \theta(\xi) \qquad 1.4.13$$

Notice that the scaling constant of 1.3.18 has been included in 1.4.13.

In the following chapter, these equations will be integrated for two specific ocean-seabed models and compared with known Green's function solutions.

2. CONFIRMATION OF STATE VARIABLE ALGORITHM

At any given wavenumber, the Green's function solution to the wave equation is equivalent to the superposition of an interfering set of up and down travelling plane waves. Construction of the solution by an appropriate combination of plane waves will provide an accurate test of the theory of Chapter I. In Chapter II an endpoint method of constructing the Green's function will be used to obtain the exact solution of the wave equation for two simple shallow water oceanic models. The state variable solution will then be compared to the exact solution.

2.1 SINGLE LAYER OCEAN EXACT SOLUTION

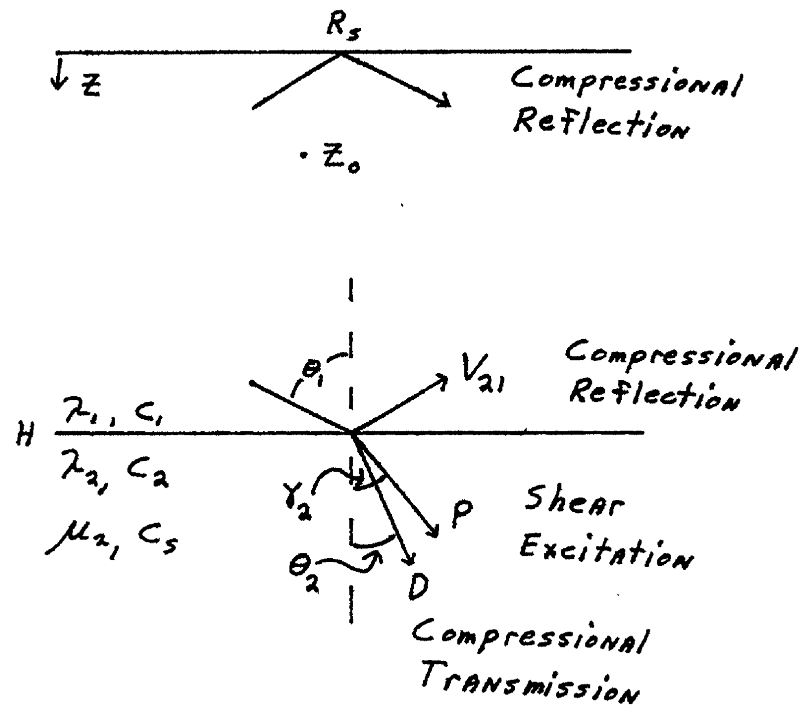
An exact solution of the inhomogeneous depth-separated wave equation can be written for the simple model of a constant sound speed fluid layer overlying an infinite elastic half space. Figure 2.1.1 depicts this model, which is commonly referred to as the generalized Pekeris waveguide.

The assumption of a pressure release surface at $z = 0$ reduces R_s, the surface reflection coefficient, to the value -1. The remaining three coefficients do not in general reduce to such a convenient value, and must be written in terms of the appropriate elastic parameters and wave numbers. For a fluid with $K_1 = \frac{2\pi f}{c_1}$, the horizontal and vertical wave numbers are defined in terms of the angle of incidence θ_1 as follows:

$$2\pi f = K_1 \sin\theta_1$$

$$K_{z1} = K_1 \cos\theta_1$$

The corresponding expressions in the bottom are



Single Layer Ocean Description

Figure 2.1.1

$$2\pi\nu = K_2 \sin\theta_2 \qquad 2\pi\nu = K_{2s}$$

$$K_{z2} = K_2 \cos\theta_2 \qquad K_{z2s} = K_{2s} \cos\gamma_2$$

The angles θ_1, θ_2, and γ_2 are measured from the z-axis to the respective propagation vectors as shown in Figure 2.1.1. Brekhovskikh(25) writes the reflection and transmission coefficients in terms of the impedances z_i of the media. His expressions are:

$$V_{21} = \frac{Z_2 \cos^2 2\gamma_2 + Z_{2s} \sin^2 2\gamma_2 - Z_1}{Z_2 \cos^2 2\gamma_2 + Z_{2s} \sin^2 2\gamma_2 + Z_2} \qquad 2.1.1$$

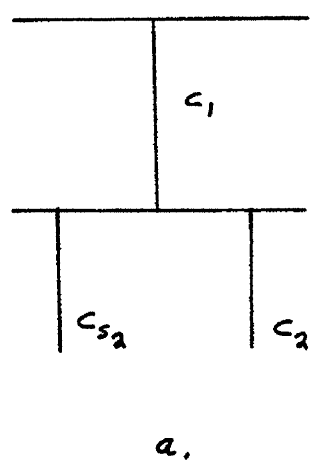
$$D = \frac{\rho_1}{\rho_2} \frac{2 Z_2 \cos 2\gamma_2}{Z_2 \cos^2 2\gamma_2 + Z_{2s} \sin^2 2\gamma_2 + Z_2} \qquad 2.1.2$$

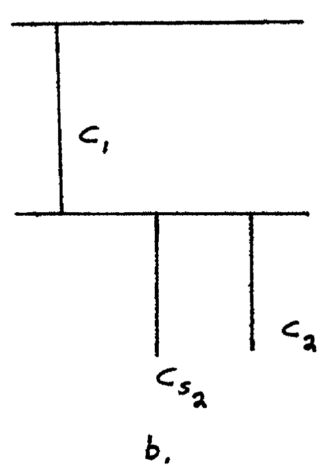
$$P = -\frac{\rho_1}{\rho_2} \frac{2 Z_{s2} \sin 2\gamma_2}{Z_2 \cos^2 2\gamma_2 + Z_{2s} \sin^2 2\gamma_2 + Z_2} \qquad 2.1.3$$

$$Z_1 = \frac{\rho_1 c_1}{\cos\theta_1} \qquad Z_2 = \frac{\rho_2 c_2}{\cos\theta_2} \qquad Z_{s2} = \frac{\rho_2 c_{s_2}}{\cos\gamma_2} \qquad 2.1.4$$

The common oceanic model has $C_2 > C_1$. The magnitude of the reflection coefficient V_{21} can be characterized by considering the various wave number domains that occur for $C_{s_2} < C_1$ and for $C_{s_2} > C_1$ ($C_{s_2} < C_2$).

For the case $C_{s_2} < C_1 < C_2$, Figure 2.1.2a, the magnitude of V_{21} is less than 1.0 for all real incident angles θ_1, since from Snell's law γ_2 is also real and the resulting shear wave in the bottom is propagating. The presence of a propagating wave, compressional or shear, in the bottom signifies energy lost from the water column and hence $|V_{21}| < 1.0$. When $\theta_1 > \theta_c = \sin^{-1} \frac{c_1}{c_2}$, the compressional wave in the bottom is





Sound Speed Relationships

Figure 2.1.2

inhomogeneous and the shear wave propagating. For $\theta_1 < \theta_c$ both waves are propagating.

For the case $C_1 < C_{s_2} < C_2$, Figure 2.1.2b, total internal reflection $|V_{21}| = 1$ occurs for $\theta_1 > \sin^{-1}\frac{C_1}{C_{s_2}}$. Both the compressional and the shear waves in the bottom are inhomogeneous. When $\sin^{-1}\frac{C_1}{C_{s_2}} > \theta_1 > \sin^{-1}\frac{C_1}{C_2}$, $|V_{21}| < 1$ and shear propagates in the bottom. For $\theta_1 < \sin^{-1}\frac{C_1}{C_2}$ both waves again propagate and $|V_{21}| < 1$. Typical plots of the reflection and transmission coefficients are given in (25) and (33).

In general, for the differential equation

$$\frac{d^2G}{dz^2} + k_z^2 G = \delta(z - z_0) \qquad 0 \le z \le H \tag{2.1.5}$$

the solution obtained via the endpoint method (24,30) has the form

$$G(z,z_0) = \begin{cases} G_a(z,z_0) = W^{-1} u_a(z)\, u_b(z_0) & 0 \le z \le z_0 \\ G_b(z,z_0) = W^{-1} u_a(z_0)\, u_b(z) & z_0 \le z \le H \end{cases} \tag{2.1.6}$$

where W is the Wronskian

$$W(z_0) = u_a(z_0)\frac{du_b(z_0)}{dz} - \frac{du_a(z_0)}{dz}u_b(z_0) \tag{2.1.7}$$

$U_a(z)$ and $U_b(z)$, which satisfy the upper and lower boundary conditions respectively, are linearly independent solutions of the homogeneous equation . The compressional and shear solutions in the lower half space satisfy the equations

Compressional: $$\frac{d^2 G_2}{dz^2} + k_{z2}^2 G_2 = 0 \qquad z \geq H \tag{2.1.8}$$

Shear: $$\frac{d^2 G_{2s}}{dz^2} + k_{z2s}^2 G_{2s} = 0 \qquad z \geq H \tag{2.1.9}$$

and can be written as

$$G_2(z, z_o) = W^{-1} U_a(z_o) U_2(z) \qquad z \geq H \tag{2.1.10}$$

$$G_{2s}(z, z_o) = W^{-1} U_a(z_o) U_{2s}(z) \qquad z \geq H \tag{2.1.11}$$

$U_2(z)$ and $U_{2s}(z)$ are solutions of 2.1.8 and 2.1.9 that satisfy the boundary condition at $z = H$ and the radiation condition as $z \to \infty$. Note that the solutions 2.1.10 and 2.1.11 are not, strictly speaking, Green's functions. The notation G_2 and G_{2s} has been employed for convenience. For the single layered ocean of Figure 2.1.1, the various expressions are

$$U_a(z) = e^{-ik_{z1}z} - e^{+ik_{z1}z} \qquad 0 \leq z \leq z_o \tag{2.1.12}$$

$$U_b(z) = e^{ik_{z1}z} + V_{21}\, e^{2ik_{z1}H} e^{-ik_{z1}z} \qquad z_0 \le z \le H \qquad 2.1.13$$

$$U_2(z) = D e^{ik_{z2}z} \qquad H \le z \qquad 2.1.14$$

$$U_{2s}(z) = P e^{ik_{z2s}z} \qquad H \le z \qquad 2.1.15$$

$$G_a(z,z_0) = -2i\,\mathrm{Sin}(k_{z1}z)\left[e^{ik_{z1}z_0} + V_{21}\, e^{2ik_{z1}H} e^{-ik_{z1}z_0}\right] W^{-1} \qquad 2.1.16$$

$$0 \le z \le z_0$$

$$G_b(z,z_0) = -2i\,\mathrm{Sin}(k_{z1}z)\left[e^{ik_{z1}z} + V_{21}\, e^{2ik_{z1}H} e^{-ik_{z1}z}\right] W^{-1} \qquad 2.1.17$$

$$z_0 \le z \le H$$

$$G_2(z,z_0) = -2i\,\mathrm{Sin}(k_{z1}z_0)\left[D\, W^{-1} e^{ik_{z1}H} e^{ik_{z2}(z-H)}\right] \qquad 2.1.18$$

$$z \ge H$$

$$G_{2s}(z,z_0) = -2i\,\mathrm{Sin}(k_{z1}z_0)\left[P W^{-1} e^{ik_{z1}H} e^{ik_{z2s}(z-H)}\right] \qquad 2.1.19$$

$$z \ge H$$

$$W^{-1} = 2k_{z1}\Big[\mathrm{Sin}\,k_{z1}z_0\left(e^{ik_{z1}z_0} - V_{21}\, e^{2ik_{z1}H} e^{-ik_{z1}z_0}\right) \qquad 2.1.20$$

$$+ i\,\mathrm{Cos}\,k_{z1}z_0\left(e^{ik_{z1}z_0} + V_{21}\, e^{2ik_{z1}H} e^{-ik_{z1}z}\right)\Big]$$

Equations 2.1.16 through 2.1.20 comprise the exact solution for the model of Figure 2.1.1. As was discussed earlier, the compressional and shear waves in the bottom are either propagating or exponentially decaying (inhomogeneous) depending on the incident angle θ_1. It should also be noted that the Wronskian is identically zero at discrete mode wavenumbers and therefore the Green's function solution diverges in those cases. Perfectly trapped discrete modes occur when $C_1 < S_2 < C_2$ and are limited to the wavenumber domain where $|V_{21}| = 1.0$. Section 2.2 will compare the above solution with the state variable solution.

2.2 SINGLE LAYER OCEAN SOLUTION COMPARISONS

Realistic values of sound speeds and elastic parameters for Figure 2.1.1 can be obtained from the geophysical literature. Hamilton (28,29) suggests this model as common in continental shelf areas. The table below details the information to be used in this section. These parameters correspond to an ocean bottom of sand or silty-sand of unknown depth.

Frequency	100 hz
Source Depth	30 m
C_1	1500 m/s
λ_1	$2.25 \cdot 10^{-5}$ dynes/cm^2
C_2	1675 m/s
λ_2	$4.80125 \cdot 10^{-5}$ dynes/cm^2
S_2	450 m/s

μ	$0.405 \cdot 10^{-5}$ dynes cm^2
Depth	100 m

The water and bottom wave numbers for this example are

$$V_W = 0.0667 \ m^{-1}$$

$$V_B = 0.0597 \ m^{-1}$$

$$V_{BS} = 0.2222 \ m^{-1}$$

Solutions to 1.2.5 for wave numbers V such that $V_W > B > V_B$ will consist of sinusoids in the fluid layer and decaying exponentials in the bottom. As discussed earlier, there are no perfectly trapped discrete modes for this model, since the parameters and sound speeds correspond to Figure 2.1.2c. The shear contribution is so small, however, that the response functions, when computed at the modal wavenumbers of the equivalent two fluid case, will appear unchanged from the mode shapes. A discontinuity in the slope of the response function will occur at the source for all other wave numbers. For wave numbers $V < V_B$ a propagating compressional wave will be excited in the bottom. Since $V_{BS} > V_{H_2O}$, a propagating shear wave will be generated in the bottom.

Figures 2.2.1, 2.2.2, 2.2.3, and 2.2.4 display the response functions obtained from the two techniques for the horizontal wave numbers V = 0.066, 0.061, 0.059, and 0.055 (the term response function is defined as the response of the given ocean model at the horizontal wavenumber V to a source at depth z_0 and is synonymous with the term Green's function.) Agreement between the two solution techniques is quite good in all cases. Notice a slight decrease in accuracy as V decreases. This phenomenon

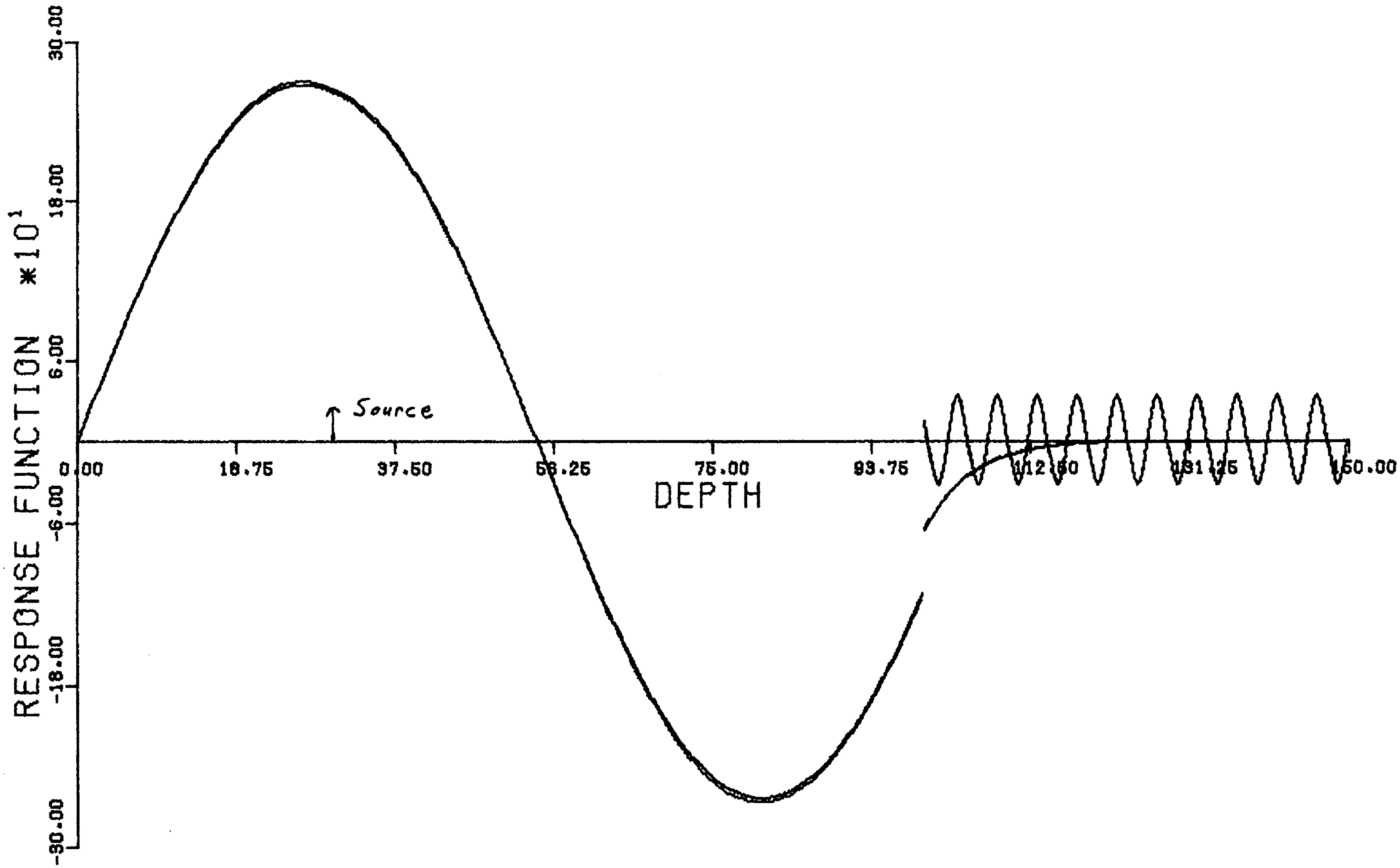


Figure 2.2.1. Response function for single layer ocean model with $V=0.066m^{-1}$ Shear field plotted five times actual magnitude. Basement depth =100m. Source depth= 30m.

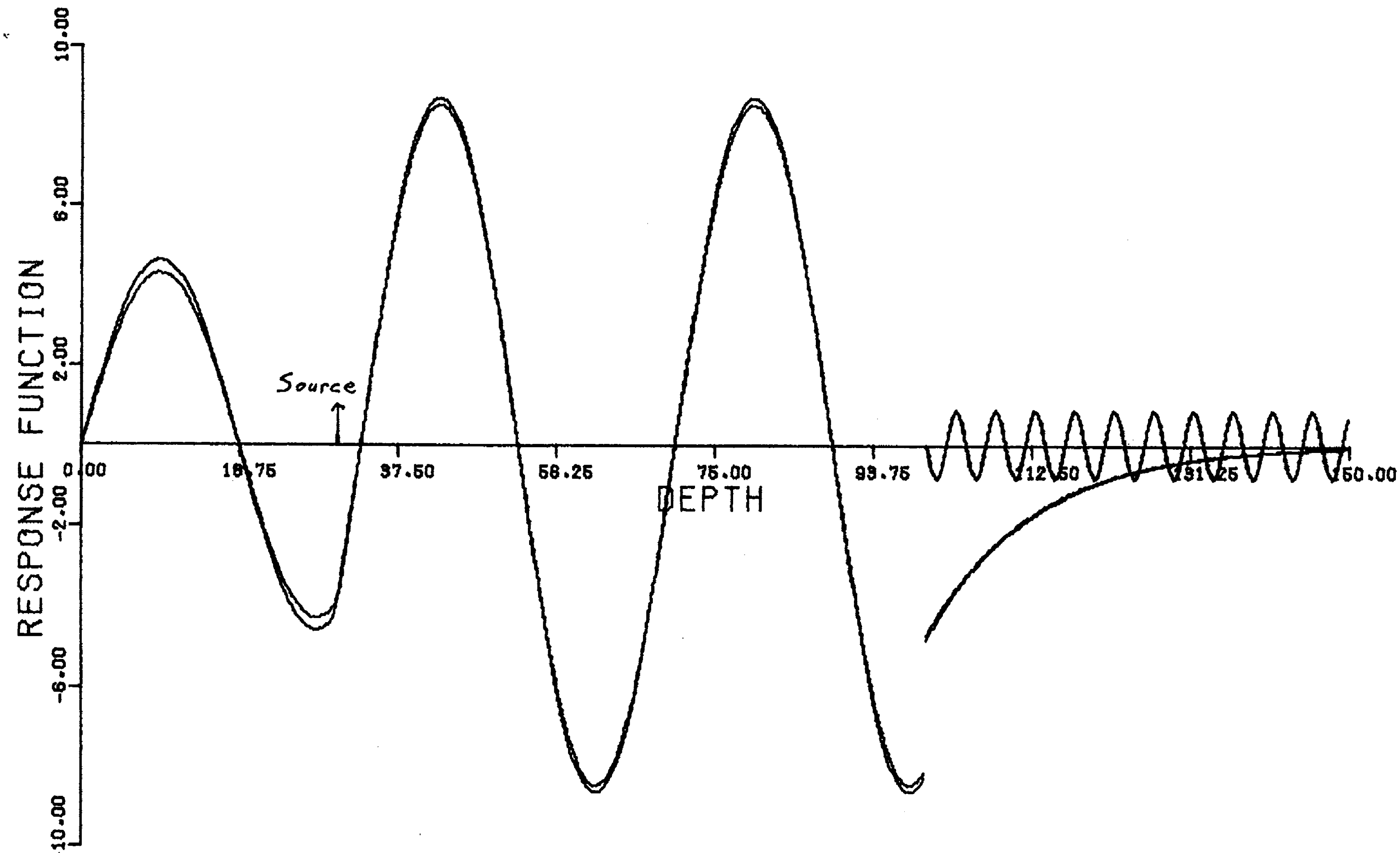


Figure 2.2.2. Response function for single layer ocean model with V=0.061m^{-1} Shear field plotted five times actual magnitude. Basement depth =100m. source depth= 30m.

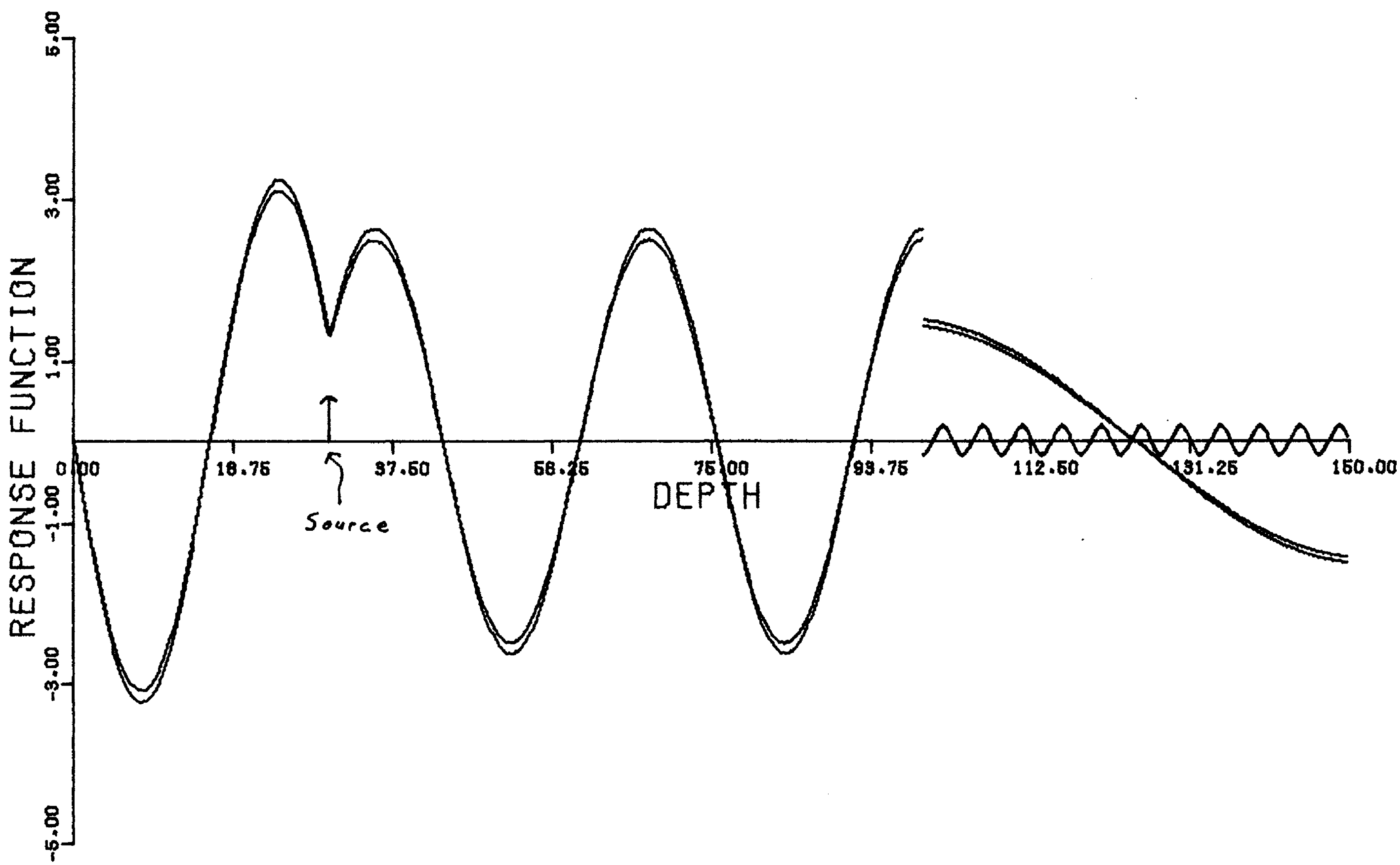


Figure 2.2.3. Response function for single layer ocean model with $V=0.059m^{-1}$ Shear field plotted five times actual magnitude. Basement depth =100m. Source depth= 30m.

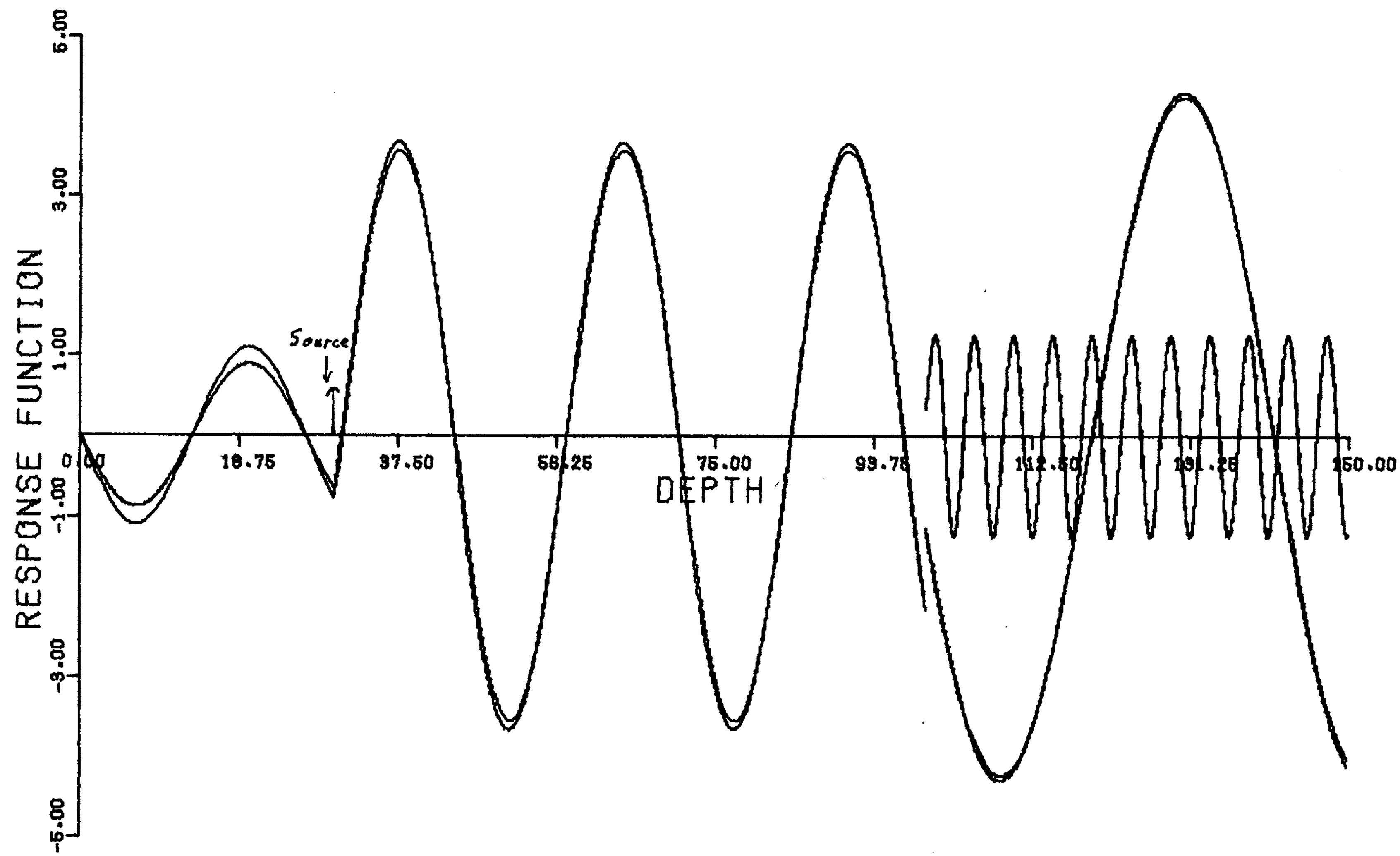


Figure 2.2.4 Response function for single layer ocean model with $V=0.055m^{-1}$ Shear field plotted five times actual magnitude. Basement depth =100m. Source depth= 30m.

is coupled with the integration grid density and will be discussed in the next section.

2.3 INTEGRATION GRID DENSITY REQUIREMENTS

Section 2.2 showed that the state variable technique does produce accurate results. To determine the efficiency of the algorithm, the integration grid size that is required to produce these solutions must be investigated. Minimum computer time will occur when the number of integration points is minimized.

The Nyquist sampling theorem of communication theory states the relationship between the sampling period of a process and the minimum period present in the data to be sampled that is required to enable exact reconstruction of the data. A comparable relationship for this application would be one which specifies the number of integration points required per spatial wavelength to insure solution accuracy. Defining the vertical spatial wavelength λ_s as

$$\lambda_s = (2\pi)^{-1}\left(\frac{f^2}{c_o^2} - \nu^2\right)^{-1/2} \qquad 2.3.1$$

the number of integration points per wavelength is defined as

$$N_\lambda = \frac{\lambda_s N_I}{H_N} \qquad 2.3.2$$

where N_I is the total number of integration points and H_N the basement depth.

A simplified version of Figure 2.1.1 will be used to study the variation of solution accuracy with grid size. The ocean bottom will be assumed fluid with a sound speed of 2000 m/s and Lamé constant of $6.0 \cdot 10^{-5}$ dynes/cm^2. Source depth and horizontal wave number will be held constant at 30 m and 0.062 m^{-1}. Brief comments will be made shortly on the effect on solution accuracy of varying these parameters.

Figures 2.3.1 through 2.3.5 are the response functions obtained by integrating with the values of N_λ ranging from 33.25 down to 2.1, as is noted below each figure. Notice that excellent results are obtained for $N_\lambda = 33.28$ and only slight degradation in accuracy visible for $N_\lambda = 16.6$. More substantial error is noticed for $N_\lambda = 8.4$ and entirely unacceptable solutions obtained with $N_\lambda = 4.2$ and $N_\lambda = 2.1$.

Repeated computation for a variety of bottom parameters and source depths has shown that the value of $N_\lambda = 16$ is the minimum for which accurate solutions can be expected. In a few individual cases, values as low as 12 or 13 produced acceptable results but these cases were not common. Nor did any unique characteristic exist which would enable the a priori knowledge that $N_\lambda = 12$ was acceptable. Similarly, increasing N_λ above 16.0 did not uniformly increase solution accuracy except as a function of horizontal wave number, which will be discussed below. Therefore, the value of $N_\lambda = 16.0$ appears to be most appropriate.

Decreasing the horizontal wave number V, which decreases the spatial wavelength λ_s, does have an influence on solution accuracy. Consider Figures 2.3.6, 2.3.7, and 2.3.8. The ocean model is identical to that used above except that $V = 0.058$ m^{-1}. The values of N_λ are 17, 34, and 51. The error for $N_\lambda = 17$ is greater in Figure 2.3.6 for $V = 0.058$ m^{-1} than in Figure 2.3.2 for $V = 0.062$ m^{-1}. Also,

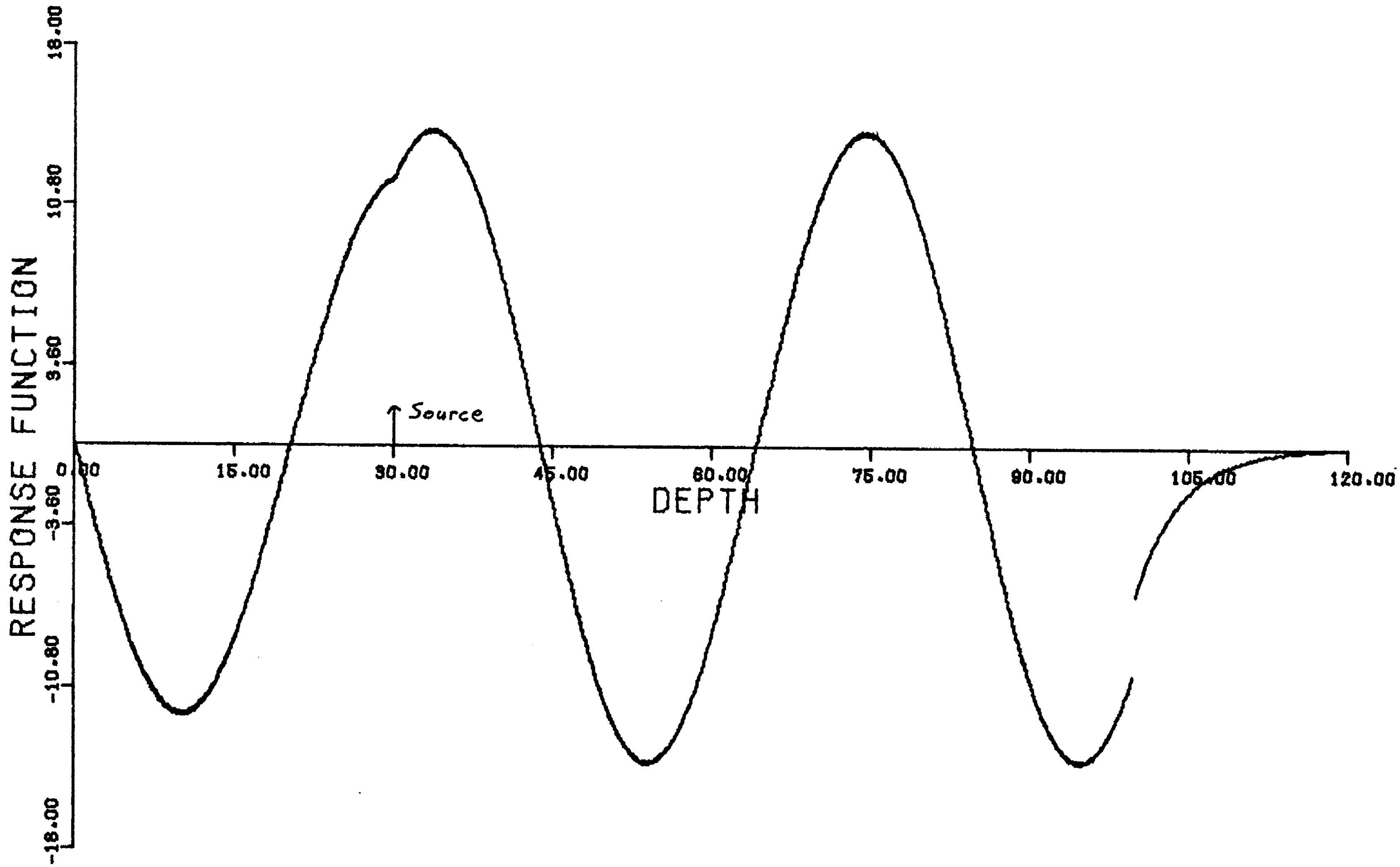


Figure 2.3.1 Response function for two fluid ocean model with $V=0.062m^{-1}$. Basement depth=100m. Source depth=30m. N =33.2.

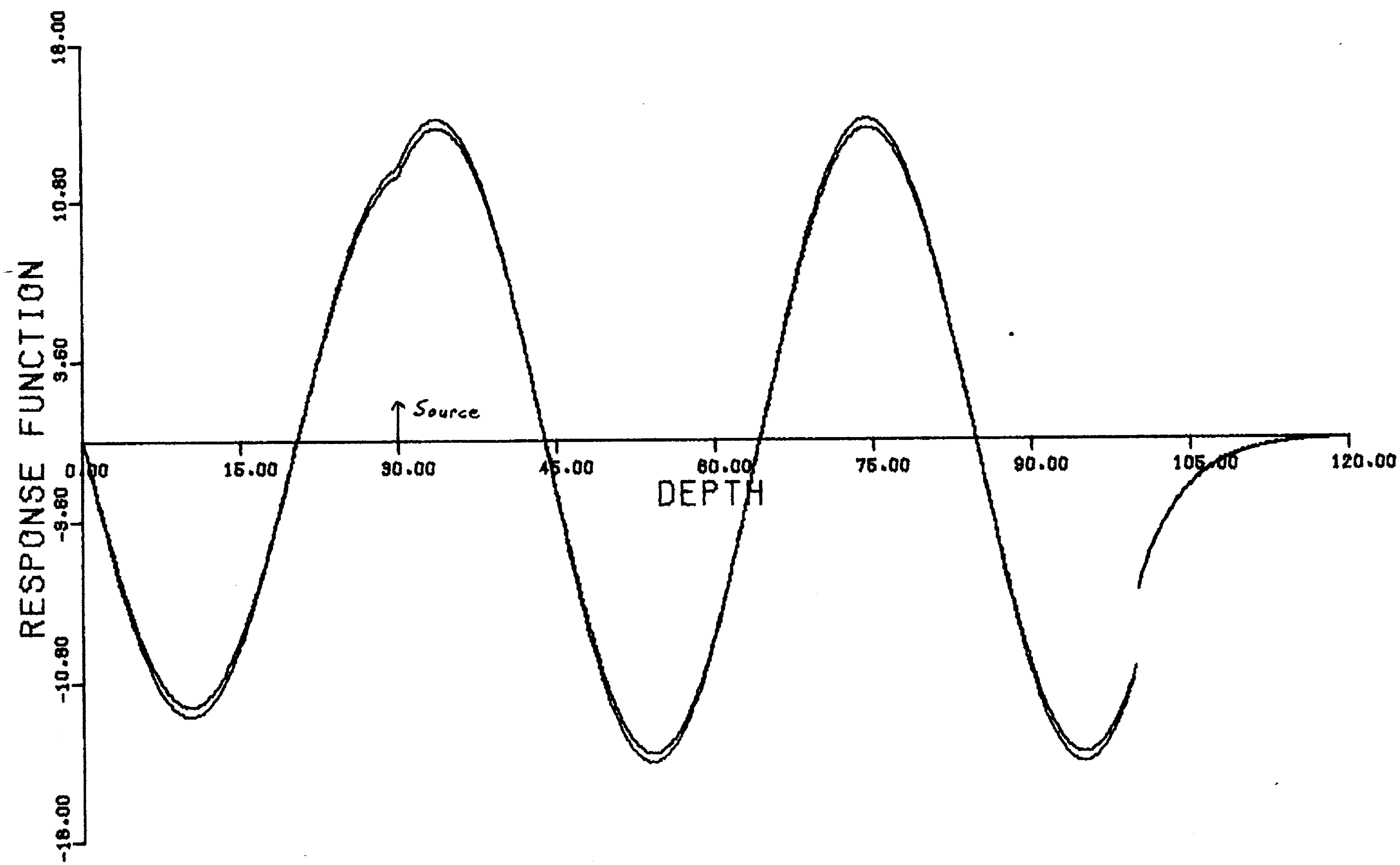


Figure 2.3.2 Response function for two fluid ocean model with V=0.062m^{-1}. Basement depth=100m. Source depth=30m. N =16.6.

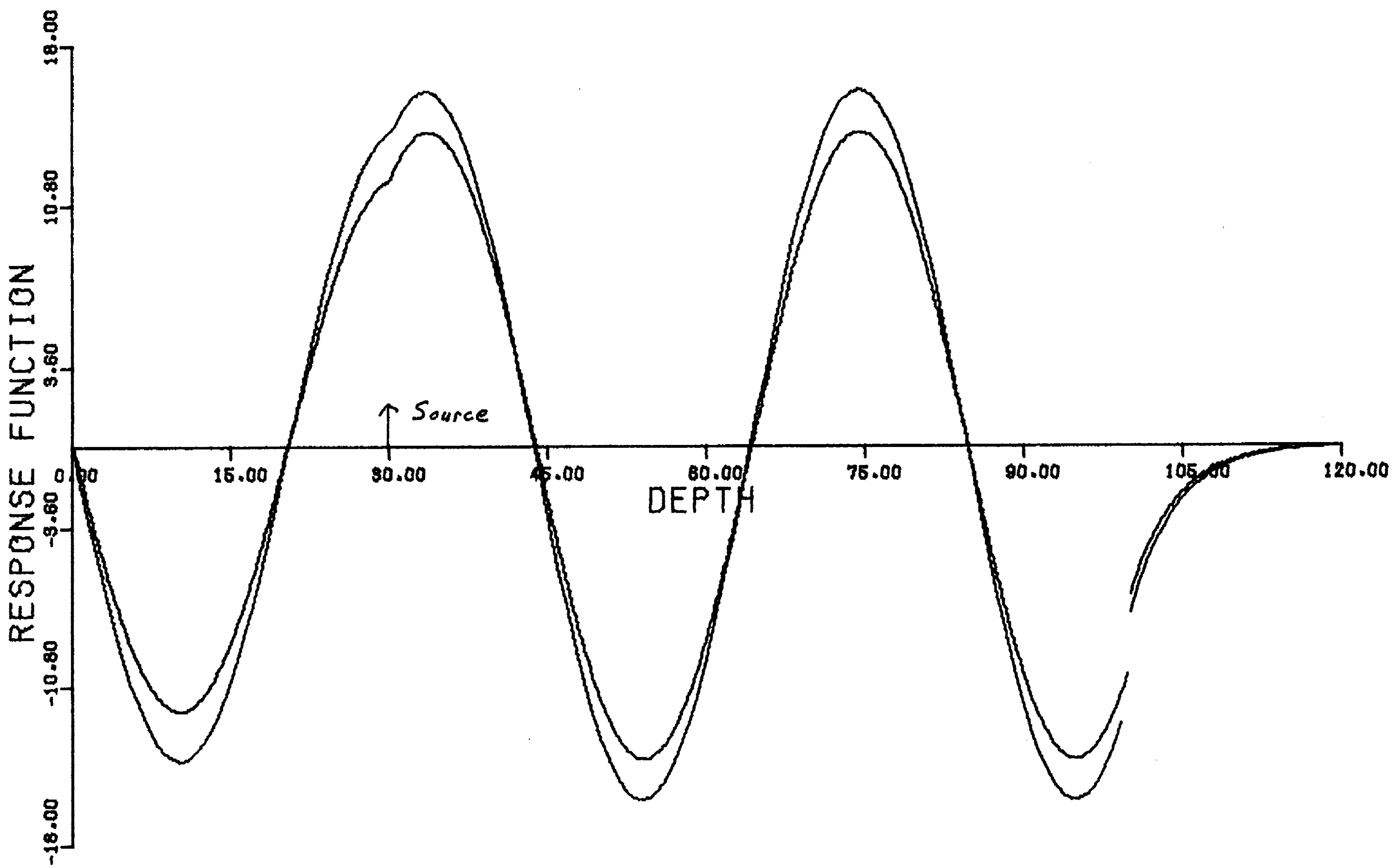


Figure 2.3.3 Response function for two fluid ocean model with $V=0.062m^{-1}$. Basement depth=100m. Source depth=30m. N =8.3.

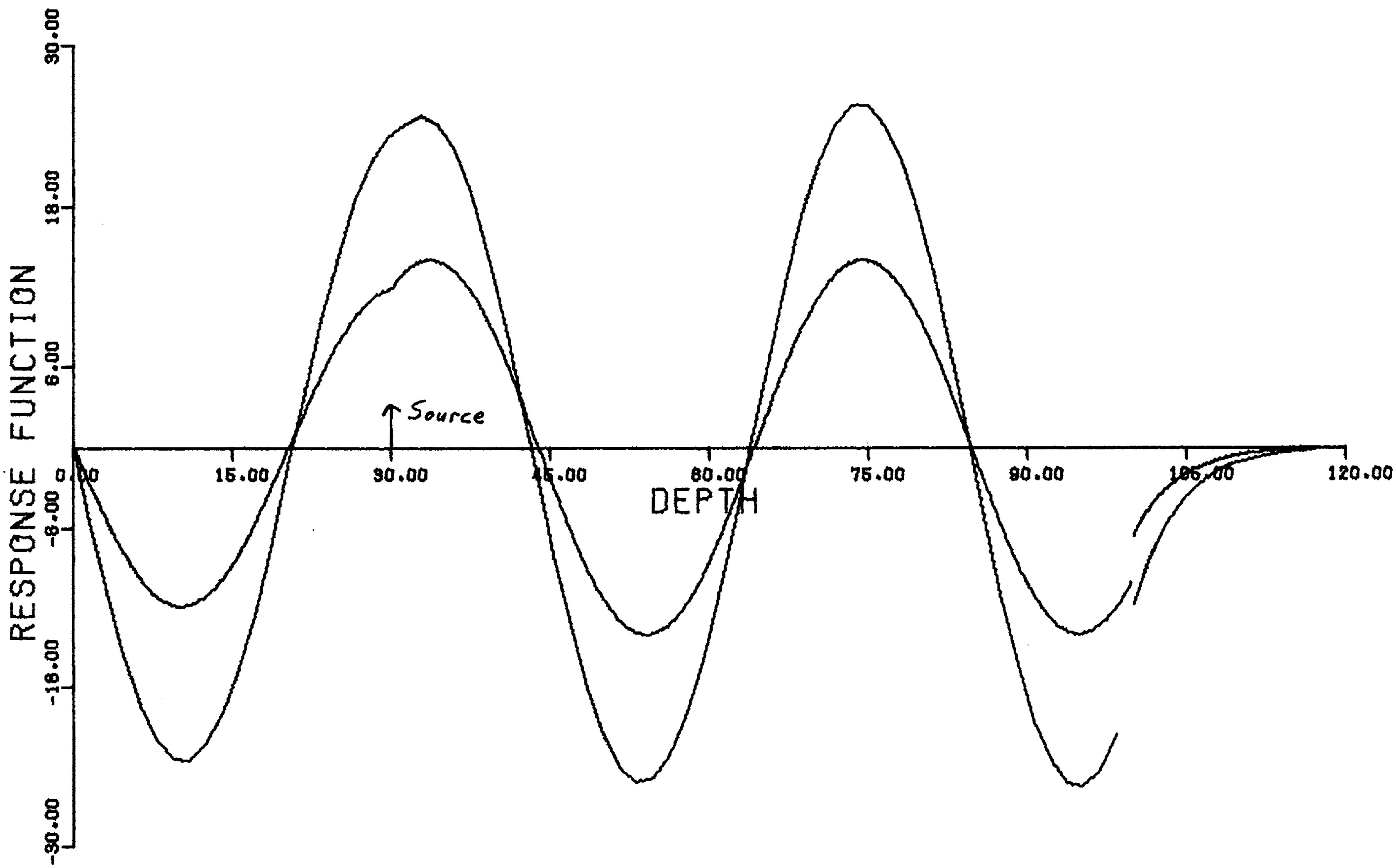


Figure 2.3.4 Response function for two fluid ocean model with $V=0.062m^{-1}$. Basement depth=100m. Source depth=30m. N =4.2.

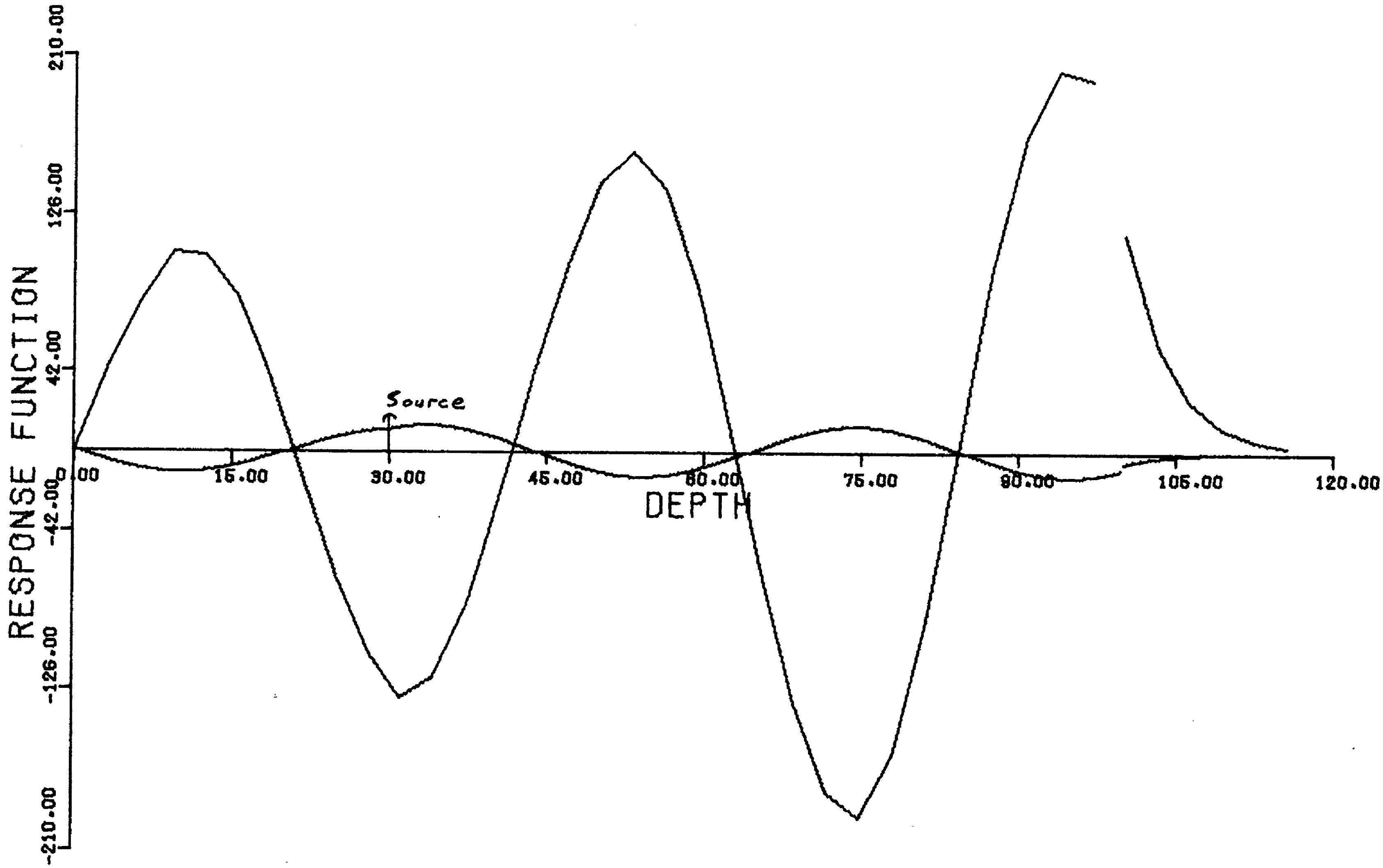


Figure 2.3.5 Response function for two fluid ocean model with $V=0.062m^{-1}$. Basement depth=100m. Source depth=30m. N =2.1.

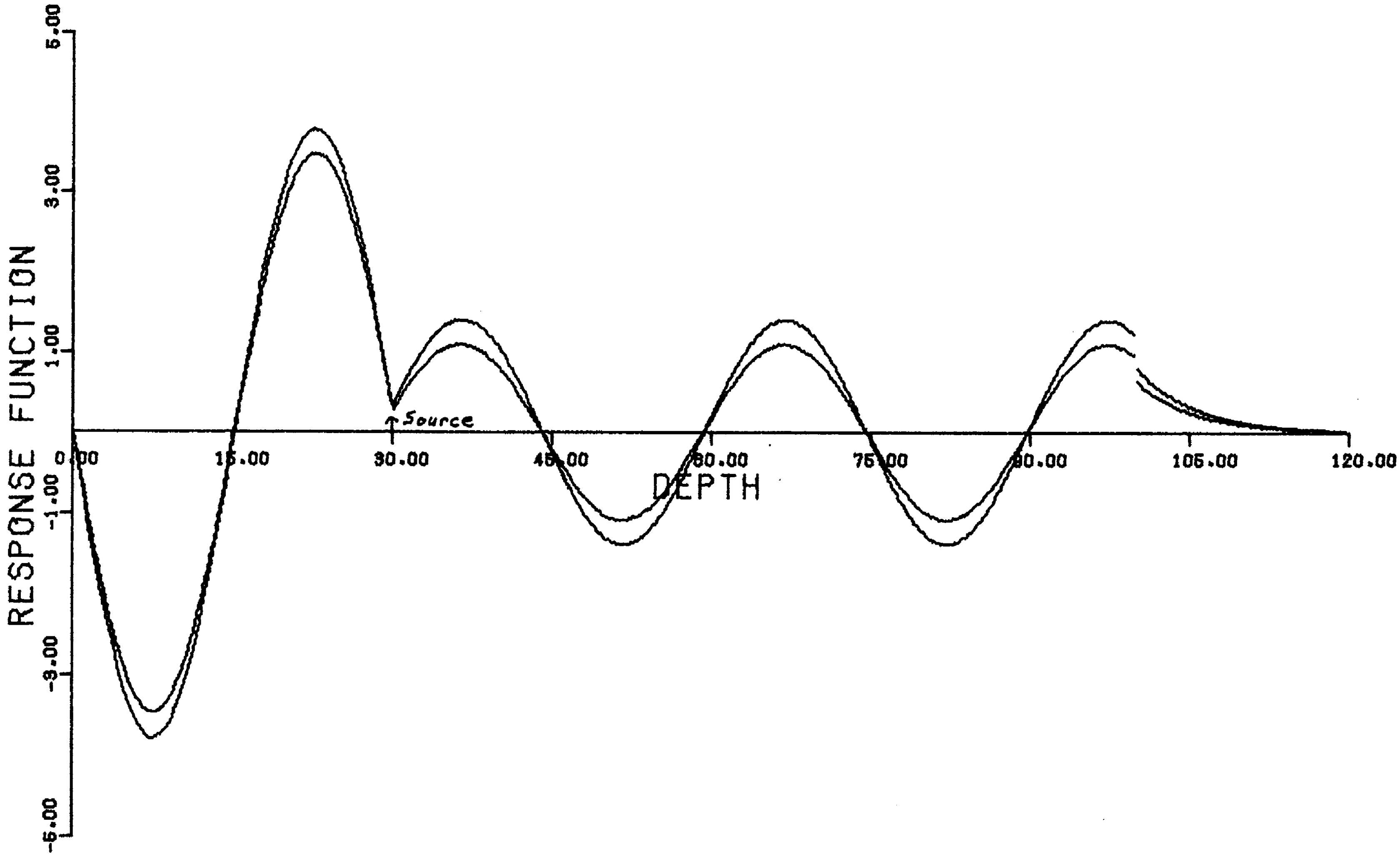


Figure 2.3.6 Response function for two fluid ocean model with $V=0.058m^{-1}$. Basement depth=100m. Source depth=30m. N =17.

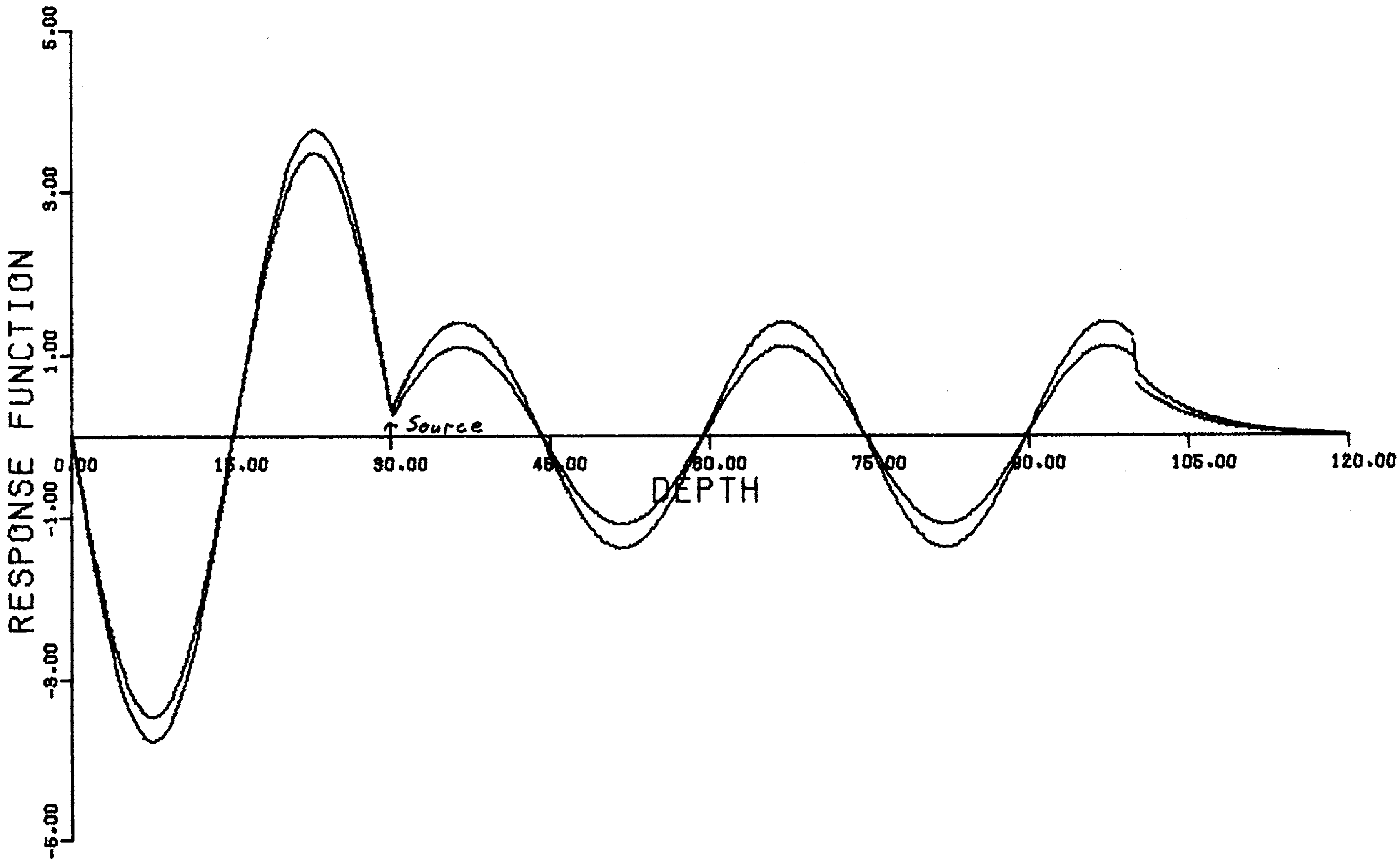


Figure 2.3.7 Response function for two fluid ocean model with V=0.058m^{-1}. Basement depth=100m. Source depth=30m. N =34.

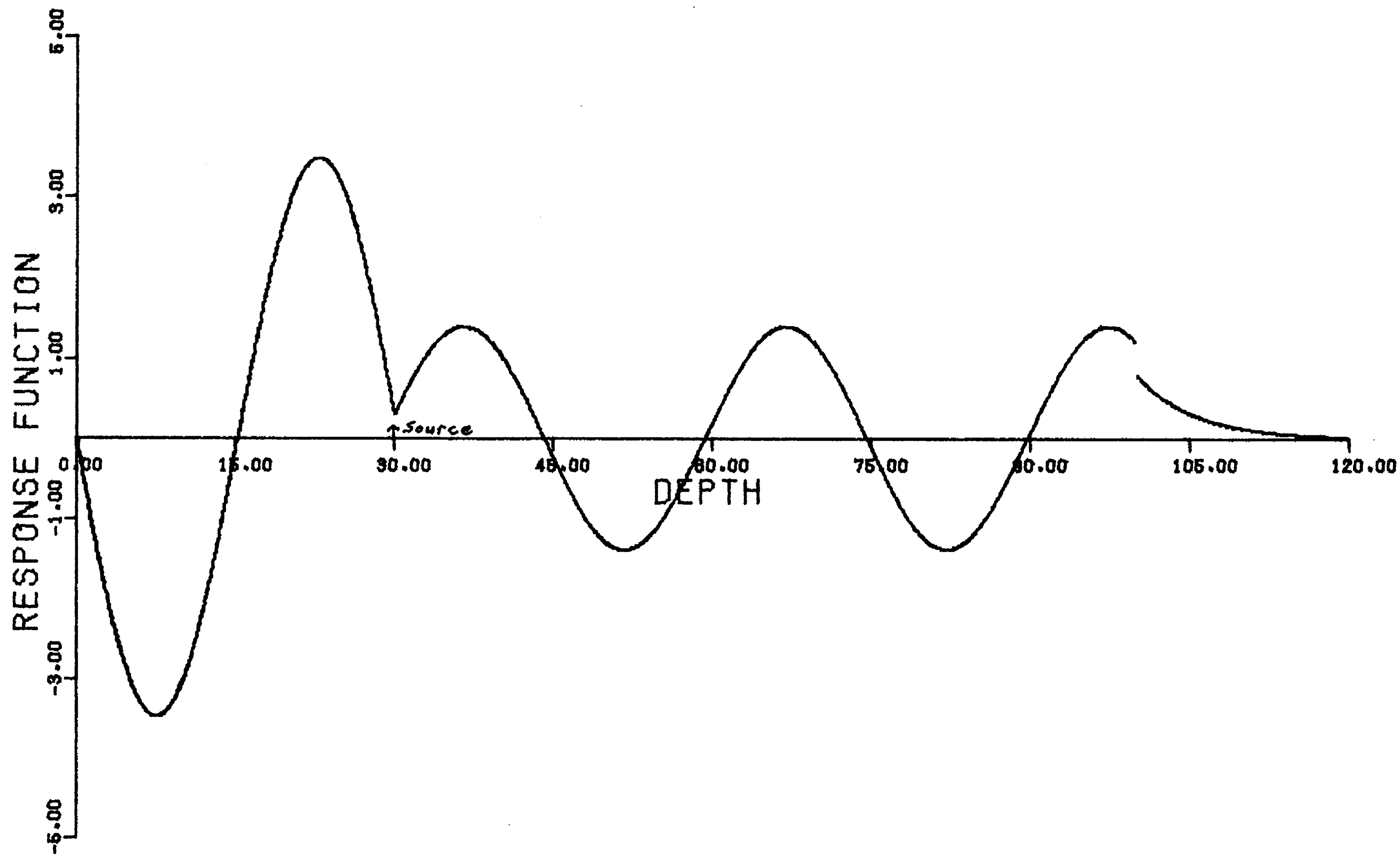


Figure 2.3.8 Response function for two fluid ocean model with $V=0.058m^{-1}$. Basement depth=100m. Source depth=30m. N =51.

there is little difference between 2.3.7 and 2.3.8, as there was between 2.3.1 and 2.3.2 for the equivalent change in N_λ. In fact, convergence to the exact solution does not occur for $V = 0.058\ m^{-1}$ until $N_\lambda = 51$, which is a very high grid density solution (1056 integration points in a 100 meter channel). Continued use of the algorithm has shown that in some situations gradual convergence to the solution occurs as N_λ increases, and in other situations there is a region of variation in N_λ where the error remains constant, as was the case above. No apparent difference exists between these two cases which enables prediction of the type of convergence to expect as V decreases.

To conclude, equation 2.3.2 provides a valid means of selecting the optimal integration grid size. For the wave number domain corresponding to the higher wave numbers (lower order modes), $N_\lambda = 16$ is sufficient. For the lower wave number region (higher order and continuous modes) it appears that increasing error can be expected and N_λ should probably be increased accordingly.

2.4 TWO LAYER OCEAN EXACT SOLUTION AND COMPARISONS

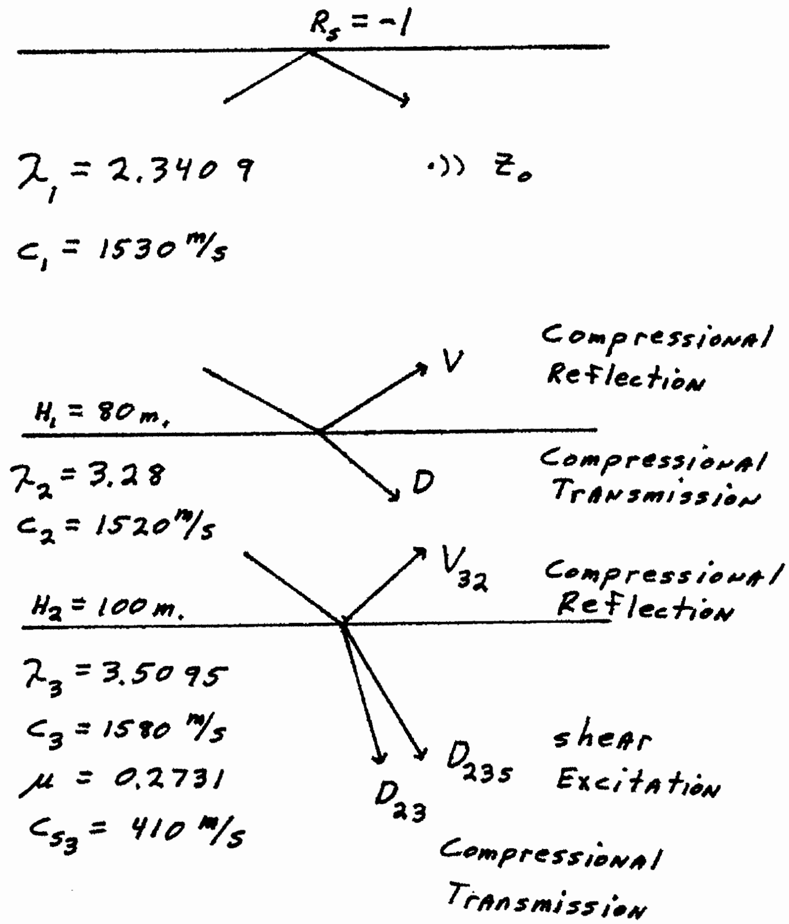
Application of the endpoint method to more complicated models of the ocean is straightforward, providing the required reflection coefficients can be written. As a second test of the state variable technique, the exact solution for the two layered ocean of Figure 2.4.1 will be written. This model, also proposed by Hamilton (28), is common in continental shelf areas where deposition of silts occurred as the sea level rose. For convenience, the second layer will be assumed a fluid. This is realistic, as silts generally have very low shear speeds.

Brekhovskikh's reflection coefficient expressions are again most convenient. The reflection, transmission, and shear conversion coefficients at the lower interface have the same form as Equations 2.1.1, 2.1.2, and 2.1.3. The reflection and transmission coefficients for the upper interface are compound expressions, taking into account the influence of both of the lower media. They are:

$$V = \frac{V_{21} + V_{32}\exp(2ik_{z2}(H_2 - H_1))}{1 + V_{21}V_{32}\exp(2ik_{z2}(H_2 - H_1))} \qquad 2.4.1$$

$$D = \frac{1 + V_{21}}{1 + V_{21}V_{32}\exp(2ik_{z2}(H_2 - H_1))} \qquad 2.4.2$$

$$V_{21} = \frac{Z_2 - Z_1}{Z_2 + Z_1} \qquad 2.4.3$$



Two Layer Ocean Description

Figure 2.4.1 λ and μ have units dynes/cm$^2 \cdot 10^{-5}$

The expressions for each term in the exact solution are

$$G_a(z,z_0) = W^{-1} u_a(z) u_b(z_0) \qquad 0 \le z \le z_0 \tag{2.4.4}$$

$$G_b(z,z_0) = W^{-1} u_a(z_0) u_b(z) \qquad z_0 \le z \le H_1 \tag{2.4.5}$$

$$G_2(z,z_0) = W^{-1} u_a(z_0) u_2(z) \qquad H_1 \le z \le H_2 \tag{2.4.6}$$

$$G_3(z,z_0) = W^{-1} u_a(z_0) u_3(z) \qquad H_2 \le z \tag{2.4.7}$$

$$G_{3s}(z,z_0) = W^{-1} u_a(z_0) u_{3s}(z) \qquad H_2 \le z \tag{2.4.8}$$

where W is the Wronskian defined in 2.1.7.

The linearly independent solutions to the homogeneous equation are

$$u_a(z) = e^{-ik_{z1}z} - e^{ik_{z1}z} \qquad 0 \le z \le z_0 \tag{2.4.9}$$

$$u_b(z) = e^{ik_{z1}z} + V e^{2ik_{z1}H_1} e^{-ik_{z1}z} \qquad z_0 \le z \le H_1 \tag{2.4.10}$$

$$U_2(z) = D e^{-ik_{z2}H_1}\left[e^{ik_{z2}z} + V_{32} e^{2ik_{z2}H_2} e^{-ik_{z2}z}\right] \qquad 2.4.11$$

$$H_1 \le z \le H_2$$

$$U_3(z) = D D_{23} e^{ik_{z3}z} \qquad H_2 \le z \qquad 2.4.12$$

$$U_{3s}(z) = D D_{23} e^{ik_{z3s}z} \qquad H_2 \le z \qquad 2.4.13$$

The resulting solution is

$$G_a(z, z_0) = -2i \sin(k_{z1} z)\left[e^{ik_{z1}z_0} + V e^{2ik_{z1}H} e^{-ik_{z1}z_0}\right] W^{-1} \qquad 2.4.14$$

$$G_b(z, z_0) = -2i \sin(k_{z1} z_0)\left[e^{ik_{z1}z} + V e^{2ik_{z1}H} e^{-ik_{z1}z}\right] W^{-1} \qquad 2.4.15$$

$$G_2(z, z_0) = -2i \sin(k_{z1} z_0)\left[e^{ik_{z2}z} + V_{32} e^{2ik_{z2}H_2} e^{-ik_{z2}z}\right] \cdot$$

$$\cdot D W^{-1} e^{iH_1(k_{z1} - k_{z2})} \qquad 2.4.16$$

$$G_3(z, z_0) = -2i \sin(k_{z1} z_0) D D_{23} W^{-1} e^{ik_{z1}H_1} \cdot \qquad 2.4.17$$

$$e^{ik_{z2}(H_2 - H_1)} e^{ik_{z3}(z - H_2)}$$

$$G_{35}(z, z_0) = -2i \sin(k_{z1} z_0)\, D D_{23} W^{-1} e^{i k_{z1} H_1} \cdot e^{i k_{z2}(H_2 - H_1)} e^{i k_{z35}(z - H_2)} \qquad 2.4.18$$

$$W^{-1} = 2 k_{z1} \Big[\sin(k_{z1} z_0)\left(e^{i k_{z1} z_0} - V e^{2 i k_{z1} H} e^{-i k_{z1} z_0}\right) + i \cos(k_{z1} z_0)\left(e^{i k_{z1} z_0} + V e^{2 i k_{z1} H} e^{-i k_{z1} z}\right) \Big] \qquad 2.4.19$$

Figures 2.4.2 through 2.4.5 are the solutions obtained from Equations 2.4.14 through 2.4.19 and from the state variable algorithm. Integration step size was chosen to satisfy 2.3.2 for $N_\lambda \geq 16$. The respective horizontal wavenumbers are $V = 0.065$, 0.064, 0.061, and $0.058\ m^{-1}$. As in the single layered ocean results, these figures show that accurate results are again obtained by the state variable algorithm. Further study of this two layered ocean proves that, as in the single layer case, solution accuracy is not a function of any oceanic parameters other than vertical wavenumber. The same characteristics of the solution were found for the two layer ocean as were found for the single layer ocean with regard to the value of N_λ in 2.3.2.

The state variable technique can be used to obtain accurate solutions of the depth-separated wave equation. The solutions are stable with regard to all oceanic parameters. Required grid density has been specified by 2.3.2. In Chapter 3 the technique will be used to study the effects of including complex sound speeds to model attenuation.

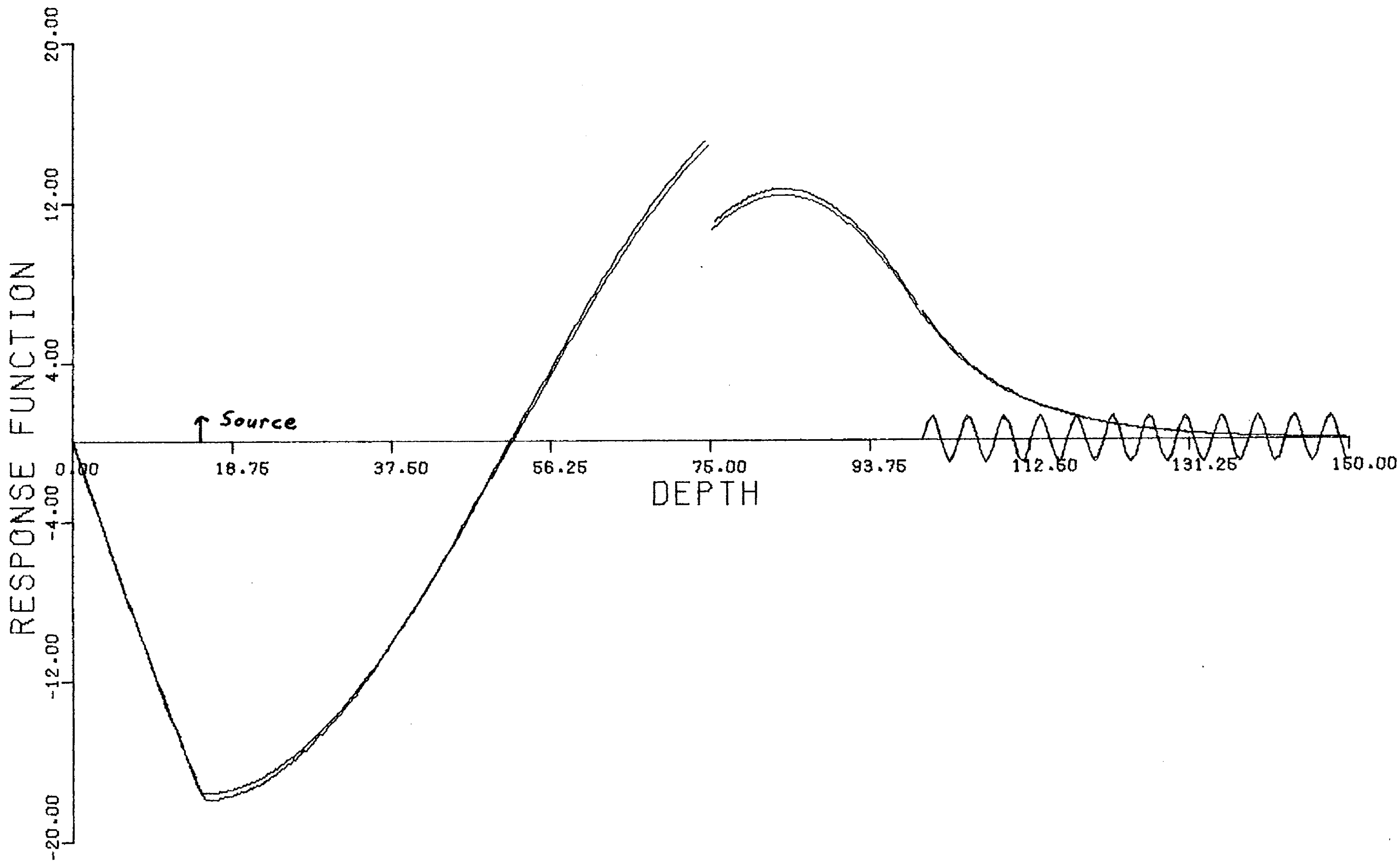


Figure 2.4.2 Response function for two layer ocean model with $V=0.065m^{-1}$. Shear field plotted five times actual magnitude. Basement depth=100m. Layer depth=75m. Source depth=15m.

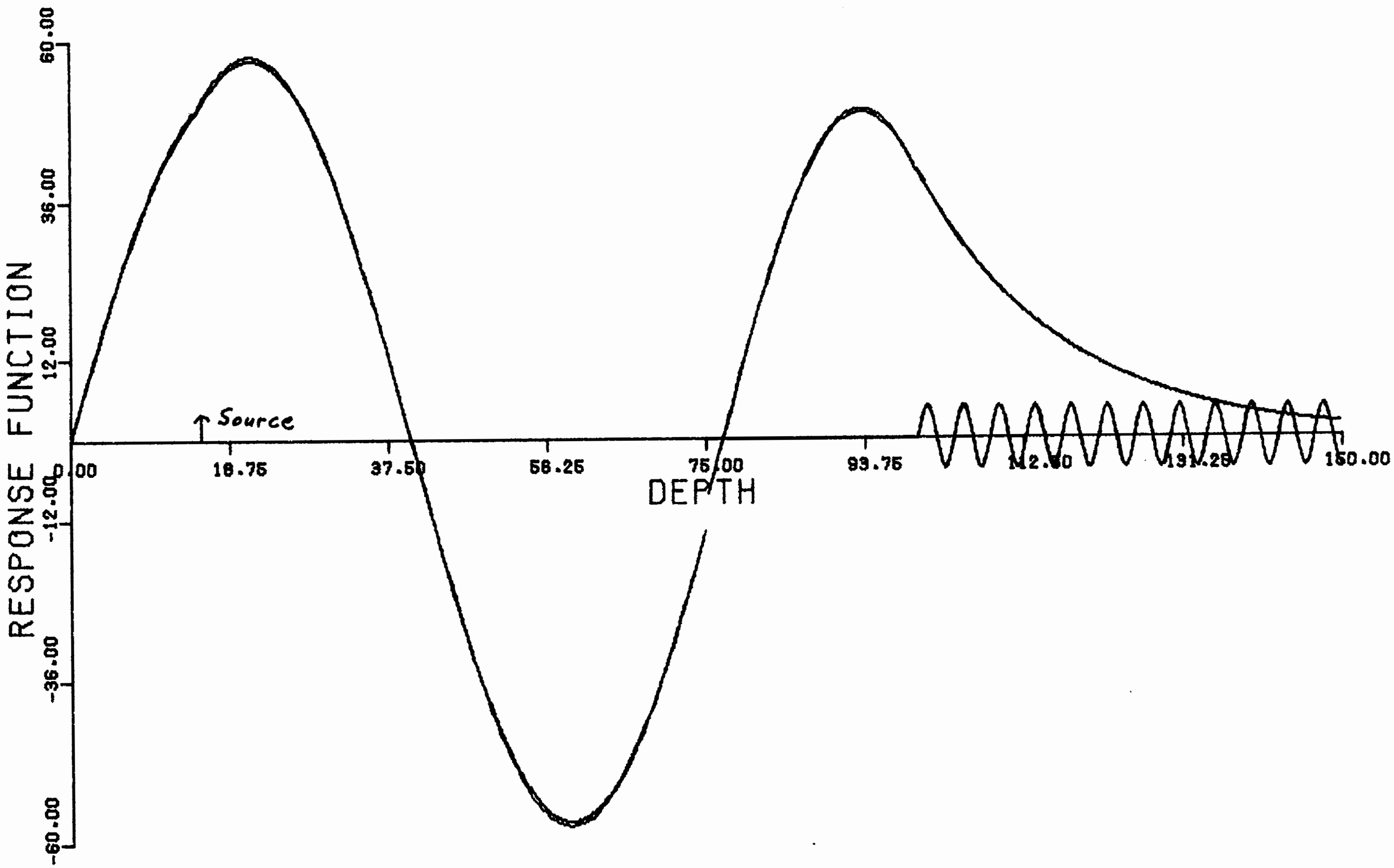


Figure 2.4.3 Response function for two layer ocean model with $V=0.064m^{-1}$. Shear field plotted five times actual magnitude. Basement depth=100m. Layer depth=75m. Source depth=15m.

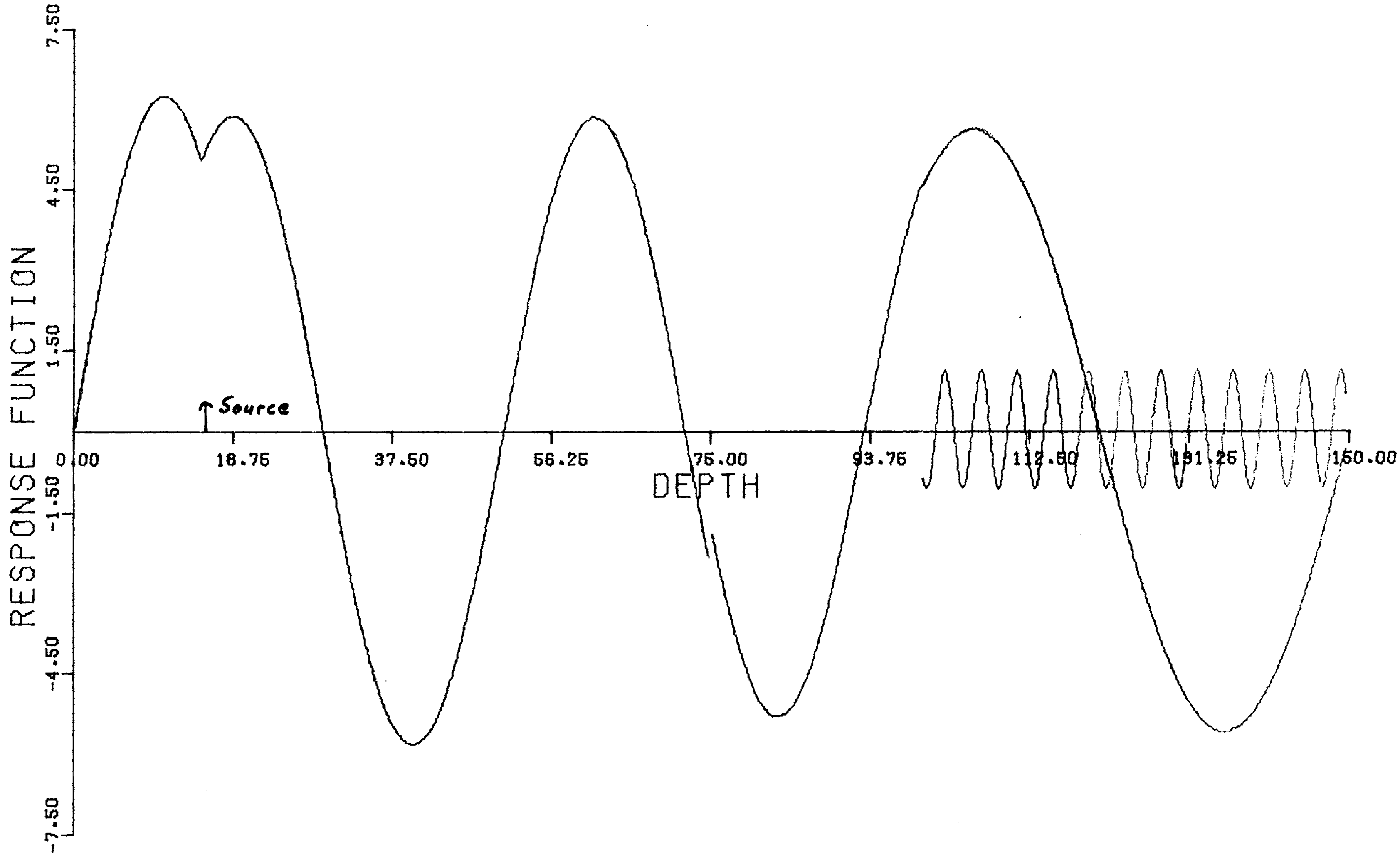


Figure 2.4.4 Response function for two layer ocean model with $V=0.061m^{-1}$. Shear field plotted five times actual magnitude. Basement depth=100m. Layer depth=75m. Source depth=15m.

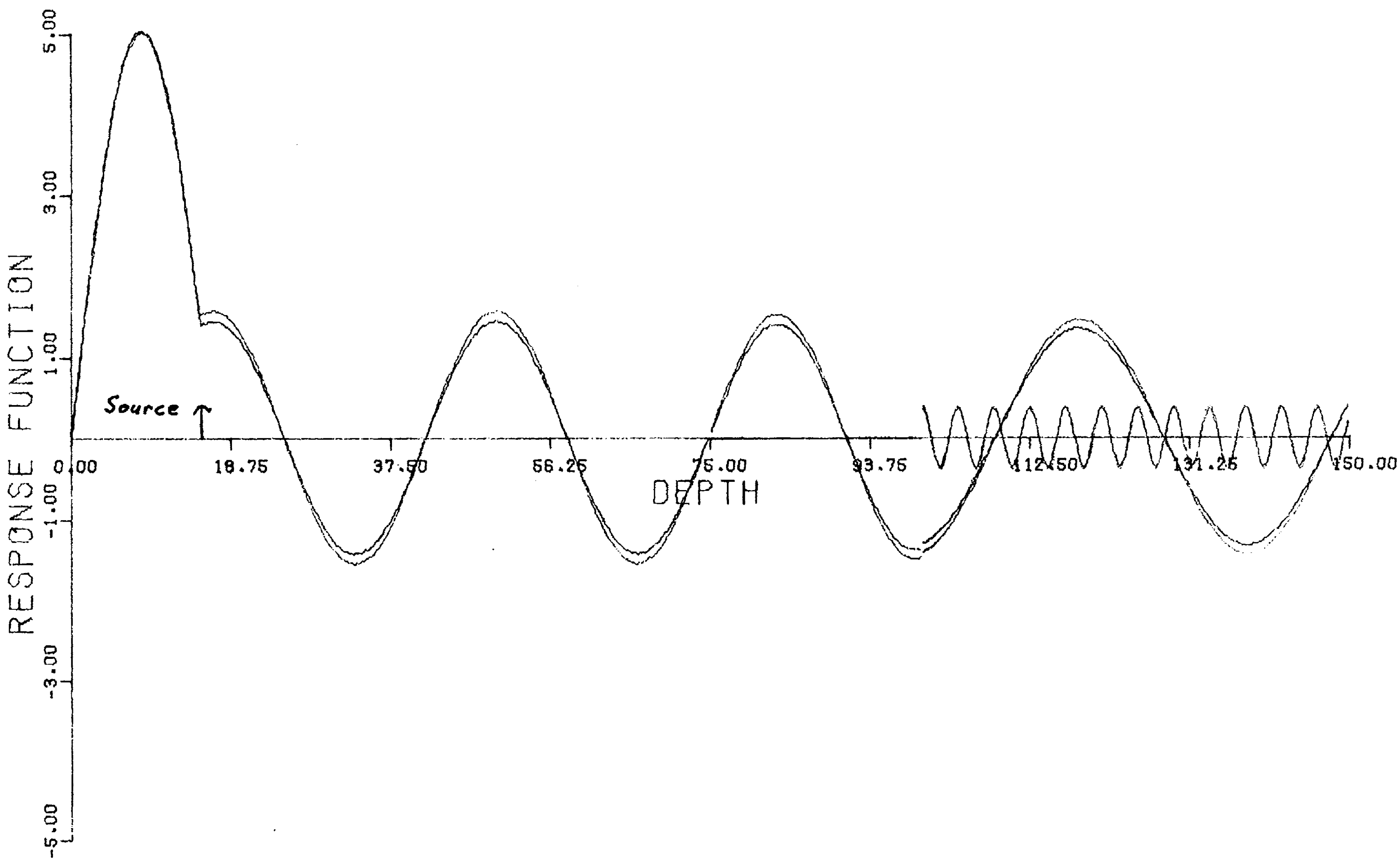


Figure 2.4.5 Response function for two layer ocean model with $V=0.058m^{-1}$. Shear field plotted five times actual magnitude. Basement depth=100m. Layer depth=75m. Source depth=15m.

3. ATTENUATION AND DEEP OCEAN MODEL

Chapters 1 and 2 introduced the state variable technique for obtaining solutions to the inhomogeneous depth separated wave equation. By incorporating complex sound speeds, the technique can also be used to study the influence of bottom attenuation on the solution. Section 3.1 will consider the single layer ocean of Chapter 2 with bottom attenuation. Section 3.2 will discuss an instability that arises during the integration process when complex sound speeds are included and the influence of that instability on the resulting response function's accuracy. Section 3.3 will conclude with a general discussion of the use of the algorithm on a deep ocean model. The complex sound speeds will, in general, cause the response functions to have real and imaginary components. Consequently, both the magnitude and phase of the complex response function will be discussed. The phase state variable will also be presented with and without attenuation.

3.1 SINGLE LAYER OCEAN BOTTOM ATTENUATION

Shear and compressional wave attenuation in marine sediments has been reviewed by Hamilton (34, 35). Experimental data indicates that compressional attenuation coefficients have a dependence on the first power of frequency, i.e., $a_c = kf^1$ (34). The coefficient k is largest for sands, smallest for silty-sands, and intermediate for silt-clay muds. Very little data is available on shear wave attenuation. Relating the shear attenuation coefficient a_s to the compressional attenuation coefficient a_c is most conveniently done by defining the logarithmic as

$$\Delta_s = \frac{a_s c_s}{f} \qquad\qquad \Delta_c = \frac{a_c c_c}{f}$$

where c_s and c_c are bottom shear and compressional wave speeds and f the frequency. The ratio of Δ_c to Δ_s has been found to be 0.3 for sands and 0.1 for silt-clays (35). Until more thorough experimental data is available, the best method of obtaining a_s is to determine a_c from reference 34 and use

$$\frac{\Delta_c}{\Delta_s} = \frac{a_c c_c}{a_s c_s} = \begin{cases} 0.3 & \text{sand} \\ 0.1 & \text{silt-clay} \end{cases} \qquad 3.1.1$$

to compute the best estimate of shear attenuation.

When attenuation mechanisms are present, the horizontal wavenumber becomes complex, i.e., $V + i\delta$. Horizontally propagating plane waves then consist of a product of two exponentials.

$$e^{i2\pi(V+i\delta)X} = e^{i2\pi VX} e^{-2\pi\delta X} \qquad 3.1.2$$

The second exponential provides the decay with distance due to attenuation.

When bottom attenuation is the only loss mechanism to be considered, a slightly more convenient method of including attenuation uses complex sound speeds in the propagation constant $K_c = 2\pi f/c_c$ for the bottom. Recall from Section 2.1,

$$2\pi V = \frac{2\pi f}{c_c} \sin\theta_1$$

Writing the bottom sound speed as $c_c - ic_c'$, K_c becomes

$$K_c = \frac{2\pi f}{c_c - ic_c'} = \frac{2\pi f c_c}{c_c^2 + c_c'^2} + i\frac{2\pi f c_c'}{c_c^2 + c_c'^2}$$

Similarly for $c_s - ic_s'$

$$K_s = \frac{2\pi f}{c_s - ic_s'} = \frac{2\pi f c_s}{c_s^2 + c_s'^2} + i\frac{2\pi f c_s'}{c_s^2 + c_s'^2}$$

Therefore,

$$a_c = \frac{2\pi f c_c'}{c_c^2 + c_c'^2}$$

$$a_s = \frac{2\pi f c_s'}{c_s^2 + c_s'^2} \qquad 3.1.3$$

Notice that the imaginary sound speed term has a negative sign. This insures that Equation 3.1.2 does in fact have a decaying exponential.

Representative values of the attenuation coefficients can be obtained from the literature (34, 35). At 100 Hz, for example, experimental data gives the following limits on a_c and a_s

$$7 \cdot 10^{-4} < a_c < 1 \cdot 10^{-2}$$

$$4 \cdot 10^{-2} < a_s < 2 \cdot 10^{-1}$$

For sand, a typical value from reference 35 is $a_c = 0.0056$. Using 3.1.1 and 3.1.3, the sound speeds that result are

$$1675 - i25$$

$$450 - i22.4$$

Figures 3.1.1 through 3.1.7 display various facets of the solution. On each of the first five, the magnitude of the response function evaluated both with and without bottom attenuation is plotted. Figures 3.1.1, 3.1.2, and 3.1.3 correspond to the first, third, and fifth modes of the comparable Pekeris waveguide. For the Pekeris case without bottom attenuation, infinite response would result at modal wavenumbers. In this case, the shear acts as a loss mechanism and the response function is finite. The fourth and fifth are for wavenumbers on either side of the critical angle, which occurs at V = 0.0597. Figure 3.1.6 plots the phase of the complex potential for the fifth mode. Figure 3.1.7 is an example of the variation of the state variable $\theta(\xi)$ with depth for V = 0.0651205.

The first three figures exhibit characteristics that are predictable. The magnitudes for the attenuated cases are all smoothed compared to the unattenuated cases; that is, the maxima are smaller and the minima larger. The propagating shear wave, formerly of constant magnitude, now exhibits the expected exponential damping. The inhomogeneous compressional wave in the bottom shows a slight decrease in amplitude from the unattenuated case. Finally, increased coupling to the bottom as wave number decreases is apparent via the increasing proportion of shear wave generation.

Decreasing the horizontal wave number further to V = 0.061 and V = 0.059 produces an unexpected effect on the response function. On the first three figures, all lobes of the magnitude were uniformly attenuated. This characteristic is not evident as V decreases. In particular in 3.1.4, the upper two lobes are only very slightly smaller, while the lower four are more noticeably attenuated, as was the case

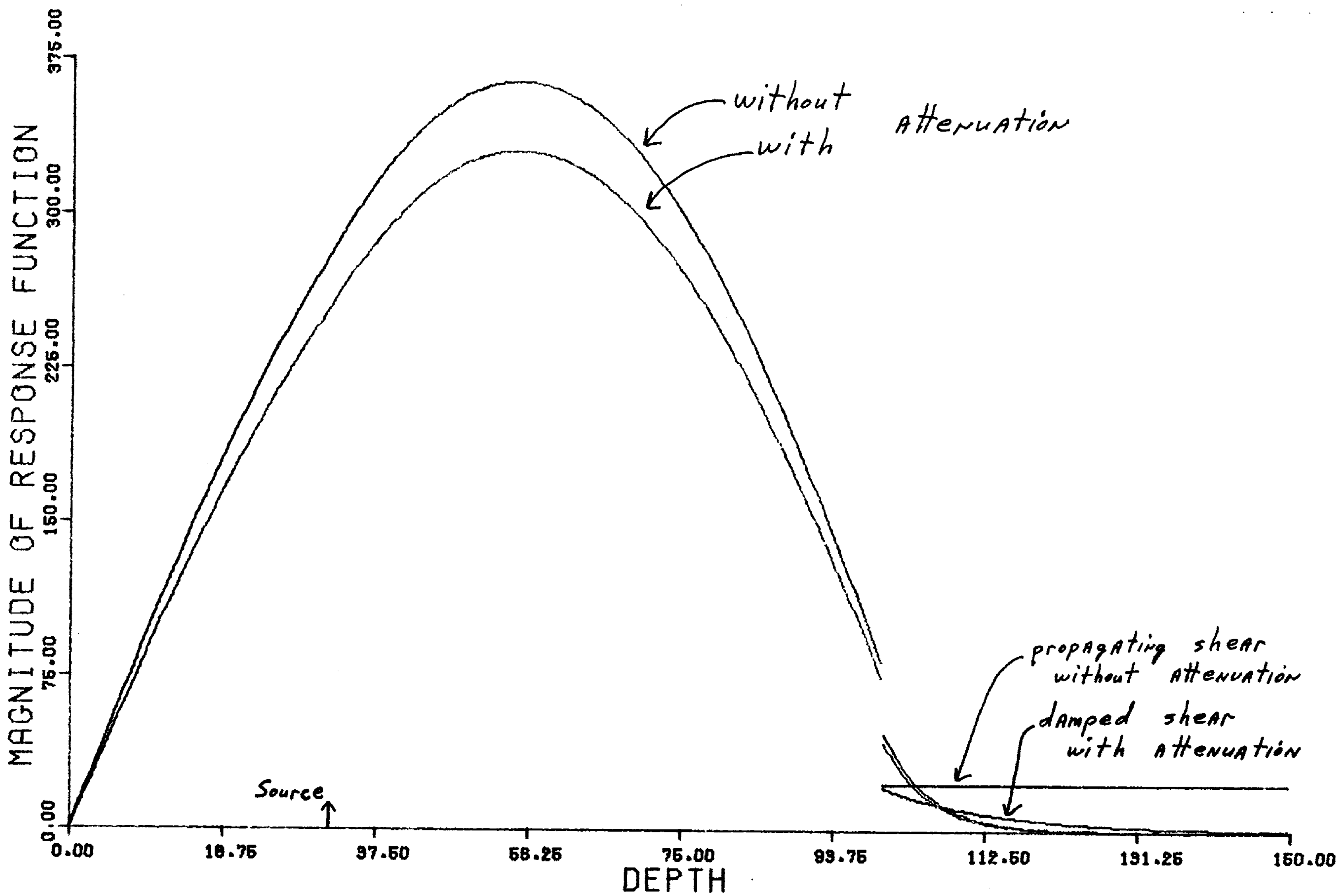


Figure 3.1.1 Magnitude of response function for two layer ocean with $V=0.0664959m^{-1}$. Basement depth=100m. Source depth=30m.

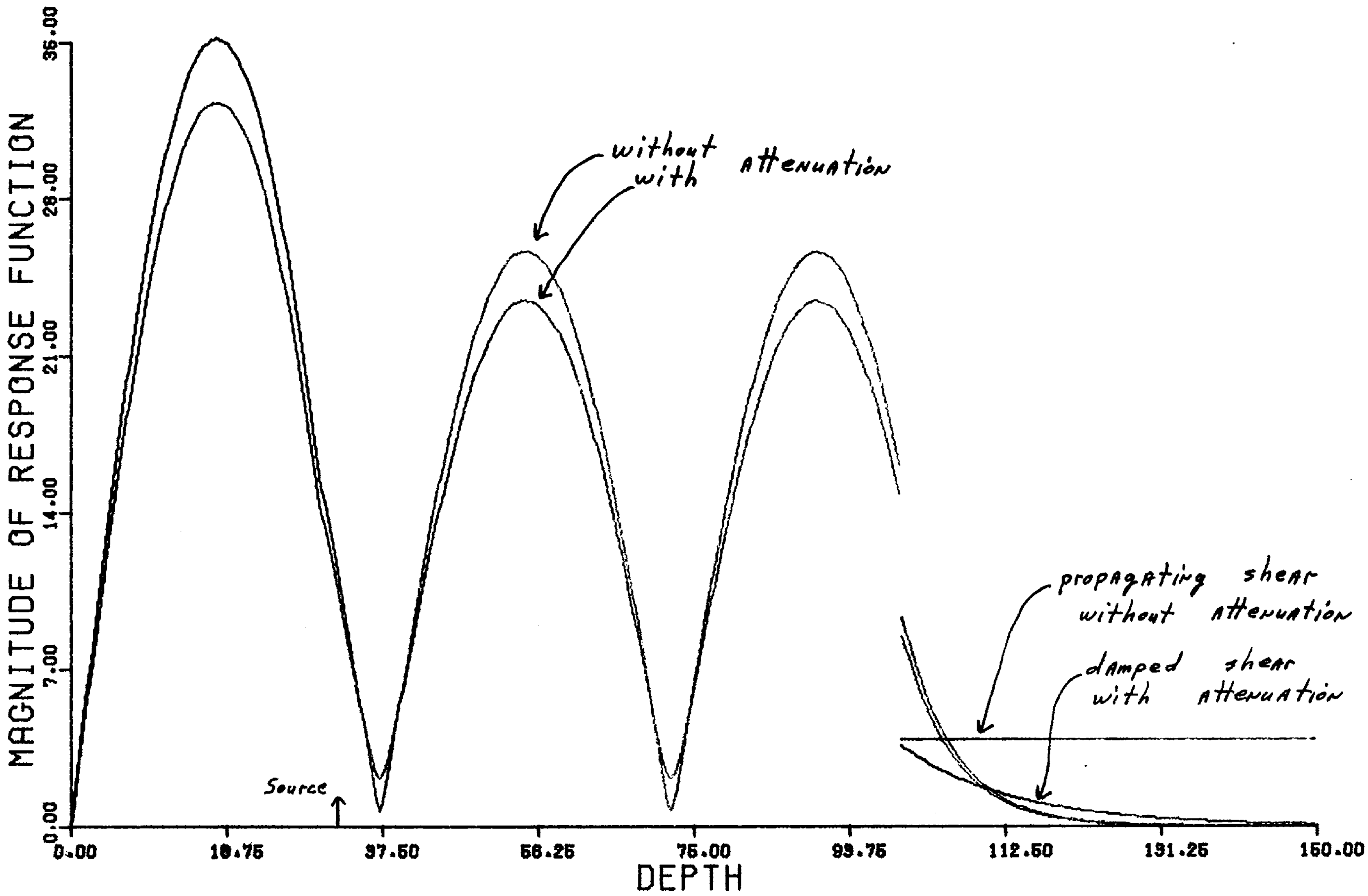


Figure 3.1.2 Magnitude of response function for two layer ocean with $V=0.0651205 m^{-1}$. Basement depth=100m. Source depth=30m.

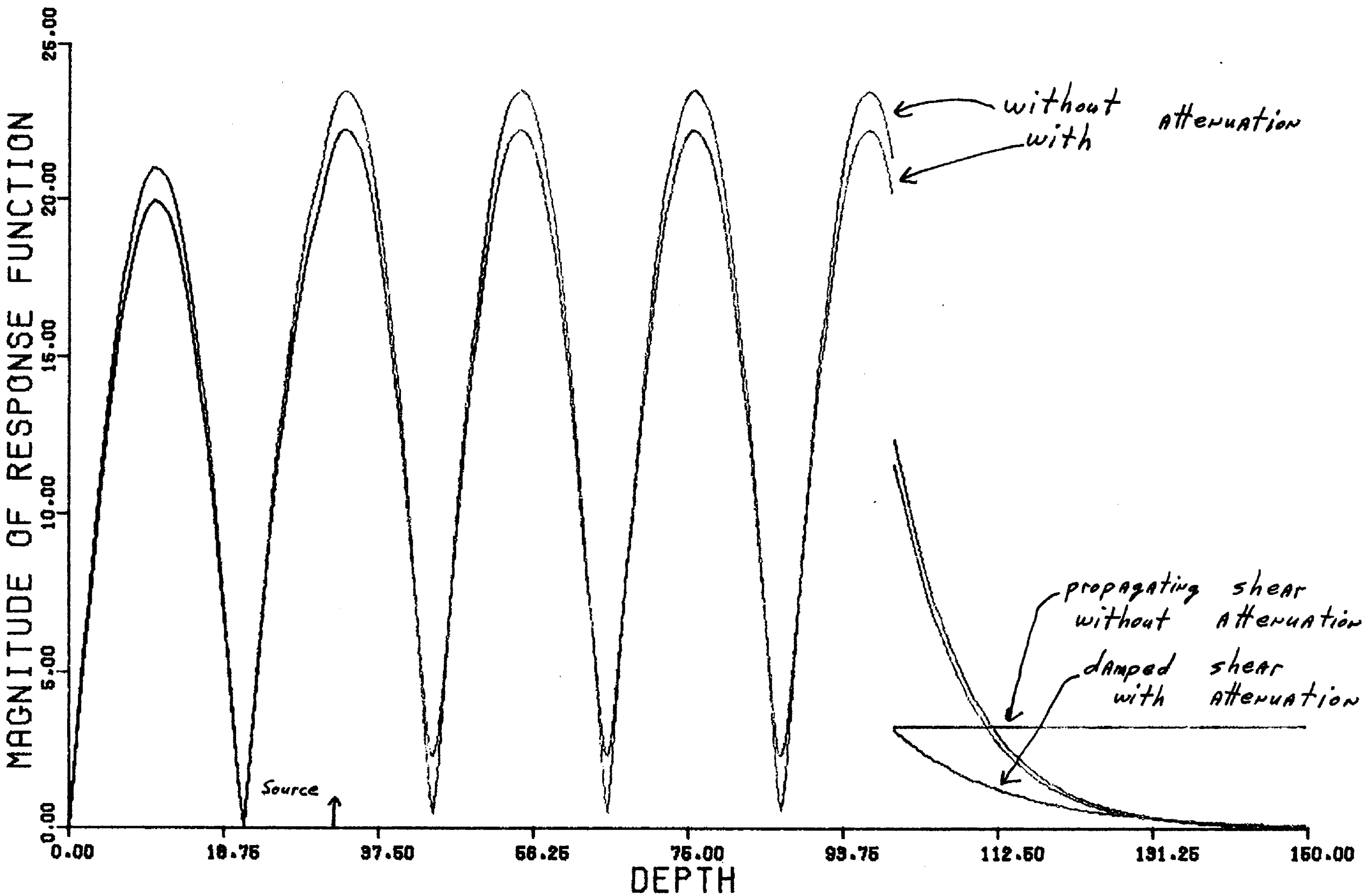


Figure 3.1.3 Magnitude of response function for two layer ocean with $V=0.0622979m^{-1}$. Basement depth=100m. Source depth=30m.

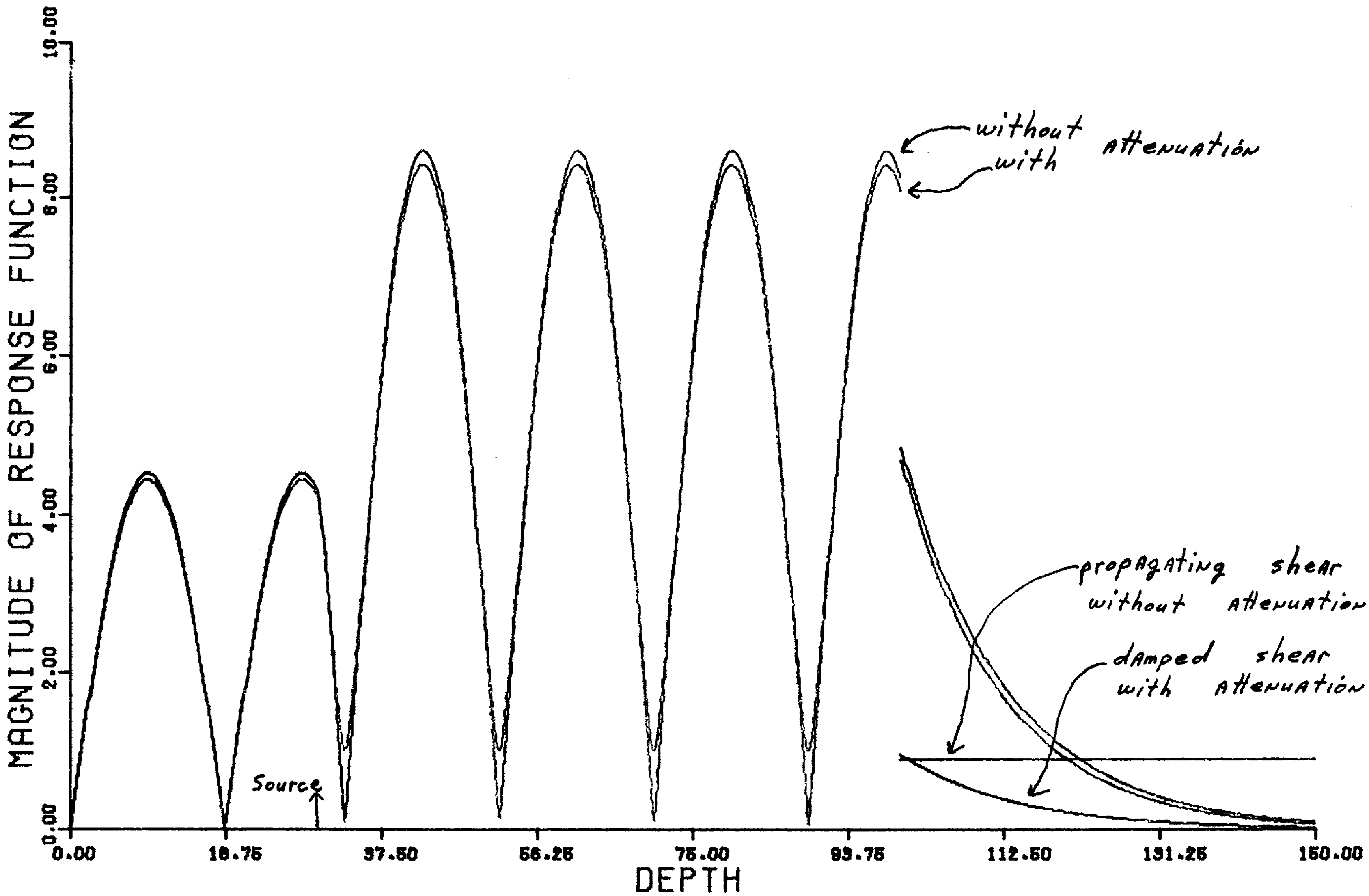


Figure 3.1.4 Magnitude of response function for two layer ocean with $V=0.061m^{-1}$. Basement depth=100m. Source depth=30m.

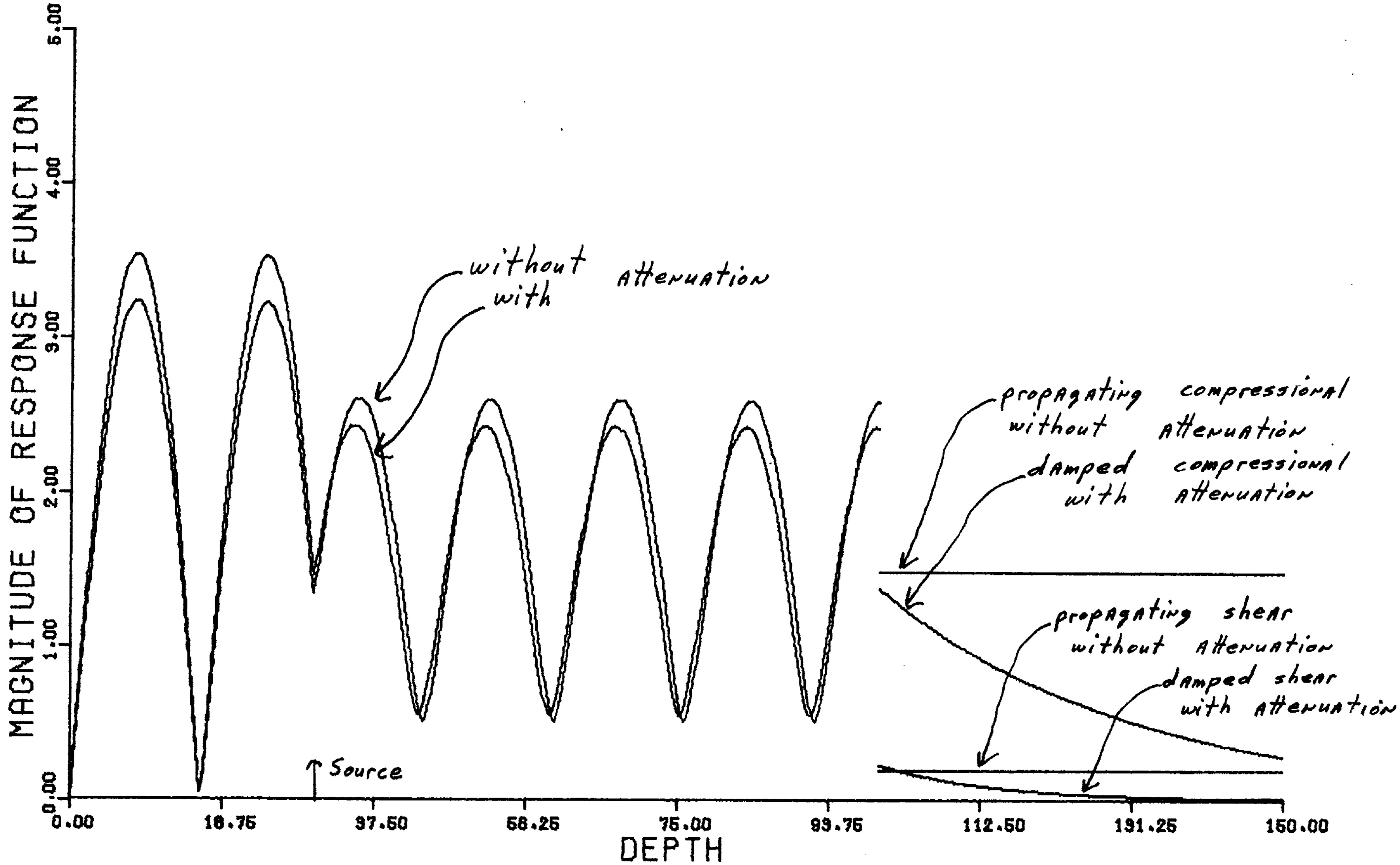


Figure 3.1.5 Magnitude of response function for two layer ocean with $V=0.059m^{-1}$. Basement depth=100m. Source depth=30m.

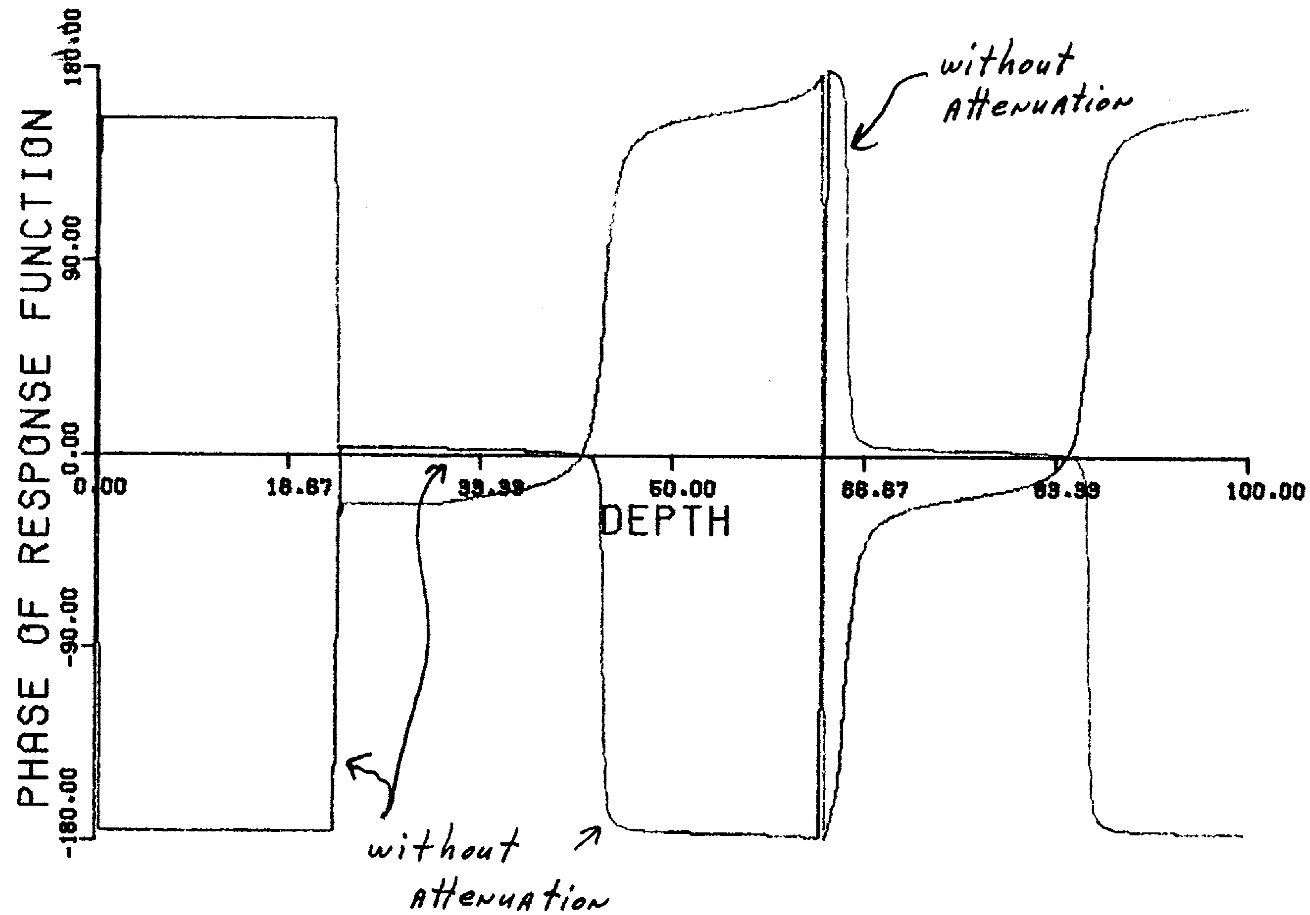


Figure 3.1.6 Phase of response function for two layer ocean with $V=0.0622979m^{-1}$. Basement depth=100m. Source depth=30m.

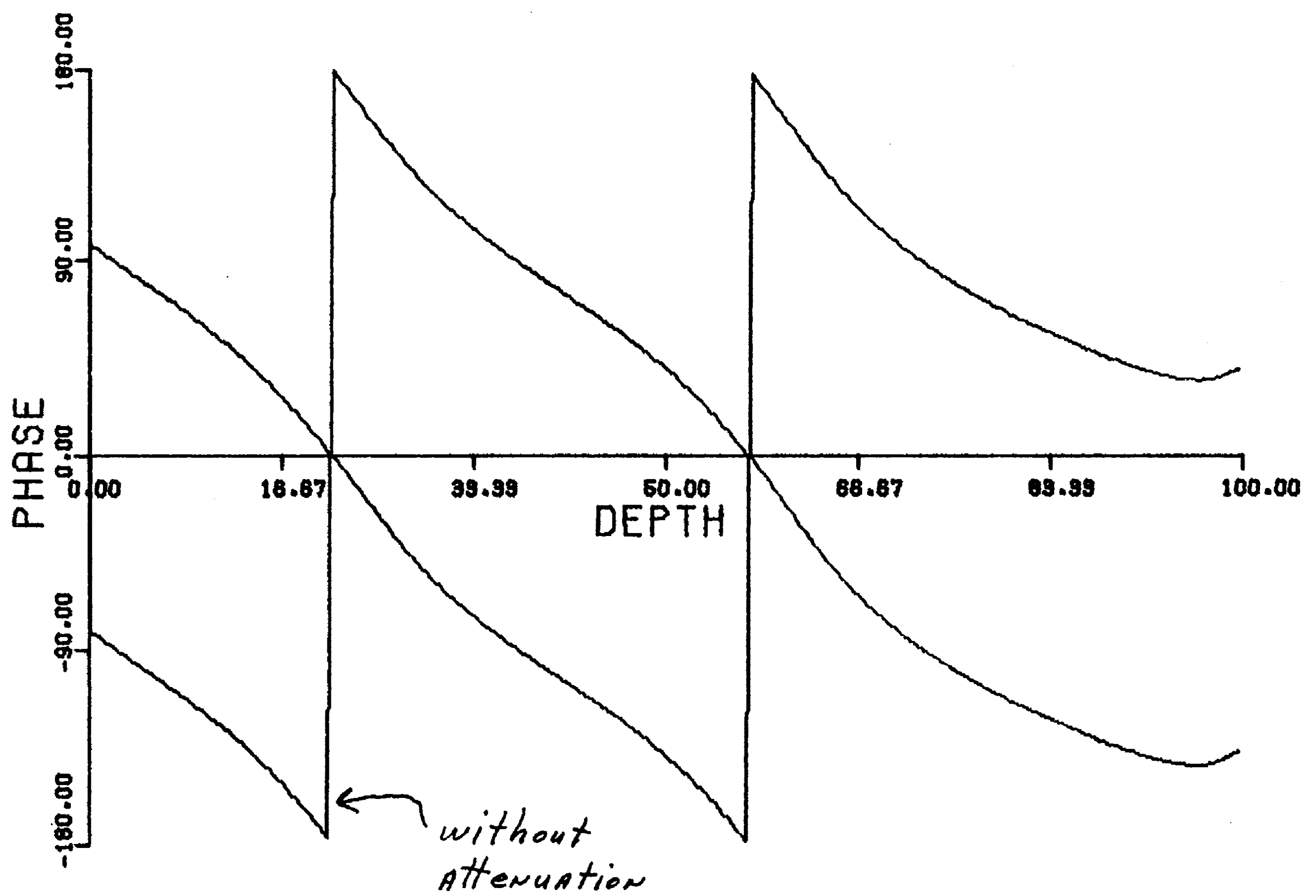


Figure 3.1.7 State phase of response function for two layer ocean with $V=0.0651205m^{-1}$. Basement depth=100m. Source depth=30m.

earlier. In Figure 3.1.5. the upper two lobes have substantially larger magnitude in the attenuated case than in the unattenuated case. The remaining lobes continue the uniformly reduced amplitude trend.

Note that the source depth for this case is 30 meters, and it becomes clear that the larger magnitude region is above the source. This tendency was investigated for various source depths and held true in all cases. The region above the source is slightly amplified and that below slightly attenuated as the horizontal wave number V decreases below critical. Unfortunately, the numerical instability discussed in the following section became apparent for V below 0.059 and this phenomenon could not be investigated further. It is, however, believed to be an accurate description of the influence of attenuation on the response function and not a product of the numerical technique.

A typical plot of the phase of the complex potential is given in Figure 3.1.6. The only effect of attenuation is to shift the phase by approximately 20°. Without attenuation, the phase is initially -175° and remains constant until 22 m depth, where it jumps to about 5°. It then decreases to nearly 540°. All values are plotted on the primary branch of -180° to $+180^{\circ}$ so the phase at 63 m jumps up to $+180^{\circ}$. With attenuation, the phase is initially -195° (plotted at $+165^{\circ}$) and tracks the unattenuated phase throughout the profile. No variation in the phase occurred as a function of wave number, as was noticed in the magnitude figures.

The phase state variable $\theta(\xi)$ for V = 0.059 is shown in Figure 3.1.7. The complex sound speed shifts $\theta(\xi)$ by nearly 180° and, as with the phase of the complex potential, has no other influence. The plots are otherwise exactly identical. This characteristic was consistent throughout the wave number domain.

Stability problems with the technique arose in certain profiles and wave number domains. The source of these problems will be discussed in the next section.

3.2 COMPLEX SOUND SPEED NUMERICAL INSTABILITY

Reference 18 includes a thorough discussion of the numerical stability of the solution of the Riccati and phase differential equations. While investigating the use of complex sound speeds to model attenuation, instability during the integration of the phase differential equation occasionally occurred which prevented a solution from being obtained. Although a specific expression defining the limits within which the solution, with attenuation, converges was not found, a discussion of the cause of the problem is in order.

Consider the phase differential equation:

$$\dot{\theta}(\zeta) = F(\zeta)\sin 2\theta(\zeta) - \Gamma^2\cos^2\theta(\zeta) - \sin^2\theta(\zeta) \qquad 3.2.1$$

Given an ocean model with pure real sound speeds, the solution of 3.2.1 can be obtained. Including complex sound speeds in the basement will alter initial conditions for upward integration of $\dot{\theta}(\zeta)$. To study the change in $\theta(\zeta)$, it is convenient to use a perturbation method. First write

$$\theta(\zeta) = \theta_0(\zeta) + \epsilon\,\delta\theta(\zeta) \qquad 3.2.2$$

$$F(\zeta) = F_0(\zeta) + \epsilon\,\delta F(\zeta) \qquad 3.2.3$$

where the first term on the right hand side of each equation is the unperturbed, without attenuation, solution. The second term is a first-order variation on that solution due to the inclusion of attenuation. To obtain the first order linearized equation for $\delta\theta(\xi)$, expand the right hand side of 3.2.1 about $\theta_0(\xi)$ by a Taylor series.

In general, if

$$\dot{\theta}_0(\xi) = F\left(\theta_0(\xi), \xi, \mathcal{F}_0(\xi), r^2\right) \tag{3.2.4}$$

then

$$\begin{aligned} \dot{\theta}_0(\xi) + \delta\dot{\theta}(\xi) &= F\left(\theta_0(\xi) + \delta\theta(\xi), \xi, \mathcal{F}_0(\xi) + \delta\mathcal{F}(\xi), r^2\right) \\ &= F\left(\theta_0(\xi), \xi, \mathcal{F}_0(\xi), r^2\right) + \left.\frac{\partial F}{\partial \theta}\right|_0 \delta\theta(\xi) \\ &\quad + \left.\frac{\partial F}{\partial \mathcal{F}}\right|_0 \delta\mathcal{F}(\xi) + \text{higher order terms} \end{aligned} \tag{3.2.5}$$

The equation of first variation is

$$\delta\dot{\theta}(\xi) = \left.\frac{\partial F}{\partial \theta}\right|_0 \delta\theta(\xi) + \left.\frac{\partial F}{\partial \mathcal{F}}\right|_0 \delta\mathcal{F}(\xi) \tag{3.2.6}$$

where the partial derivatives are evaluated using unattenuated values of all parameters. Notice that the variation in $\dot{\theta}(\xi)$ is a function of the variation in both $\theta(\xi)$ and $\mathcal{F}(\xi)$. Therefore an equation of variation for the Riccati equation must be studied before continuing with the analysis of equation 3.2.6.

Equation 1.3.16 is of the form

$$\dot{\mathcal{F}}(\xi) = g\left(\mathcal{F}(\xi), \vartheta(\xi)\right) \tag{3.2.7}$$

The equation of first variation is obtained as follows:

$$\dot{\mathcal{F}}_0(\xi) + \delta\dot{\mathcal{F}}(\xi) = g\left(\mathcal{F}_0(\xi), \vartheta_0(\xi)\right) + \left.\frac{\partial g}{\partial \mathcal{F}}\right|_0 \delta\mathcal{F}(\xi) + \left.\frac{\partial g}{\partial \vartheta}\right|_0 \delta\vartheta(\xi)$$

$$\delta\dot{\mathcal{F}}(\xi) = \left.\frac{\partial g}{\partial \mathcal{F}}\right|_0 \delta\mathcal{F}(\xi) + \left.\frac{\partial g}{\partial \vartheta}\right|_0 \delta\vartheta(\xi)$$

$$\delta\dot{f}(\xi) = 2f_o(\xi)\,\delta f(\xi) - 2g_o(\xi)\,\delta g(\xi) \qquad 3.2.8$$

Equation 3.2.8 was obtained via a locally stationary analysis. That is, the solution to 3.2.8 is valid only in a particular neighborhood of $f(\xi)$. The neighborhood is specified by that region of ξ in which $f(\xi)$ is approximately constant. The solution to 3.2.8 is

$$\delta f(\xi) = e^{2f_o(\xi)\Delta\xi} + \frac{g_o(\xi)\,\delta g(\xi)}{f_o(\xi)} \qquad 3.2.9$$

Equation 3.2.9 describes the changes in the Riccati solution that result from a change in the initial conditions. Notice that for integration upward through the water column $\Delta\xi < 0.0$ and therefore 3.2.9 has a stable solution. The conclusion is that the second term on the right hand side of 3.2.6 is not the source of the numerical problem and the first term must now be considered.

The equation of first variation for 3.2.3 is

$$\delta\dot{\theta}(\xi) = \left[(1-r^2)\sin 2\theta_o(\xi) - 2f_o(\xi)\cos 2\theta_o(\xi)\right]\delta\theta(\xi) + \sin 2\theta_o(\xi)\,\delta f(\xi) \qquad 3.2.10$$

To study the instability which arose during the integration of the phase differential equation, the feedback term of equation 3.2.10 (the coefficient of $\delta\theta(\xi)$) must be studied in conjunction with equation 3.2.1.

Before examining 3.2.10 in detail, it is useful to recall a result of Laplace transform theory(37). For the differential equation

$$\dot{X}(\xi) = AX(\xi) \qquad X(0) = 1$$

with transform

$$X(s) = \frac{1}{s-A}$$

the stability of the solution can be analyzed by considering the location of the pole at A. Stable solutions, in the bounded-input bounded-output sense, are obtainable for $A < 0$ and integration down the profile(pole in the left half plane) or for $A > 0$ and integration up the profile(pole in the right half plane). In both cases, the solution is a decaying exponential.

Now reconsider 3.2.10, which has a pole at

$$A = (1-\nu^2)\sin 2\theta_0(\zeta) - 2F_0(\zeta)\cos 2\theta_0(\zeta) \qquad 3.2.11$$

Note again this is a locally stationary analysis. The solution of 3.2.10 will be stable for integration up the profile if $A > 0$. Therefore, the local requirement is that

$$\operatorname{TAN} 2\theta_0(\zeta) \geq \frac{2F_0(\zeta)}{1-\nu^2} \qquad 3.2.9$$

If stable solutions of 3.2.1 can be obtained such that 3.2.9 is satisfied, then the use of complex sound speeds to model attenuation would be acceptable. If not, however, then the numerical instability will result. A discussion of the critical points of 3.2.1 is now in order.

Baggeroer(18) discussed the importance of the equilibrium points of equation 3.2.1. In general, the larger of the two roots of 3.2.1, given by

$$\operatorname{TAN} \theta(\zeta) = F_8^2(\zeta) \pm \sqrt{F_8^2(\zeta) - \nu_0^2} \qquad F_8^2(\zeta) > \nu_0^2$$

is stable for upward integration. Modes form when,at some point,

$$\nu_0^2 > F_8^2(\zeta)$$

and the equilibrium points disappear. In this case $\dot{\theta}(\zeta) < 0.0$ and $\theta(\zeta)$ is continually decreasing up the profile. Note, for example, figure 3.1.7 where $\theta(\zeta)$ has been plotted modulo 360.

The source of the instability is now clear. As Γ_0^2 increases, the lower boundary on the region for stable solutions of 3.2.10, as given by 3.2.12, increases(the denominator becomes more negative, therefore $\theta_0(\zeta)$ must approach zero from below the origin). However, the solution of 3.2.1 tracks $\theta_0(\zeta)$ in the opposite direction- by continually decreasing. At some point, 3.2.12 is no longer satisfied and A becomes less than zero. It is important to note that the solution to 3.2.10 is unstable in this case at the same time that the solution to 3.2.1 is stable. If the solution to 3.2.1 were not stable the entire analysis would be meaningless.

This unstable solution is not solely the cause of the breakdown of the numerical technique. It means simply that the perturbed solution diverges from the unperturbed solution in these regions of the depth profile. As A becomes increasingly more negative, the solution becomes relatively more unstable. In other words, $\theta(\zeta)$ in equation 3.2.2 diverges from $\theta_0(\zeta)$.

The conclusion is that the region of convergence of the solution with attenuation is controlled by the sound speed profile through $f_0(\zeta)$, the horizontal wave number through Γ^2, and the amount of attenuation through $\delta\theta(\zeta)$. The ability to study the attenuation characteristics of any given ocean model must be determined to a large extent by trial and error since the technique has this inherent instability.

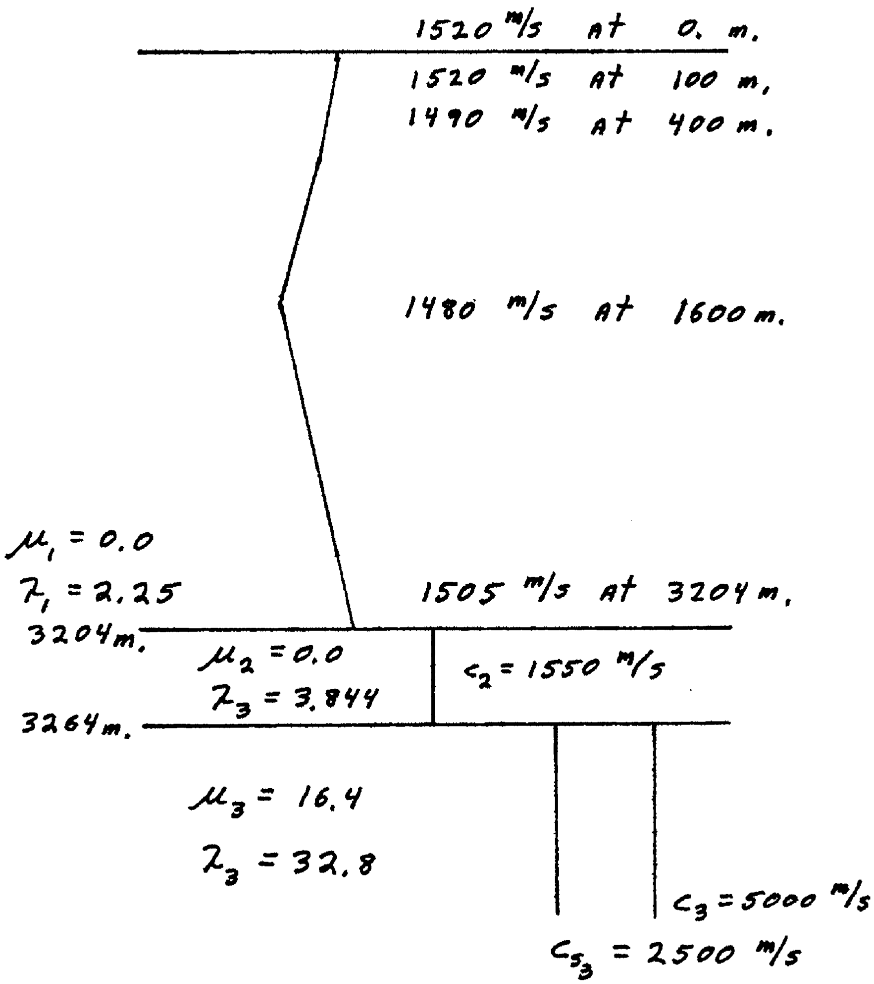
3.3 Deep Ocean Model Considerations

The thesis to this point has dealt entirely with shallow water ocean models. This section will begin with a discussion of a simple deep ocean model that consists of a fluid layer and a thin sediment layer assumed to act as a fluid. The basement is allowed to be elastic. It will conclude with an investigation of the technique for ocean models in which the bottom is a multilayered elastic medium.

A simple deep ocean model is depicted in figure 3.3.1. The seafloor, characterized as young oceanic crust, consists of a thin sediment layer that separates the ocean from the elastic basement. The sediment is assumed fluid since typical shear speeds are generally small and resulting shear waves of negligible amplitude. The shallow water models of chapter 2 are proof of this assumption. In both of those cases the basement shear field had substantially smaller magnitude than the compressional field.

The sampling density requirement that N_λ=16 was derived from the study of a simple constant sound speed shallow water model. A question that arises is whether or not this density is valid for the more complex model. Since the exact solution cannot be written, another method of verification must be found. Reconsideration of figures 2.3.1 through 2.3.4 suggests the method. Notice that as the sampling density is increased, the state solution approaches the exact solution in steps of decreasing size. For example, in figure 2.3.4, the state solution has approximately one-half the magnitude of the exact solution. In figure 2.3.3, the ratio is about nine-tenths and in figures 2.3.2 and 2.3.1, the difference is almost negligible. By varying the sampling density and computing the response function for figure 3.3.1, this asymptotic tendency can be used to study the validity of N_λ=16 for the deep ocean model.

The result of this study is that N_λ=16 is valid in general. Accurate solutions can be expected for all profiles in which this criterion is satisfied. As with the shallow water study, values below 16 appeared occasionally to be acceptable. These occurrences were infrequent and unpredictable however.



Simple Deep Ocean Model

Figure 3.3.1 λ and μ have units dynes/cm$^2 \cdot 10^{-5}$

Study of the deep ocean profile for decreasing wavenumber V gave the same results as was obtained earlier for the shallow water cases. N_z should be increased as V is decreased, especially in the continuous mode region. Good procedure would involve selecting several typical horizontal wave numbers and computing the response of each for N_z=12, N_z=16, and N_z=20. Study of the results should then aid in selecting the optimal sampling density.

Figures 3.3.2, 3.3.3, and 3.3.4 are typical response functions for figure 3.3.1 where an eight hertz source is at 400 meters. The horizontal wave numbers V=0.0053 and V=0.00469 are near the fifth and seventeenth modes of the 18 mode profile and V=0.003 is in the continuous region.

Notice that the basement shear waves, which are plotted at full value, are on the same order of magnitude as the compressional waves. In all cases shear has the larger magnitude, substantially so for V=0.003. This is to be expected for the dense, high speed basement. The excitation of shear is a much more important attenuation mechanism than in any of the models considered earlier. In addition, the propagating shear wave present in the basement for V=0.003 appears to have a strong influence on the compressional wave response function below the source. This is a good example of the type of phenomenon that can be easily investigated using the state variable algorithm.

A more complex model of the same ocean region is shown in figure 3.3.5. Shear propagation is allowed in the sediment layers and the basement is divided into two elastic media. Most of the listed parameters and sound speeds are assumed from laboratory experiments and field data(33,38). The remainder of the section will discuss the application of the algorithm to this model.

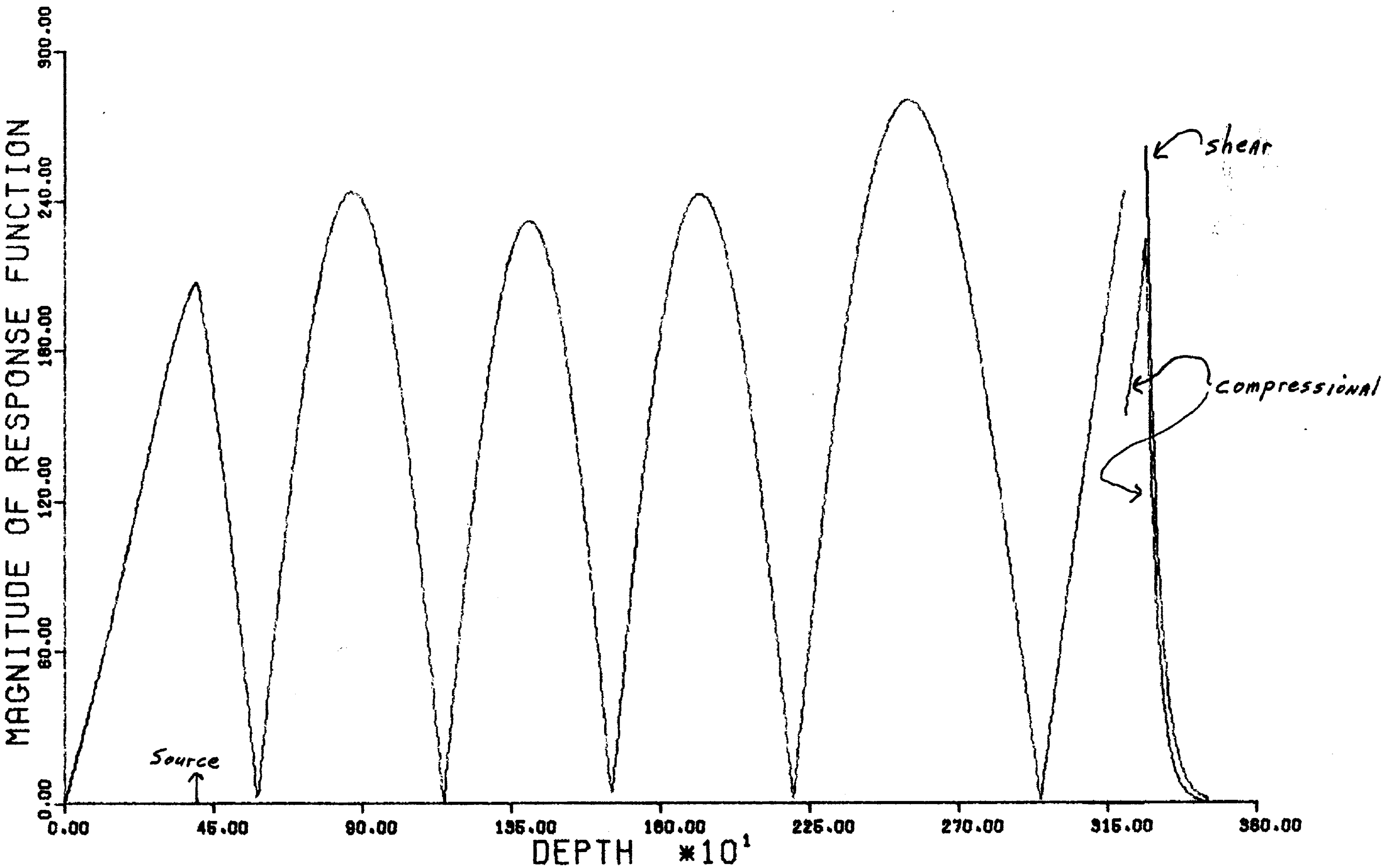


Figure 3.3.2 Response function for deep ocean case with V=0.0053m^{-1}. Basement depth=3264m. Layer depth=3204m. Source depth=400m.

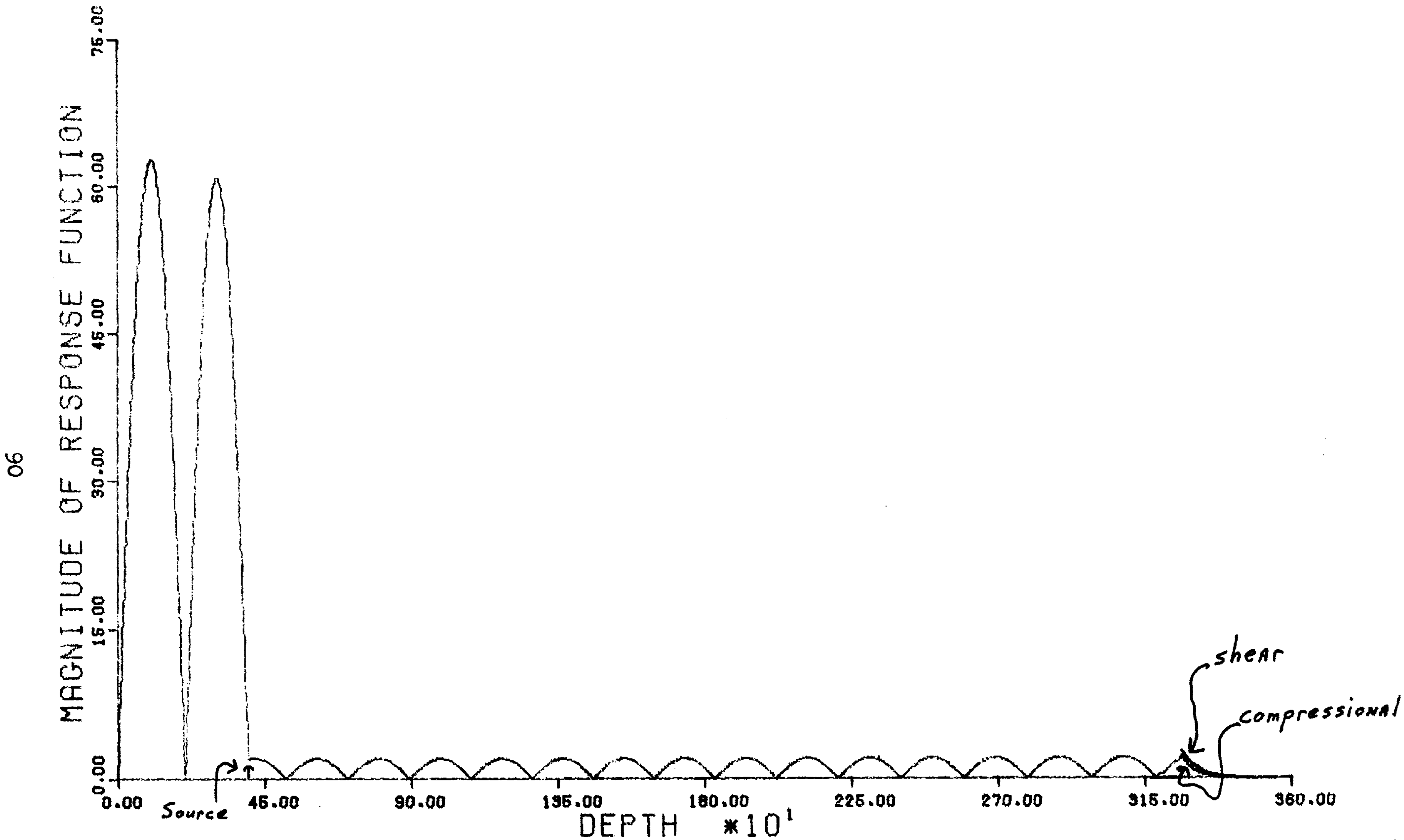


Figure 3.3.3 Response function for deep ocean case with $V=0.00469m^{-1}$. Basement depth=3264m. Layer depth=3204m. Source depth=400m.

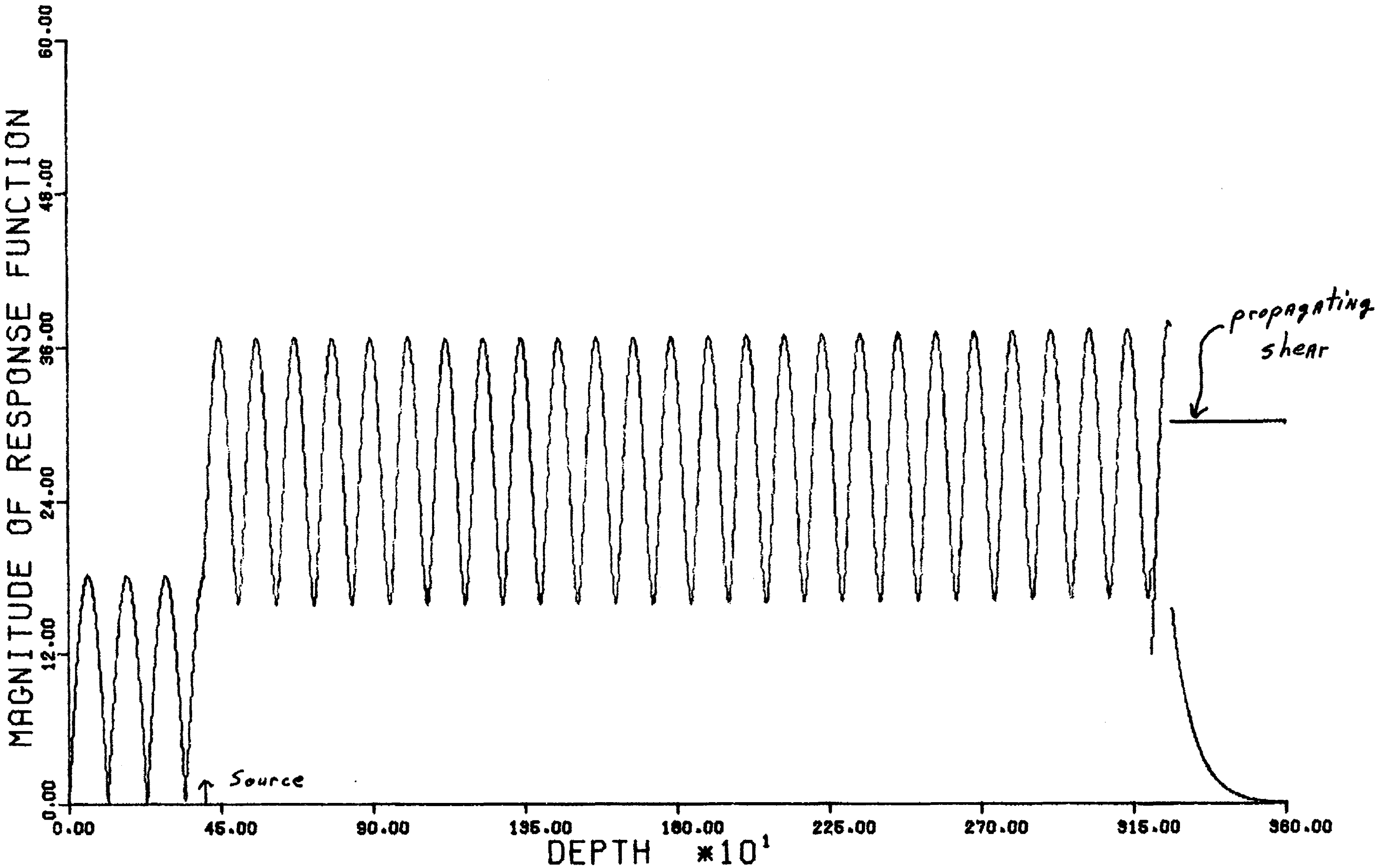
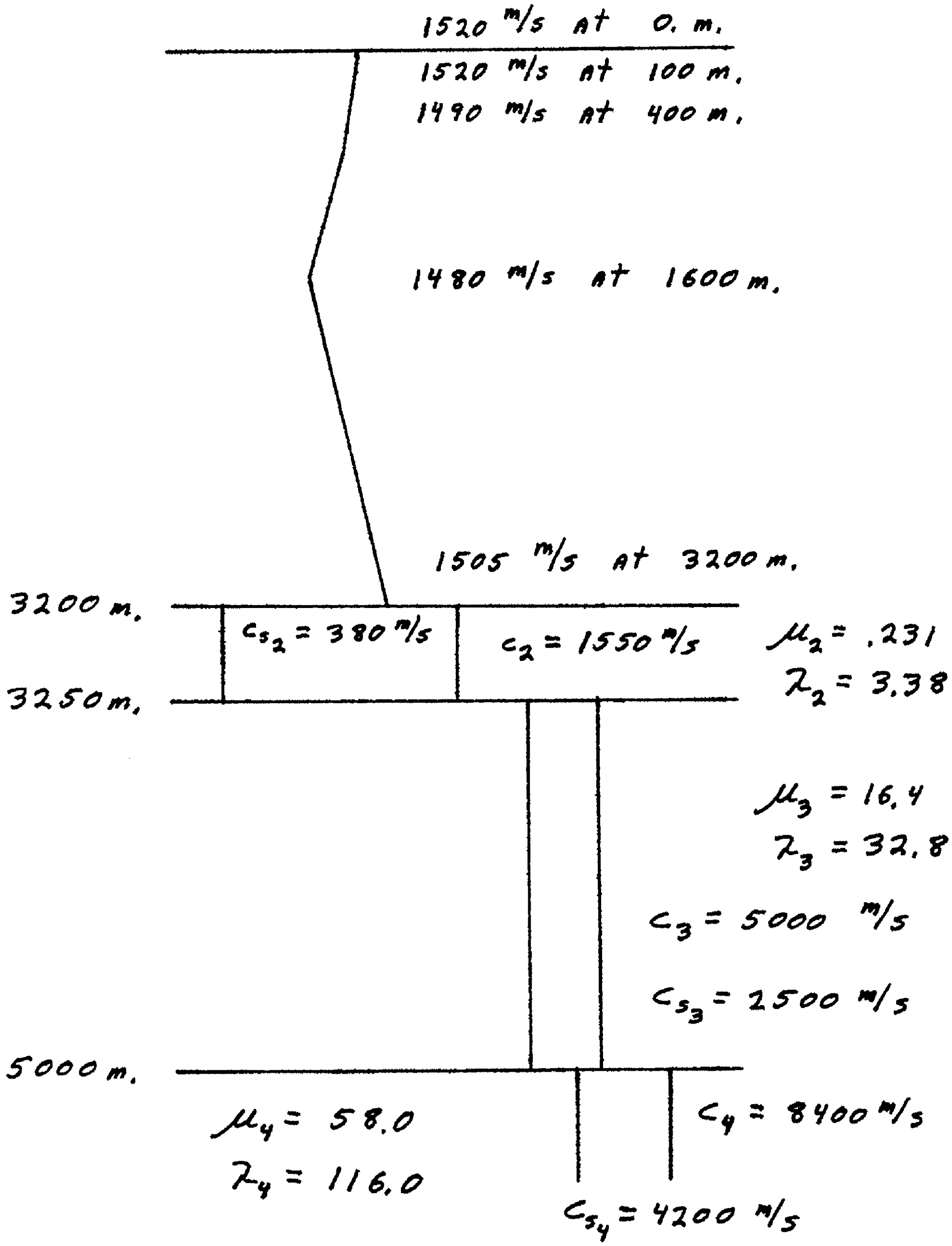


Figure 3.3.4 Response function for deep ocean case with $V=0.003m^{-1}$. Basement depth=3264m. Layer depth=3204m. Source depth=400m.



Multilayered Deep Ocean Model

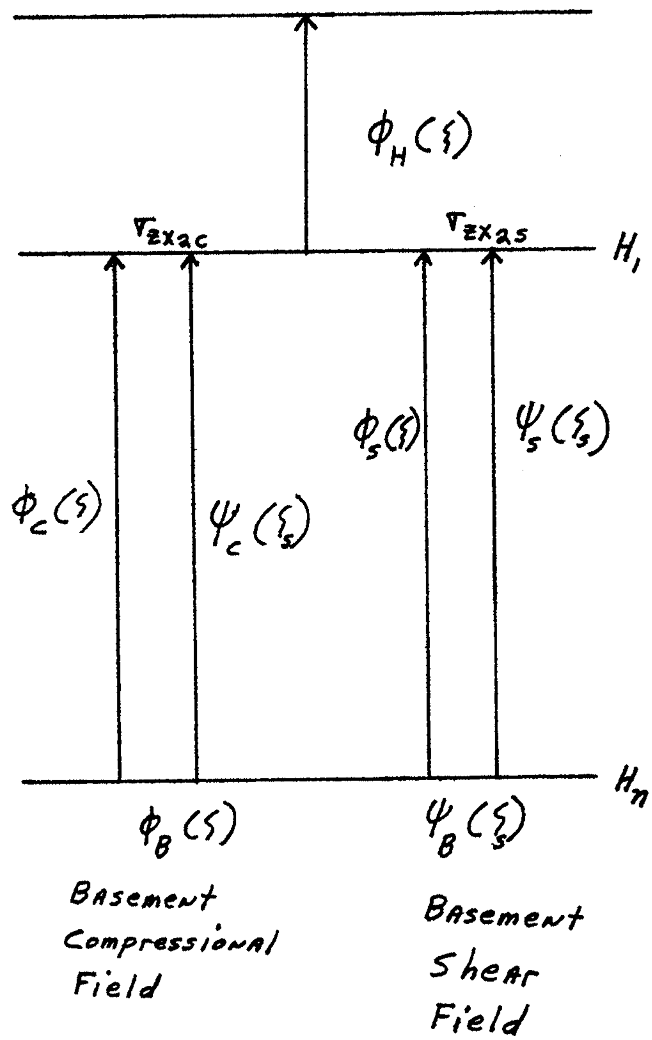
Figure 3.3.5 λ and μ have units dynes/cm^2 10^{-5}

From section 1.3 and figure 3.3.6 recall the method of solution for this model. Two independent integrations of the state equations will occur. The first will use $\phi_B(\zeta)$ as the initial condition and the second $\psi_B(\zeta_s)$. The principle of superposition will then be used to combine the two solutions in a manner that satisfies equation 1.4.6c. This superposition amounts to a scaling of $\phi_s(\zeta)$ and $\psi_s(\zeta_s)$ such that the total tangential stress at H is zero. In particular

$$\tau_{zx_{2T}} = 0.0 = \tau_{zx_{2C}} + A_1 \tau_{zx_{2S}} \qquad 3.3.1$$

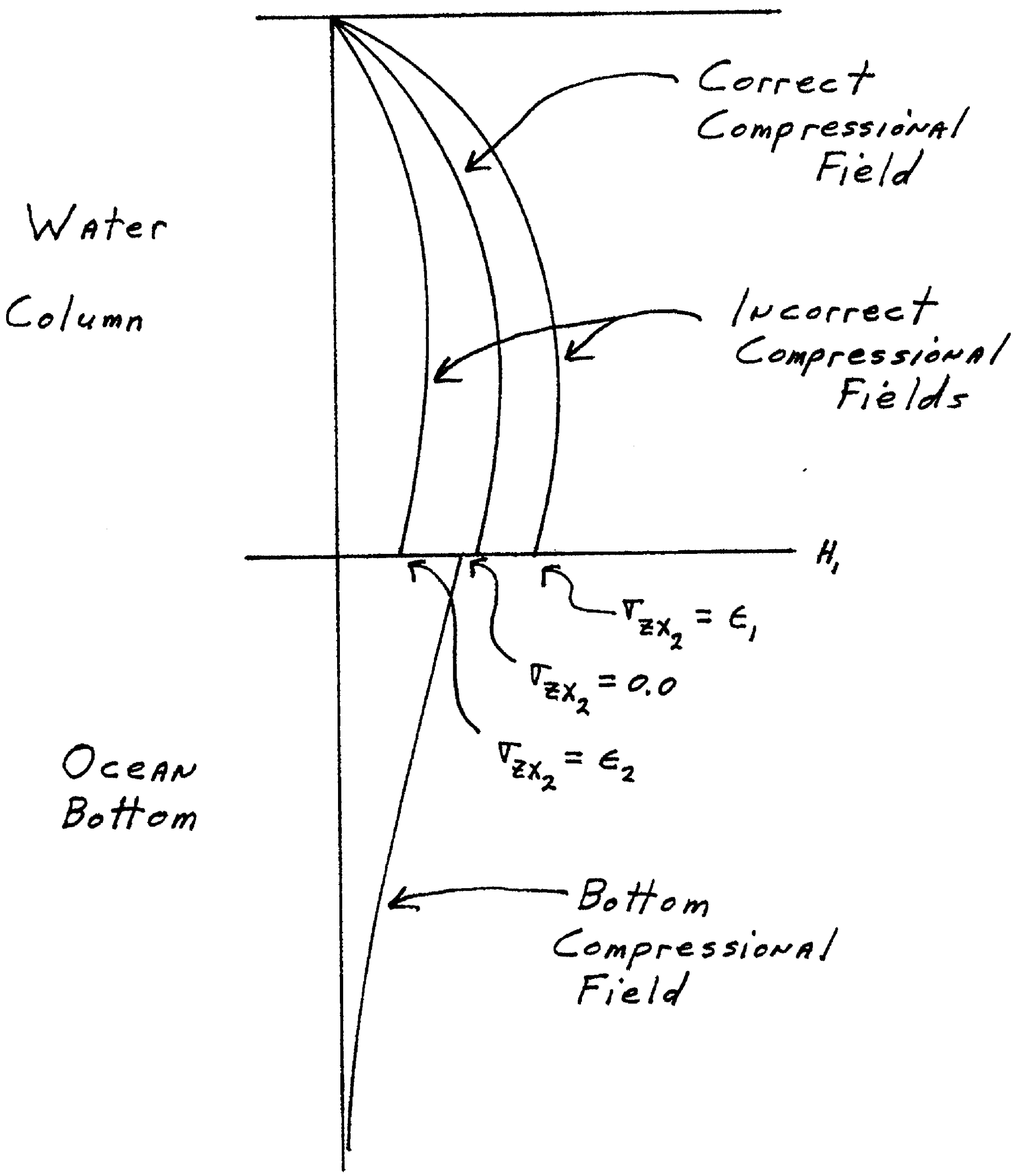
where A_1 is the appropriate scaling factor. This procedure determines the relative magnitude of the total shear field to the total compressional field and specifies one of the unknown constants of equation 1.3.15. Equation 1.3.6a and 1.3.6b are then used to obtain $\phi_H(H_1)$ and integration proceeds to the ocean surface. The remaining constant of 1.3.15 is then determined and the entire solution scaled to satisfy equation 1.3.7.

This theory breaks down upon implementation however. Enough precision cannot be retained during the computation to insure that $\tau_{zx_{2T}}$ does in fact equal zero. Referring again to figure 3.3.6, the apparent cause of the problem lies in the relative magnitude of $\phi_c(\zeta)$, $\psi_c(\zeta_s)$, $\phi_s(\zeta)$, and $\psi_s(\zeta_s)$. $\psi_s(\zeta_s)$ is generally half an order of magnitude smaller than $\psi_c(\zeta_s)$. $\psi_c(\zeta_s)$ in turn is generally at least two orders of magnitude smaller than $\phi_c(\zeta)$ and $\phi_s(\zeta)$. The result is that the constant A_1 in equation 3.3.1 is strongly dependent on $\phi_c(\zeta)$ and $\phi_s(\zeta)$ and weakly dependent on $\psi_c(\zeta_s)$ and $\psi_s(\zeta_s)$. Difficulty then arises in keeping $\tau_{zx_{2T}}=0.0$. Figure 3.3.7 depicts this sensitivity problem. For convenience, the shear field and the bottom elastic layers are not drawn. As was discussed earlier, a total compressional field is computed for the ocean bottom. The correct solution requires that the tangential stress at H_1 equal zero. In this case, the appropriate continuity



Multilayered Model Integration

Figure 3.3.6



Stress Sensitivity Example

Figure 3.3.7

conditions are used to calculate the compressional field in the water column, and integration continues to the ocean surface. However, the relative magnitudes of the potentials cause difficulty in controlling the exact value of the tangential stress, and in general the stress has a small nonzero value, depicted as ϵ_1 and ϵ_2 in figure 3.3.7. Notice that although the bottom compressional field is not influenced by the stress error, the water column compressional field is extremely sensitive to the error. As a result, the correct water column compressional field could not be computed. No other criterion exists for determining the correct magnitude of the compressional field above H_1. The possibility of using double precision arithmetic was precluded by the requirement of complex variables. The conclusion is that further analysis of the state variable algorithm is required before being useful in the study of oceanic models of multilayered media.

To conclude, an accurate and efficient technique for computing the Green's function solution to the depth-separated wave equation has been presented. The technique has no inherent limitations when sound speeds are real quantities. The use of complex sound speeds to simulate attenuation is limited. In most cases attenuation is acceptable and solutions for wavenumbers corresponding to the discrete mode region. Study of the continuous mode region is limited.

The technique can be applied to both shallow and deep ocean sound speed profiles. The basement can be modelled as elastic and one or more sediment layers assumed to act as fluids are acceptable. Sensitivity limitations do not allow the modelling of the basement as a multilayered elastic medium.

APPENDIX I

$C(z)$	= Compressional Sound Speed
C_0	= Compressional Sound Speed Minimum
$C_s(z)$	= Shear Sound Speed
C_{s0}	= Shear Sound Speed Minimum
D	= Transmission Coefficient From 1st to 2nd Layer
$\delta(\vec{r}-\vec{r}_0)e^{-i\omega t}$	$= \delta(x-x_0)\ \delta(y-y_0)\ \delta(z-z_0)\ e^{-i\omega t}$ Harmonic Sound Source At (x_0, y_0, z_0)
$\Delta\xi$	= Integration Step Size
f	= Frequency
$f(\xi)$	= Riccatti Parameter
$G_u(z,z_0)$	= Green's Function Solution Above Source at z_0
$G_L(z,z_0)$	= Green's Function Solution Below Source at z_0
H_i	= Depth of i^{th} Layer
k	= Radian Wave Number $2\pi f/c$
k_s	= Shear Radian Wave Number $2\pi f/c_s$
k_{z_i}	= Vertical Radian Wave Number in i^{th} Layer
k_{zs_i}	= Vertical Shear Radian Wave Number in i^{th} Layer
λ_i	= Lamé's Constant; i^{th} Layer
λ_0	= Maximum Compressional Wavelength $\lambda_0 = f/c_0$
λ_{0s}	= Maximum Shear Wavelength $\lambda_{0s} = f/c_{0s}$
$M(\xi)$	= Magnitude of Compressional Potential In Phase Plane
$N(\xi)$	= Magnitude of Shear Potential In Phase Plane
μ	= Rigidity of i^{th} Layer

P	= Transmission Coefficient of Shear Wave in 2nd Layer Excited by Compressional Wave in 1st Layer
P_{ij}	= Stress Component, Convention Related to 3-D Cube, i Referring to Direction of Face On Which Stress Is Acting and j Being Direction On Which Stress Is Acting
$P(\xi)$	= Compressional State Variable
$P_s(\xi)$	= Shear State Variable
$\phi(z)$	= Compressional Velocity Potential
$q(\xi)$	= Normalized Compressional Sound Speed Parameter
$q_s(\xi)$	= Normalized Shear Sound Speed Parameter
R_s	= Surface Compressional Reflection Coefficient
ρ_i	= Density i^{th} Layer
$\psi(z)$	= Shear Velocity Potential
σ^2	= Normalized Compressional Vertical Wave Number
$\sigma_s^{\ 2}$	= Normalized Shear Vertical Wave Number
$\theta(\xi)$	= Phase of Compressional Potential in Phase Plane
$\gamma(\xi)$	= Phase of Shear Potential in Phase Plane
θ_i	= Direction of Plane Wave Propagation in i^{th} Layer
γ_i	= Direction of Shear Propagation in i^{th} Layer
θ_d	$= \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}$ = dilatation
$\vec{V}$	= (u, v, w) Velocity Vector
V	= Horizontal Wave Number
V_{ij}	= Reflection Coefficient for Wave in j^{th} Layer Off Of i^{th} Interface
z_i	= Acoustic Impedance in i^{th} Layer
z_{s_i}	= Shear impedance in i_{th} Layer

ξ = Normalized Compressional Depth Parameter

ξ_s = Normalized Shear Depth Parameter

APPENDIX II

The implementation of the algorithm of Chapter I for the HP 2100 computer consists of a main program "SMAIN" and four subroutines "SWBC", "RNSV","SCSPH", and "SSBC". "SMAIN" asks a series of questions which completely specify the ocean model to be studied. Table AII-1 lists these quantities. This profile is then displayed on the console for review by the operator before calculation begins.

The four subroutines are called at various times during calculation of the response function. "SWBC" implements equations 1.4.6, given the potentials in the second layer. "SSBC" implements equations 1.4.5, given the potentials in the lower layer. "RNSV" computes the transformation of equations 1.5.1,1.5.2,1.5.5, and 1.5.6. "SCSPH" integrates the differential equations 1.5.3,1.5.4,1.5.7, and 1.5.8 upwards in any layer.

The calculated compressional and shear potentials are stored in disc files for later use. In addition, printed output of most important quantities is provided. Table AII-2 lists those quantities.

Figure AII-1 is a flowchart of program "SMAIN". It can be divided into several general calculation sections. The first section contains the input sequence as well as the computation of the normalized sound speed parameters $q(\zeta)$ and $q_s(\zeta_s)$ and Riccati parameters $f(\zeta)$ and $f_s(\zeta_s)$. It also contains the integration of the linear state equations for the particular solution. Initialization of the potentials as discussed in §1.4 then occurs.

The decision block titled "interface number" requires explanation. Each interface is numbered, with the fluid-sediment interface(the uppermost interface) number 1. Integration starts at the lowest interface. Therefore, for a one layer model, the interface number is one and the program jumps to "SWBC".

For two or more layers the superposition principle of §1.4 is required and "SMAIN" continues to "SSBC", after setting $\psi_H = \dot{\psi}_H = 0.0$ as discussed earlier. The trio of subroutines then computes the field in the layer up to the next interface. If this interface number is greater than one, the procedure is repeated. If not, the program continues with the calculation of the shear excited field in a completely analogous manner.

Once the fields due to compressional and shear excitation have both been calculated, the correction factor of equation 3.3.2 can be computed and the total field in the elastic layers obtained by superposition. The next sequence of subroutines then calculates the homogeneous compressional solution in the water column, which is combined with the particular solution in the following step. Finally,the basement solution is computed.

The numerical integration technique used throughout "SMAIN" is a third-order Adams-Bashforth method(27). This method provides the same accuracy as the more common Runge-Kutta techniques, and is more straightforward to use in the desired application.

Notice that this program has been organized to accomodate any desired ocean model, including those which were found not solvable in Chapter 3. This allows further study of the problems which these models have presented.

"SMAIN" stores all information in fifteen binary data disc files, and all computations are done in groups of 32 integration steps, which is the maximum number of complex quantities that can be stored on one sector of the disc. Consequently, the disc file length in sectors should be

$$S \geq \frac{H_{N+1}}{\Delta z} \cdot \frac{1}{32}$$

where Δz is the unnormalized integration step size and H_{N+1} the depth in the basement to which the solution is to be calculated.

TABLE AII-1

FRQ	=	Complex Sound Speed
ZBS	=	Basement Depth
CMN	=	Minimum Value of Compressional Sound Speed(real number)
SCMN	=	Minimum Value of Shear Sound Speed(real number)
NDZ	=	Number of Integration Points
KBT	=	Number of Sound Speed Transition Points
CSSP(i)	=	Complex Compressional Sound Speed at Transition Point i
SSSP(i)	=	Complex Shear Sound Speed at Transition Point i
ZBT(i)	=	Depth at Transition Point i
NLAYS	=	Number of Layers
JPB	=	Basement Type; =1 if propagating, =0 if rigid
MU(j)	=	Rigidity in Layer j
LAM(j)	=	Lame's Constant in Layer j
LDEP(j)	=	Depth of Layer j
CCBS	=	Complex Compressional Basement Speed
SCBS	=	Complex Shear Basement Speed
VP	=	Complex Horizontal Wave Number
S	=	Depth of Source
SS	=	Strength of Source

TABLE AII-2

The following data is printed out only when the appropriate sense switch is on.

Switch 1	Sound Speed Profile Data
Switch 2	Riccati Data
Switch 3	Particular Solution
Switch 4	RNSV Data
Switch 5	SCSPH Data
Switch 6	Superposed Total Solution

The following data is printed out only when the appropriate sense switch is off.

Switch 7	Total Solution Above Basement
Switch 8	Total Solution In Basement

When switches 7 and 8 are off and switch 10 is on, data for each integration step is printed out. If 10 is off , a number of steps are skipped between each step that is printed out; the interval equal to NDZ/32.

Switch 11 must be off to use the batch input mode.

Switch 12, when on, prevents calculation of the basement solution.

Switch 13, when on, prevents calculation of sound speed and Riccati information and is useful when information stored in disc files from previous computation will remain unchanged.

Switch 14, when on, prevents calculation of particular solution and is useful when information stored in disc file from previous computation will remain unchanged.

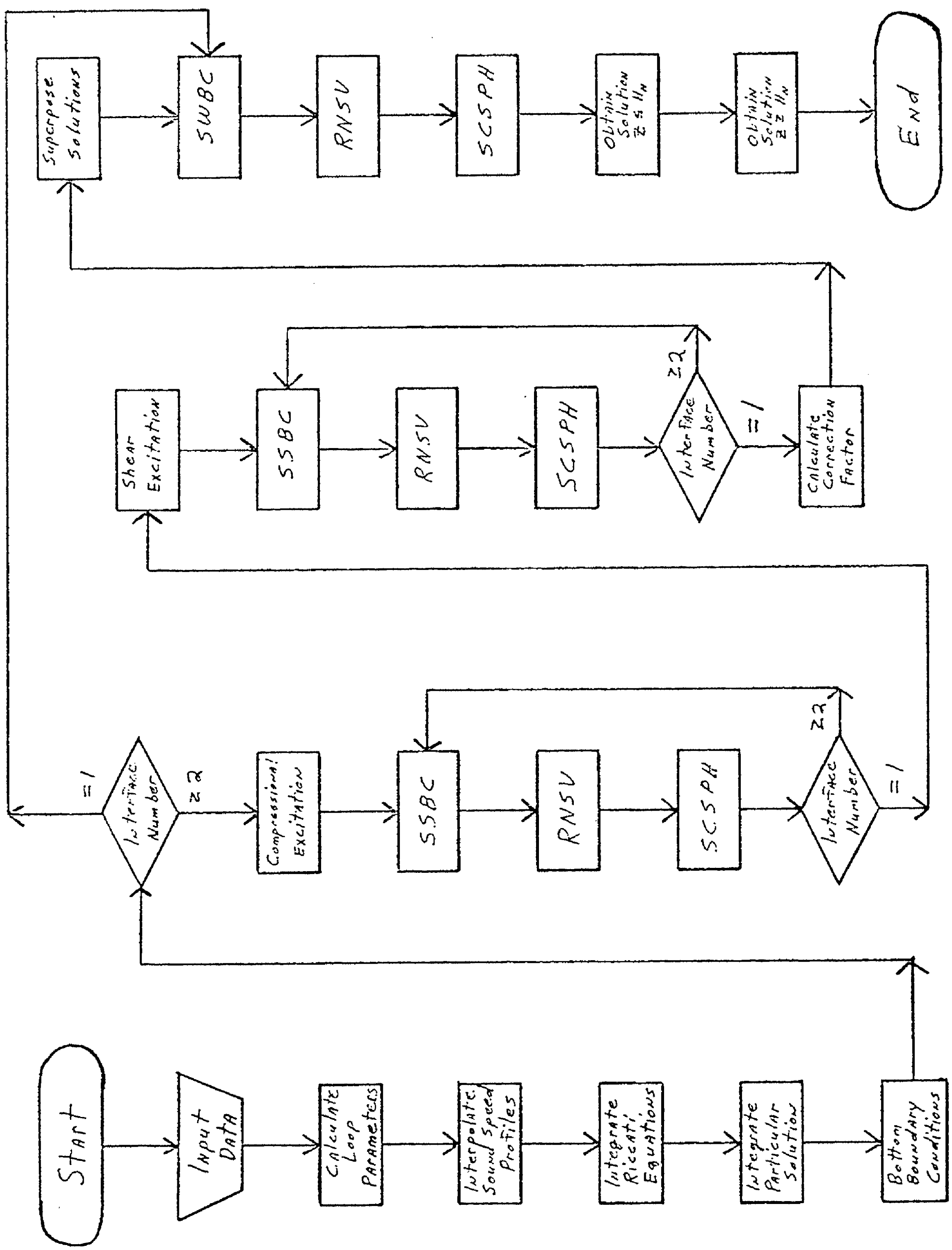

Start
Input Data
Calculate Loop Parameters
Interpolate Sound Speed Profiles
Integrate Riccati Equations
Integrate Particular Solution
Bottom Boundary Conditions
Interface Number
=1
≥2
Compressional Excitation
SSBC
RNSV
SCSPH
Interface Number
≥2
=1
Shear Excitation
SSBC
RNSV
SCSPH
Interface Number
≥2
=1
Calculate Correction Factor
Superpose Solutions
SWBC
RNSV
SCSPH
Obtain Solution z ≤ H_N
Obtain Solution z ≥ H_N
End

Figure AII-1.

Program SMAIN

```
&SMAIN T=00003 IS ON CR00032 USING 00024 BLKS R=0000

FTN4
C  TO COMPUTE THE SOLUTION TO THE INHOMOGENEOUS HELMHOLTZ EQUATION
C  FOR A FLUID OCEAN OVERLYING A LAYERED ELASTIC BOTTOM.  FREQUENCY
C  HORIZONTAL WAVE NUMBER AND ALL SOUND SPEEDS ARE COMPLEX.  LAME
C  CONSTANT AND RIGIDITY ARE USED TO DESCRIBE THE ELASTIC MEDIA AN
C  ARE REAL NUMBERS.  SOUND SPEED PROFILE IS DESCRIBED BY STRAIGHT
C  LINE SEGMENTS DRAWN BETWEEN TRANSITION POINTS WHICH OCCUR AT
C  ALL CHANGES IN SLOPE OF THE PROFILE.
C
C  THE TOTAL NUMBER OF GRID POINTS MUST BE DIVISIBLE BY 32.
C
C  THE NUMBER OF GRID POINTS MUST BE SELECTED SUCH THAT A POINT FA
C  ON EACH INTERFACE BETWEEN LAYERS.  TRANSITION FROM ONE LAYER TO
C  NEXT MUST OCCUR EXACTLY AT THE DEPTH OF THE INTERFACE.
      PROGRAM MAIN
      COMPLEX CFQCH,SFQBH,XLMS,XLMS2,XLM2
      COMPLEX W1,W2,W3,W4,JAY,WEQ,PPB,PSB
      COMPLEX FRQ,CSSP,SSSP,CCBS,SCBS,XLM,CXQ,SXQ,CS,SS,CFQX,SFQX
      COMPLEX CFQP,SFQP,CDFQ,SDFQ,CDFQP,SDFQP,SMPR,CFQ,SFQ
      COMPLEX CCL,SCL,CCU,SCU,CFQBS,SFQBS,DZ,VS,VP,SF1,SF2,CF1,CF2
      COMPLEX TH,M,GA,N,THDOT,MDOT,GADOT,NDOT,PH,PHDOT,PP,PPDOT
      COMPLEX SI,SIDOT,PS,PSDOT,SIGS,SIGP,RP,RS,THB,GAB,YBF,ZBF,PH
      COMPLEX PHII,PRII,DPHI,DPRI,PHI,PRI,DPRII,DPHII,RP1,RS1,SIB
      COMPLEX COR,A1,A2,A3,A4,W5,W6,W7,W8,YP,DZS,SF3,CF3
      COMPLEX B1,B2,B3,B4,B5,B6,B7,B8,SWEQ,VP2,VS2,Y2
      COMPLEX A,M,E,G,F,H,C,D,Y1,Y2,Y3,Y4,D5,D6,D7,D8,D1,D2,D3,D4
      DIMENSION W1(32),W2(32),W3(32),W4(32),W5(32),W6(32),W7(32)
      DIMENSION W8(32),YP(32),YBF(32),ZBF(32),XBF(32),ZBT(25),CSSP
      DIMENSION SSSP(25),MU(15),CXQ(32),SXQ(32),SFQX(32),CFQX(32)
      DIMENSION ALAM(15),LDEP(15),IF1(3),IF2(3),IF3(3),IF4(3),IF5(
      DIMENSION IF6(3),IF7(3),IF8(3),IF9(3),IF10(3),IF11(3),IF12(3
      DIMENSION IF13(3),IF14(3),IF15(3),DN(15)
      REAL MU,LDEP,MAG,MAGS,MAGB
      DATA PI,JAY,IN/3.1415926,(0.0,1.0),1/
      IF1(1)=2HDW
      IF1(2)=2H1
      IF2(1)=2HDW
      IF2(2)=2H2
      IF3(1)=2HDW
      IF3(2)=2H3
      IF4(1)=2HDW
      IF4(2)=2H4
      IF5(1)=2HDW
      IF5(2)=2H5
      IF6(1)=2HDW
      IF6(2)=2H6
      IF7(1)=2HDW
      IF7(2)=2H7
      IF8(1)=2HDW
      IF8(2)=2H8
      IF9(1)=2HDY
      IF9(2)=2HP
      IF10(1)=2HDY
      IF10(2)=2HBF
      IF11(1)=2HDZ
      IF11(2)=2HBF
      IF12(1)=2HDC
      IF12(2)=2HXQ
      IF13(1)=2HDS
      IF13(2)=2HXQ
      IF14(1)=2HDC
```

```
      IF14(2)=2H R
      IF14(3)=2HX
      IF15(1)=2HDS
      IF15(2)=2HFQ
      IF15(3)=2HX
      IF1(3)=2H
      IF2(3)=2H
      IF3(3)=2H
      IF4(3)=2H
      IF5(3)=2H
      IF6(3)=2H
      IF7(3)=2H
      IF8(3)=2H
      IF9(3)=2H
      TPI=PI*2
      IF10(3)=2H
      IF11(3)=2H
      IF12(3)=2H
      IF13(3)=2H
      PI2=PI**2
      IF(ISSW(11)) 99,98
   98 IN=22
   99 CONTINUE
      TPISQ=TPI**2
C
C  INPUT
C
    5 WRITE(1,100)
  100 FORMAT("TYPE COMPLEX FREQUENCY,NUMBER OF GRID POINTS "
     1" AND NUMBER OF SOUND SPEED TRANSITIONS")
      READ(IN,*) FRQ,NDZ,KBT
      CMN=10000.0
      SCMN=10000.0
      DO 102 J=1,KBT
      WRITE(1,103) J
  103 FORMAT("TYPE COMPLEX C AND S SPEEDS AND DEPTH AT TRANSITION"
      READ(IN,*) CSSP(J),SSSP(J),ZBT(J)
      IF(CMN.GT.REAL(CSSP(J))) CMN=REAL(CSSP(J))
      IF(SCMN.GT.REAL(SSSP(J)).AND.SSSP(J).NE.0.0) SCMN=REAL(SSSP(
  102 CONTINUE
      IF(SCMN.EQ.10000.0) SCMN=CMN
      WRITE(1,105)
  105 FORMAT("TYPE # OF LAYERS AND 1 OR 0 IF PROPAGATING OR RIGID"
      READ(IN,*) NLAYS,JPB
      NL=NLAYS-1
      DO 106 J=1,NL
      WRITE(1,107) J
  107 FORMAT("TYPE VISCOSITY, LAME AND DEPTH OF LAYER ",I3)
  106 READ(IN,*) MU(J),ALAM(J),LDEP(J)
      ZBS=LDEP(NL)
      IF(JPB.EQ.0) GO TO 153
      WRITE(1,104)
  104 FORMAT("TYPE COMPLEX BASEMENT C AND S SPEEDS, COMPUTATION "
     1"DEPTH,VISCOSITY AND LAME")
      READ(IN,*) CCBS,SCBS,LDEP(NLAYS),MU(NLAYS),ALAM(NLAYS)
  153 WRITE(1,109)
  109 FORMAT("TYPE COMPLEX HORIZ. WAVE NUMBER,SOURCE DEPTH "
     1"AND SOURCE STRENGTH")
      READ(IN,*) VP,S,SS
C
C  DISPLAY PROFILE PARAMETERS
C
      WRITE(1,160) FRQ,NDZ,VP,S
  160 FORMAT("FREQUENCY ",2F6.2,/"NUMBER OF GRID POINTS ",I7,
     1/"HORIZ. WAVE NUMBER ",2F10.8,/"SOURCE DEPTH",F6.0)
      DO 169 J=1,NL
  169 WRITE(1,162) J,MU(J),ALAM(J),LDEP(J)
```

```
162 FORMAT("LAYER #",I2/10X,"VISCOSITY "F10.5,/
   110X,"LAME CONSTANT ",F10.5,/10X,"DEPTH ",F10.5)
    IF(JPB.EQ.1) WRITE(1,462) MU(NLAYS),ALAM(NLAYS)
462 FORMAT("BASEMENT"/10X,"VISCOSITY",F10.5,/10X,
   1"LAME CONSTANT",F10.5)
    DO 170 J=1,KBT
170 WRITE(1,163) J,CSSP(J),SSSP(J),ZBT(J)
163 FORMAT("TRANSITION ",I3,/,14X,"COMPRESSIONAL SOUND SPEED",
   12F10.4,/,14X,"SHEAR SOUND SPEED",8X,2F10.4,/,
   214X,"DEPTH",20X,F10.4)
    IF(JPB.EQ.0.0) GO TO 309
    WRITE(1,306)
    WRITE(1,461) CCBS,SCBS
461 FORMAT("BASEMENT SPEEDS",2F10.4,/15X,2F10.4)
    GO TO 310
309 WRITE(1,307)
310 WRITE(1,165)
165 FORMAT("TO CHANGE PARAMETER TYPE 1 OTHERWISE 0")
    READ(IN,*) J
    IF(J.EQ.1) GO TO 5
    WRITE(1,110)
110 FORMAT("TYPE LINE PRINTER LU #")
    READ(IN,*) LP
    WRITE(LP,160) FRQ,NDZ,VP,S
    DO 926 J=1,NL
926 WRITE(LP,162) J,MU(J),ALAM(J),LDEP(J)
    IF(JPB.EQ.1) WRITE(LP,462) MU(NLAYS),ALAM(NLAYS)
    DO 927 J=1,KBT
927 WRITE(LP,163) J,CSSP(J),SSSP(J),ZBT(J)
    IF(JPB.EQ.0) GO TO 305
    WRITE(LP,306)
306 FORMAT("PROPAGATING BOTTOM")
    WRITE(LP,461) CCBS,SCBS
    GO TO 308
305 WRITE(LP,307)
307 FORMAT("RIGID BOTTOM")
C
C  CALCULATE VARIOUS CONSTANTS AND INTERATION PARAMETERS.
C
308 CONTINUE
    XLM=CMN/FRQ
    XLMS=SCMN/FRQ
    XLMS2=XLMS**2
    XLM2=XLM**2
    SD=REAL(XLM)*SS
    CMN2=CMN*CMN
    FRQ2=FRQ*FRQ
    SCMN2=SCMN*SCMN
    BETAS=FRQ2/SCMN2
    BETA=FRQ2/CMN2
    DZS=ZBS/(XLMS*FLOAT(NDZ))
    DZ=ZBS/(FLOAT(NDZ)*XLM)
    VP2=VP*VP
    SIGP=TPISQ*(BETA-VP2)*XLM2
    VS=VP
    IF(SCBS.EQ.CMPLX(0.0,0.0)) VS=CMPLX(0.0,0.0)
    VS2=VS*VS
    SIGS=TPISQ*(BETAS-VS2)*XLMS2
    IF(VS.EQ.CMPLX(0.0,0.0)) SIGS=CMPLX(0.0,0.0)
    NA=IFIX(S/(ZBS/FLOAT(NDZ))+1.)
    LM=NDZ/32
    LMAX=LM
    MAG=0.0
    IF(ISSW(10)) 440,441
440 NP=1
    GO TO 442
```

```
      GO TO 442
  441 NP=NDZ/32
  442 CONTINUE
      DO 640 J=1,NLAYS
  640 DN(J)=LDEP(J)*NDZ/(ZBS*32)
      WRITE(LP,641) (DN(J),J=1,NLAYS)
  641 FORMAT(6F10.5)
      WDEP=LDEP(1)
      IF(MU(2).EQ.0.0) WDEP=LDEP(2)
      COR=CMPLX(1.0,0.0)
      WRITE(LP,210) XLM,XLMS,SIGP,SIGS,DZ,DZS
  220 FORMAT("XLM=",2F10.5," XLMS=",2F10.5,/
     1"SIGP=",2F10.5," SIGS=",2F10.5,/
     2"DZ=",2F10.5," DZS=",2F10.5)
      KA=1
      CCL=CSSP(1)
      SCL=SSSP(1)
      ZL=0.0
      L=LMAX
      IF(ISSW(13)) 191,190
C
C  CALCULATE SOUND SPEED PROFILE NORMALIZED FUNCTION
C  STORE ON DISC FILES DSXQ AND DCXQ
C
  190 I=0
      L=0
      DO 50 J=1,NDZ
      I=I+1
      CS=CMPLX(0.0,0.0)
      SS=CMPLX(0.0,0.0)
      CXQ(I)=CMPLX(0.0,0.0)
      SXQ(I)=CMPLX(0.0,0.0)
      Z=FLOAT(J-1)*ZBS/FLOAT(NDZ)
      DO 40 K=KA,KBT
      IF(ZBT(K).GE.Z) GO TO 41
      ZL=ZBT(K)
      CCL=CSSP(K)
      SCL=SSSP(K)
   40 CONTINUE
      CCU=CCBS
      SCU=SCBS
      ZU=ZBS
      GO TO 42
   41 CONTINUE
      CCU=CSSP(K)
      SCU=SSSP(K)
      ZU=ZBT(K)
   42 CONTINUE
      CS=CCL+((CCU-CCL)/(ZU-ZL))*(Z-ZL)
      SS=SCL+((SCU-SCL)/(ZU-ZL))*(Z-ZL)
      W1(I)=CS
      W2(I)=SS
      CXQ(I)=TPISQ*FRQ2*(1./CMN2-(1./CS)**2)*XLM2
      SXQ(I)=TPISQ*FRQ2*(1./SCMN2-(1./SS)**2)*XLMS2
      XBF(I)=Z
      IF(SS.EQ.0.0) SXQ(I)=0.0
      KA=K
   52 IF(I.LT.32) GO TO 50
      CALL EXEC(15,1038,CXQ,128,IF12,L)
      CALL EXEC(15,1039,SXQ,128,IF13,L)
      I=0
      L=L+1
      IF(ISSW(1)) 51,50
   51 DO 55 JX=1,32,NP
   55 WRITE(LP,154) XBF(JX),W1(JX),W2(JX),CXQ(JX),SXQ(JX)
  154 FORMAT(F5.1,2X,2(F7.2,2X,F6.4,2X),4(F6.4,2X))
   50 CONTINUE
```

```
   60 CONTINUE
  191 CONTINUE
      I=1
      LMAX=L
      CS=CSSP(KBT)
      SS=SSSP(KBT)
      CXQ(1)=TPISQ*FRQ2*(1./CMN2-(1./CS)**2)*XLM2
      SXQ(1)=TPISQ*FRQ2*(1./SCMN2-(1./SS)**2)*XLMS2
      IF(SS.EQ.0.0) SXQ(1)=0.0
      IF(ISSW(1)) 503,502
  503 WRITE(LP,154) ZBS,CS,SS,CXQ(1),SXQ(1)
  502 CXQ(2)=TPISQ*FRQ2*(1./CMN2-(1./CCBS)**2)*XLM2
      SXQ(2)=TPISQ*FRQ2*(1./SCMN2-(1./SCBS)**2)*XLMS2
      B1=SXQ(2)
      IF(SCBS.EQ.CMPLX(0.0,0.0)) SXQ(2)=CMPLX(0.0,0.0)
      Z=ZBS+1.0
      IF(ISSW(1)) 505,504
  505 IF(JPB.EQ.1) WRITE(LP,154) Z,CCBS,SCBS,CXQ(2),SXQ(2)
  504 CALL EXEC(15,103B,CXQ,8,IF12,L)
      CALL EXEC(15,103B,SXQ,8,IF13,L)
C
C  RICCATI PROPAGATING BASEMENT INITIALIZATION
C
      IF(JPB.EQ.1) GO TO 17
      CFQBH=TPI*FRQ*CSQRT(1./CMN2-(1./CSSP(KBT))**2)*XLM
      SFQBH=TPI*FRQ*CSQRT(1./SCMN2-(1./SSSP(KBT))**2)*XLMS
      IF(SSSP(KBT).EQ.0.0) SFQBH=0.0
      CFQX(1)=CFQBH
      SFQX(1)=SFQBH
      CALL EXEC(15,103B,CFQX,4,IF14,L)
      CALL EXEC(15,103B,SFQX,4,IF15,L)
      CFQ=CFQBH
      SFQ=SFQBH
      NF=LM
      NS=1
      IF(ISSW(2)) 506,16
  506 WRITE(LP,152) ZBS,CFQX(1),SFQX(1)
      GO TO 16
C
C  RICCATI RIGID BOTTOM INITIALIZATION
C
   17 CFQBS=TPI*FRQ*CSQRT(1./CMN2-(1./CCBS)**2)*XLM
      SFQBS=TPI*FRQ*CSQRT(1./SCMN2-(1./SCBS)**2)*XLMS
      IF(SCBS.EQ.CMPLX(0.0,0.0)) SFQBS=CMPLX(0.0,0.0)
      CFQX(2)=CFQBS
      SFQX(2)=SFQBS
      NF=LM+1
      NS=32
      CFQ=CFQBS
      SFQ=SFQBS
      IF(ISSW(2)) 507,16
  507 WRITE(LP,152) Z,CFQX(2),SFQX(2)
   16 DO 475 J=1,32
  475 YP(J)=CMPLX(0.0,0.0)
C
C  RICCATI INTEGRATION---ADAMS-BASHFORTH METHOD
C  STORE ON DISC FILES DCFQX AND DSFQX
C
      CDFQ=CMPLX(0.0,0.0)
      SDFQ=CMPLX(0.0,0.0)
      SDFQP=CMPLX(0.0,0.0)
      CDFQP=CMPLX(0.0,0.0)
      SFQP=CMPLX(0.0,0.0)
      CFQP=CMPLX(0.0,0.0)
      IF(ISSW(13)) 192,193
  193 DO 60 J=1,NF
      LN=NF-I
```

```
      CALL EXEC(14,103B,CXQ,128,IF12,LN)
      CALL EXEC(14,103B,SXQ,128,IF13,LN)
      DO 60 I=NS,32
      NX=33-I
      CFQX(NX)=CMPLX(0.0,0.0)
      SFQX(NX)=CMPLX(0.0,0.0)
      CFQP=CFQ-DZ*CDFQ
      SFQP=SFQ-DZS*SDFQ
      CDFQP=CFQP**2-CXQ(NX)
      SDFQP=SFQP**2-SXQ(NX)
      CFQ=CFQ-.5*DZ*(CDFQ+CDFQP)
      SFQ=SFQ-.5*DZS*(SDFQ+SDFQP)
      CDFQ=CFQ**2-CXQ(NX)
      SDFQ=SFQ**2-SXQ(NX)
      CFQX(NX)=CFQ
      SFQX(NX)=SFQ
      Z=FLOAT(NX-1+LN*32)*ZBS/FLOAT(NDZ)
      IF(Z.LT.WDEP) SFQX(NX)=CMPLX(0.0,0.0)
      IF(NX.NEQ.1) GO TO 60
      CALL EXEC(15,103B,CFQX,128,IF14,LN)
      CALL EXEC(15,103B,SFQX,128,IF15,LN)
  474 CALL EXEC(15,103B,YP,128,IF9,LN)
      NS1=NS
      NS=1
  473 IF(ISSW(2)) 61,60
   61 DO 65 JY=NS1,32,NP
      JX=33-JY
      Z=FLOAT(JX-1+LN*32)*ZBS/FLOAT(NDZ)
   65 WRITE(LP,152) Z,CFQX(JX),SFQX(JX)
  152 FORMAT(F10.3,4(F10.6,2X))
   60 CONTINUE
  192 CONTINUE
C
C  INTEGRATE PARTICULAR SOLUTION USING LINEAR EQUATIONS
C  STORE ON DISC FILE DYP
C
      IF(ISSW(14)) 194,195
  195 CONTINUE
      PHII=CMPLX(0.0,0.0)
      PRII=CMPLX(0.0,0.0)
      DPHII=CMPLX(0.0,0.0)
      DPRII=CMPLX(0.0,0.0)
      DPHI=CMPLX(0.0,0.0)
      DPRI=CMPLX(0.0,0.0)
      PHI=CMPLX(0.0,0.0)
      PRI=CMPLX(0.0,0.0)
      LS=NA/32
      NX=LMAX+1
      IF(ISSW(3)) 501,500
  501 WRITE(LP,62)
   62 FORMAT(/"PARTICULAR SOLUTION")
  500 NA=(LS+1)*32-NA
      DO 13 J=1,NA
   13 YP(33-J)=CMPLX(0.0,0.0)
      PRII=-0.5*DZ*SD*XLM
      DPRI=SD*XLM
      LS=LS+1
      NA=NA+1
      DO 12 I=1,LS
      LT=LS-I
      CALL EXEC(14,103B,CFQX,128,IF14,LT)
      DO 12 J=NA,32
      NY=33-J
      PHI=PHII-DZ*DPHI
      PRI=PRII-DZ*DPRI
```

```
      DPRII=-SIGP*PHI+CFQX(NY)*PRI
      PHII=PHII-.5*DZ*(DPHI+DPHII)
      PRII=PRII-.5*DZ*(DPRI+DPRII)
      DPHI=-CFQX(NY)*PHII+PRII
      DPRI=-SIGP*PHII+CFQX(NY)*PRII
      IF(ISSW(3)) 176,175
  176 Z=(LT*32+NY-1)*ZBS/FLOAT(NDZ)
      WRITE(LP,152) Z,PHII
  175 YP(NY)=PHII
      IF(J.LT.32) GO TO 12
      CALL EXEC(15,103B,YP,128,IF9,LT)
      NA=1
   12 CONTINUE
  194 CONTINUE
C
C  PROPAGATING BASEMENT INITIALIZATION
C
      PS=CMPLX(0.0001,0.0)
      IF(VS.EQ.CMPLX(0.0,0.0)) PS=CMPLX(0.0,0.0)
  602 PP=CMPLX(0.0001,0.0)
      RP=CSQRT(CFQBS**2-SIGP)
      RS=CSQRT(SFQBS**2-SIGS)
      RP1=CFQBS-RP
      RS1=SFQBS-RS
      PH=PP/RP1
      PHDOT=-CFQBS*PH+PP
      IF(VS.EQ.CMPLX(0.0,0.0)) GO TO 601
      SI=PS/RS1
      SMPR=(XLM*(B1-SIGS)/(4.*PI*XLMS2)+PI*VS2*XLM)*JAY/VP
      NX=NLAYS-1
      IF(MU(NX).NEQ.0.0) GO TO 601
      NX=NLAYS-1
      IF(MU(NX).NEQ.0.0) GO TO 601
      PHDOT=-SMPR*SI
      PH=PHDOT/(RP1-CFQBS)
      PP=PH*RP1
  601 CONTINUE
      IF(JPB.EQ.0) GOTO 814
      SIDOT=-SFQBS*SI+PS
      WRITE(LP,990)
  990 FORMAT(//"PROPAGATING BASEMENT"/)
  210 FORMAT(/4F15.6)
  634 FORMAT(//4F12.6,//)
      PHB=PH
      SIB=SI
      PPB=PP
      PSB=PS
      SCL=PHDOT
      CCU=SIDOT
      NX=1
      WRITE(LP,210) PH,PHDOT,SI,SIDOT
  814 IF(JPB.EQ.1) GO TO 201
C
C  RIGID BOTTOM INITIALIZATION
C
      PS=CMPLX(0.0,0.0)
      SI=CMPLX(0.0,0.0)
      PP=CFQBS
      PH=CMPLX(1.0,0.0)
      PHDOT=-CFQBH*PH+PP
      SIDOT=-SFQBH*SI+PS
      WRITE(LP,991)
  991 FORMAT(//"RIGID BOTTOM"//)
      WRITE(LP,210) PH,PHDOT
      PHB=PH
      SCL=PHDOT
```

```
      SIB=SI
      CCU=SIDOT
      JK=NLAYS
      NF=0
      NX=1
      CALL EXEC(14,103B,CFQX,4,IF14,LMAX)
      CALL EXEC(14,103B,SFQX,4,IF15,LMAX)
      SF3=SFQX(1)
      CF3=CFQX(1)
      JK=JK-1
      IF(JK.EQ.1) GO TO 303
      GO TO 202
C
C  COMPRESSIONAL POTENTIAL EXCITATION
C  USE INTERFACE EQUATIONS TO OBTAIN POTENTIAL ABOVE INTERFACE
C  INTEGRATE MAGNITUDE AND PHASE EQUATIONS UPWARD
C
  201 JK=NLAYS
      SI=CMPLX(0.0,0.0)
      SIDOT=CMPLX(0.0,0.0)
      PS=CMPLX(0.0,0.0)
      NF=32
      CALL EXEC(14,103B,CXQ,8,IF12,LMAX)
      CALL EXEC(14,103B,SXQ,8,IF13,LMAX)
      SF1=SXQ(1)
      SF2=SXQ(2)
      CF1=CXQ(1)
      CF2=CXQ(2)
      CALL EXEC(14,103B,CFQX,4,IF14,LMAX)
      CALL EXEC(14,103B,SFQX,4,IF15,LMAX)
      SF3=SFQX(1)
      CF3=CFQX(1)
      L=LMAX-1
      NS=1
      MAG=0.0
      WRITE(LP,707)
  707 FORMAT(/"COMPRESSIONAL EXCITATION-PH,PHDOT,SI,SIDOT"/)
  200 JK=JK-1
      IF(JK.EQ.1) GO TO 300
      ML=JK
      MM=JK+1
      CALL SSBC(ML,VS,VP,SF1,SF2,CF1,CF2,XLM,MU,ALAM,SIGS,SIGP,
     1PH,PHDOT,SI,SIDOT,LP,MM,XLMS)
  202 CONTINUE
      CALL RNSV(SF3,CF3,LP,SIGS,SIGP,PH,PHDOT,SI,SIDOT,TH,THDOT,
     1M,MDOT,N,NDOT,GA,GADOT)
      IF(MU(ML).NEQ.0.0) GO TO 27
      GADOT=CMPLX(0.0,0.0)
      NDOT=CMPLX(0.0,0.0)
      GA=CMPLX(0.0,0.0)
      SIGS=CMPLX(0.0,0.0)
      SI=CMPLX(0.0,0.0)
      SIDOT=CMPLX(0.0,0.0)
      N=CMPLX(0.0,0.0)
   27 CONTINUE
      IF(JK.NEQ.(NLAYS-1)) GO TO 418
      W1(1)=TH
      W2(1)=M-7
      W3(1)=GA
      W4(1)=N-7
      D1=PH
      D2=SI
      D3=PHDOT
      D4=SIDOT
      CALL EXEC(15,103B,W1,4,IF1,LMAX)
      CALL EXEC(15,103B,W2,4,IF2,LMAX)
```

```
      CALL EXEC(15,103B,W3,4,IF3,LMAX)
      CALL EXEC(15,103B,W4,4,IF4,LMAX)
  418 CONTINUE
      FN=DN(JK)-DN(JK-1)
      CALL SCSPH(SFOX,CFOX,LP,NS,NF,ZBS,DZ,NDZ,SIGP,SIGS,W1,W2,W3,
     1TH,THDOT,GA,GADOT,M,MDOT,N,NDOT,PH,PHDOT,PP,PPDOT,SI,SIDOT,
     2PS,PSDOT,DZS,FN,L,IF1,IF2,IF3,IF4,IF14,IF15,MAG)
      L1=L+1
      IF(NF.NE.32) L1=L
      CALL EXEC(14,103B,CXQ,128,IF12,L1)
      CALL EXEC(14,103B,SXQ,128,IF13,L1)
      CALL EXEC(14,103B,CFOX,128,IF14,L1)
      CALL EXEC(14,103B,SFOX,128,IF15,L1)
      SF1=SXQ(32-NS+1)
      SF2=SXQ(32-NS+2)
      CF1=CXQ(32-NS+1)
      CF2=CXQ(32-NS+2)
      SF3=SFOX(32-NS+1)
      CF3=CFOX(32-NS+1)
      IF(NS.NE.1) NS=NS+1
   66 GO TO 200
  300 CONTINUE
      IF(NLAYS.EQ.2) GO TO 301
      IF(MU(NLAYS).EQ.0.0) SI=CMPLX(0.0,0.0)
C     WRITE(LP,210) MAG
      IF(MU(NLAYS).EQ.0.0)  GO TO 702
C
C  SHEAR POTENTIAL EXCITATION
C  INTEGRATE MAGNITUDE AND PHASE EQUATIONS UPWARD
C
      WRITE(LP,706)
  706 FORMAT(/"SHEAR EXCITATION-PH,PHDOT,SI,SIDOT"/)
      SMPR=((SF2-SIGS)*XLM/(4.*PI*XLMS2)+PI*VS2*XLM)*(JAY/VP)
      COR=PHDOT+SMPR*SI
C     WRITE(LP,210) PH,PHDOT,SI,SIDOT,SMPR,COR,SF2
  701 JK=NLAYS
      PH=CMPLX(0.0,0.0)
      PHDOT=CMPLX(0.0,0.0)
      PP=CMPLX(0.0,0.0)
      SI=SI0
      SIDOT=CCU
      PS=PS0
      NF=32
      CALL EXEC(14,103B,CXQ,8,IF12,LMAX)
      CALL EXEC(14,103B,SXQ,8,IF13,LMAX)
      SF1=SXQ(1)
      SF2=SXQ(2)
      CF1=CXQ(1)
      CF2=CXQ(2)
      CALL EXEC(14,103B,CFOX,4,IF14,LMAX)
      CALL EXEC(14,103B,SFOX,4,IF15,LMAX)
      SF3=SFOX(1)
      CF3=CFOX(1)
      L=LMAX-1
      NS=1
      MAGS=0.0
      IF(JPR.EQ.0) GO TO 703
  700 JK=JK-1
      IF(JK.EQ.1) GO TO 702
      ML=JK
      MM=JK+1
      CALL SSBC(ML,VS,VP,SF1,SF2,CF1,CF2,XLM,MU,ALAM,SIGS,SIGP,
     1PH,PHDOT,SI,SIDOT,LP,MM,XLMS)
      GO TO 710
  703 CONTINUE
```

```
      IF(JK.EQ.1) GO TO 702
      ML=JK
  710 CONTINUE
      CALL RNSV(SF3,CF3,LP,SIGS,SIGP,PH,PHDOT,SI,SIDOT,TH,THDOT,
     1M,MDOT,N,NDOT,GA,GADOT)
      IF(MU(ML).NE.0.0) GO TO 29
      GADOT=CMPLX(0.0,0.0)
      NDOT=CMPLX(0.0,0.0)
      GA=CMPLX(0.0,0.0)
      SIGS=CMPLX(0.0,0.0)
      SI=CMPLX(0.0,0.0)
      SIDOT=CMPLX(0.0,0.0)
      N=CMPLX(0.0,0.0)
   29 CONTINUE
      IF(JK.NEQ.(NLAYS-1)) GO TO 704
      W5(1)=TH
      W6(1)=M-7
      W7(1)=GA
      W8(1)=N-7
      D5=PH
      D6=SI
      D7=PHDOT
      D8=SIDOT
      CALL EXEC(15,103B,W5,4,IF5,LMAX)
      CALL EXEC(15,103B,W6,4,IF6,LMAX)
      CALL EXEC(15,103B,W7,4,IF7,LMAX)
      CALL EXEC(15,103B,W8,4,IF8,LMAX)
  704 CONTINUE
      FN=DN(JK)-DN(JK-1)
      CALL SCSPH(SFQX,CFQX,LP,NS,NF,ZBS,DZ,NDZ,SIGP,SIGS,W5,W6,W7,
     1TH,THDOT,GA,GADOT,M,MDOT,N,NDOT,PH,PHDOT,PP,PPDOT,SI,SIDOT,
     2PS,PSDOT,DZS,FN,L,IF5,IF6,IF7,IF8,IF14,IF15,MAGS)
      L1=L+1
      IF(NF.NE.32) L1=L
      CALL EXEC(14,103B,CXQ,128,IF12,L1)
      CALL EXEC(14,103B,SXQ,128,IF13,L1)
      CALL EXEC(14,103B,CFQX,128,IF14,L1)
      CALL EXEC(14,103B,SFQX,128,IF15,L1)
      SF1=SXQ(32-NS+1)
      SF2=SXQ(32-NS+2)
      CF1=CXQ(32-NS+1)
      CF2=CXQ(32-NS+2)
      SF3=SFQX(32-NS+1)
      CF3=CFQX(32-NS+1)
      IF(NS.NE.1) NS=NS+1
      GO TO 700
  702 CONTINUE
C
C  COMBINE CALCULATED SOLUTIONS TO SATISFY TANGENTIAL
C  STRESS AT WATER-SEDIMENT = ZERO CONDITION
C
C     WRITE(LP,210) PH,PHDOT,SI,SIDOT,SMPR,COR
      COR=-COR/(PHDOT+SMPR*SI)*EXP(MAG-MAGS)
      IF(MU(NLAYS-1).EQ.0.0) COR=CMPLX(1.0,0.0)
      IF(MU(NLAYS).EQ.0.0) COR=CMPLX(0.0,0.0)
      WRITE(LP,212) COR
  212 FORMAT(2F15.8)
      MAG=AMAX1(MAG,MAGS)
      MAG=MAG-7
      N1=DN(NL)+2-DN(1)
      JA=DN(1)
      AZ=DN(1)-JA
      IF(AZ.EQ.0.0) N1=N1-1
      AZ=1.0-AZ
      L=LMAX
```

```
      NF=32
      DO 750 I=1,N1
      CALL EXEC(14,103B,W1,128,IF1,L)
      CALL EXEC(14,103B,W2,128,IF2,L)
      CALL EXEC(14,103B,W3,128,IF3,L)
      CALL EXEC(14,103B,W4,128,IF4,L)
      CALL EXEC(14,103B,W5,128,IF5,L)
      CALL EXEC(14,103B,W6,128,IF6,L)
      CALL EXEC(14,103B,W7,128,IF7,L)
      CALL EXEC(14,103B,W8,128,IF8,L)
      CALL EXEC(14,103B,CFQX,128,IF14,L)
      CALL EXEC(14,103B,SFQX,128,IF15,L)
      DO 705 J=NS,NF
      NX=33-J
      XBF(NX)=FLOAT(NX-1+L*32)*ZBS/FLOAT(NDZ)
      B1=CCOS(W1(NX))
      B2=CSIN(W1(NX))
      B3=CCOS(W3(NX))
      IF(SFQX(NX).EQ.CMPLX(0.0,0.0)) B3=CMPLX(0.0,0.0)
      B4=CSIN(W3(NX))
      YBF(NX)=B1*CEXP(W2(NX)-MAG)
      ZBF(NX)=B3*CEXP(W4(NX)-MAG)
      W3(NX)=(-CFQX(NX)*B1+B2)*CEXP(W2(NX)-MAG)
      W4(NX)=(-SFQX(NX)*B3+B4)*CEXP(W4(NX)-MAG)
      IF(MU(NLAYS).EQ.0.0) GO TO 755
      B5=CCOS(W5(NX))
      B6=CSIN(W5(NX))
      B7=CCOS(W7(NX))
      IF(SFQX(NX).EQ.CMPLX(0.0,0.0)) B7=CMPLX(0.0,0.0)
      B8=CSIN(W7(NX))
      YBF(NX)=YBF(NX)+COR*B5*CEXP(W6(NX)-MAG)
      ZBF(NX)=ZBF(NX)+COR*B7*CEXP(W8(NX)-MAG)
      W3(NX)=W3(NX)+(-CFQX(NX)*B5+B6)*COR*CEXP(W6(NX)-MAG)
      W4(NX)=W4(NX)+(-SFQX(NX)*B7+B8)*COR*CEXP(W8(NX)-MAG)
  755 CONTINUE
      IF(L.NEQ.LMAX) GO TO 756
      YBF(1)=(D1+COR*D5)*EXP(-MAG-7)
      D1=YBF(1)
      ZBF(1)=(D2+COR*D6)*EXP(-MAG-7)
      D2=ZBF(1)
      W3(1)=(D3+COR*D7)*EXP(-MAG-7)
      D3=W3(1)
      W4(1)=(D4+COR*D8)*EXP(-MAG-7)
      D4=W4(1)
  756 CONTINUE
      IF(NX.NE.1) GO TO 705
      CALL EXEC(15,103B,YBF,128,IF10,L)
      CALL EXEC(15,103B,ZBF,128,IF11,L)
      L=L-1
  751 IF(ISSW(6)) 760,790
  760 DO 709 JY=NS,NF,NP
      JX=33-JY
  709 WRITE(LP,715) XBF(JX),YBF(JX),ZBF(JX)
  715 FORMAT(F5.1,4F15.8)
  790 NS=1
      NF=32
      IF(I.EQ.(N1-1).AND.AZ.NE.1.0) NF=AZ*32
  705 CONTINUE
      IF(NF.LE.0) GO TO 752
  750 CONTINUE
  752 CONTINUE
      IF(ISSW(6)) 509,508
  509 IF(NF.LT.32) WRITE(LP,715)(XBF(33-JY),YBF(33-JY),
     1ZBF(33-JY),JY=NS,NF,NP)
  508 PH=YBF(NX)
      SI=ZBF(NX)
```

```
      SI=W2(NX)
      PHDOT=W3(NX)
      SIDOT=W4(NX)
      Y2=MU(2)*(PHDOT+SMPR*SI)
C     WRITE(LP,210) PH,PHDOT,SI,SIDOT,Y2,SMPR
      NS=NF+2
      NF=32
      NX=NX-1
  708 FORMAT(F4.0,2X,8F8.4)
      IF(AZ.NE.0.0) CALL EXEC(15,103B,YBF,128,IF10,L)
      IF(AZ.NE.0.0) CALL EXEC(15,103B,ZBF,128,IF11,L)
      IF(NS.EQ.34) NS=2
C
C  INTEGRATE MAGNITUDE AND PHASE EQUATIONS IN THE WATER COLUMN
C
  301 CONTINUE
      IF(NX.EQ.0) NX=1
      L=DN(1)-1
      NS=NS-1
      IF(NS.NE.1) L=L+1
      IF(NS.NE.1) NX=NX+1
      N1=DN(1)
      IF(NLAYS.NEQ.2) GO TO 721
      NS=1
      NF=32
      NX=1
      N1=LMAX
      L=LMAX-1
      SI=SIB
      SIDOT=CCU
  721 CALL EXEC(14,103B,CXQ,128,IF12,N1)
      CALL EXEC(14,103B,SXQ,128,IF13,N1)
      CALL EXEC(14,103B,CFQX,128,IF14,N1)
      CALL EXEC(14,103B,SFQX,128,IF15,N1)
  720 SF2=SXQ(NX+1)
      CF1=CXQ(NX)
      CF2=CXQ(NX+1)
      CALL SUBC(YS,YP,SF2,CF1,CF2,XLM,MU,ALAM,LP,SIGP,
     1SIGS,PH,PHDOT,SI,SIDOT,XLMS)
  303 CONTINUE
      SF3=SFQX(NX)
      CF3=CFQX(NX)
      SI=CMPLX(0.0,0.0)
      SIDOT=CMPLX(0.0,0.0)
      PS=CMPLX(0.0,0.0)
      PSDOT=CMPLX(0.0,0.0)
      IF(NLAYS.NEQ.2) GO TO 453
      D1=PH*EXP(-7.0)
      D2=SI
      D3=PHDOT*EXP(-7.0)
      D4=SIDOT
  453 CALL RNSV(SF3,CF3,LP,SIGS,SIGP,PH,PHDOT,SI,SIDOT,TH,THDOT,
     1M,MDOT,N,NDOT,GA,GADOT)
      GADOT=CMPLX(0.0,0.0)
      GA=CMPLX(0.0,0.0)
  454 CONTINUE
      IF(NLAYS.GT.2) M=M-MAG+7
      IF(NLAYS.GT.2) N=N-MAG+7
      MAGB=0.0
      MAGB=MAG+7
      JA=DN(1)
      FN=JA
      SIGS=CMPLX(0.0,0.0)
      L1=L
      CALL SCSPH(SFQX,CFQX,LP,NS,NF,ZBS,DZ,NDZ,SIGP,SIGS,W1,W2,W3,
     1TH,THDOT,GA,GADOT,M,MDOT,N,NDOT,PH,PHDOT,PP,PPDOT,SI,SIDOT,
     2ZS,PSDOT,DZS,FN,L,IF1,IF2,IF3,IF4,IF14,IF15,MAG)
```

```
      SI=CMPLX(0.0,0.0)
      CALL EXEC(14,103B,YP,128,IF9,0)
      CCL=YP(1)/(CCOS(W1(1))*CEXP(W2(1)))
      WRITE(LP,212) CCL
  211 FORMAT(/F15.2,2X,F15.2)
      WRITE(LP,800)
  800 FORMAT(/"TOTAL SOLUTION-PH,SI"/)
C
C  OBTAIN TOTAL SOLUTION
C
      DO 460 I=1,LMAX
      N1=I-1
      CALL EXEC(14,103B,YP,128,IF9,N1)
      CALL EXEC(14,103B,YBF,128,IF10,N1)
      CALL EXEC(14,103B,ZBF,128,IF11,N1)
      DO 70 J=1,32
      YBF(J)=YP(J)-CCL*YBF(J)
      XBF(J)=FLOAT(J-1+N1*32)*ZBS/FLOAT(NDZ)
      ZBF(J)=-CCL*ZBF(J)
      YBF(J)=YBF(J)*NDZ/ZBS
      ZBF(J)=ZBF(J)*NDZ/ZBS
  155 FORMAT(F6.2,4F15.5)
   70 CONTINUE
      CALL EXEC(15,103B,YBF,128,IF10,N1)
      CALL EXEC(15,103B,ZBF,128,IF11,N1)
      NM=DN(1)-1.0
      NN=DN(1)+1.0
      NP=NDZ/32
      IF(I.GE.NM.AND.I.LE.NN) NP=1
      IF(ISSW(7)) 460,178
  178 DO 711 JX=1,32,NP
  711 WRITE(LP,155) XBF(JX),YBF(JX),ZBF(JX)
  460 CONTINUE
      IF(JP8.EQ.0) GO TO 1000
C
C  OBTAIN BASEMENT SOLUTION
C
      LB=LMAX
      IF(ISSW(12)) 314,315
  315 WRITE(LP,347)
  347 FORMAT(//"VALUES AT BASEMENT INTERFACE")
      PH=-CCL*NDZ/ZBS*D1
      PHDOT=-CCL*NDZ/ZBS*D3
      SI=-CCL*NDZ/ZBS*D2
      SIDOT=-CCL*NDZ/ZBS*D4
      WRITE(LP,210) PH,PHDOT,SI,SIDOT
      IF(NLAYS.EQ.2) MAGB=7.0
      WRITE(LP,990)
      PH=PHB*(NDZ/ZBS)*(-CCL)*EXP(-MAGB)
      PHDOT=SCL*(NDZ/ZBS)*(-CCL)*EXP(-MAGB)
      IF(MU(NLAYS-1).EQ.0.0) COR=CMPLX(1.0,0.0)
      SI=SIB*NDZ/ZBS*(-CCL)*EXP(-MAGB)*COR
      SIDOT=CCU*NDZ/ZBS*(-CCL)*EXP(-MAGB)*COR
      WRITE(LP,210) PH,PHDOT,SI,SIDOT
      YBF(1)=PH
      W3(1)=PHDOT
      ZBF(1)=SI
      W4(1)=SIDOT
      SWEQ=SI
      WEQ=PH
  812 CONTINUE
      SMPR=-JAY*2.0*PI*VS*XLM
      LB=DN(NLAYS)
      XBF(1)=ZBS
      NS=0
```

```
      DO 26 J=1,31
      N1=J+1
      XBF(N1)=FLOAT(N1-1+L*32)*ZBS/FLOAT(NDZ)
      YBF(N1)=PH*CEXP(-RP*FLOAT(J+NS)*DZ)
      ZBF(N1)=SI*CEXP(-RS*FLOAT(J+NS)*DZS)
  571 FORMAT(I4,2X,8F8.5)
   26 CONTINUE
      IF(ISSW(8)) 415,416
  416 DO 415 J=1,32,NP
      WRITE(LP,155) XBF(J),YBF(J),ZBF(J)
  415 CONTINUE
      CALL EXEC(15,103B,YBF,128,IF10,L)
      CALL EXEC(15,103B,ZBF,128,IF11,L)
 1000 CONTINUE
      PH=YBF(32)
      SI=ZBF(32)
      YBF(1)=PH*CEXP(-RP*DZ)
      ZBF(1)=SI*CEXP(-RS*DZS)
      XBF(1)=FLOAT(N1+L*32)*ZBS/FLOAT(NDZ)
      NS=1
  313 CONTINUE
  314 CONTINUE
      SXQ(1)=CMPLX((FLOAT(NLAYS)),0.0)
      SXQ(2)=CMPLX((FLOAT(NDZ)),0.0)
      SXQ(3)=CMPLX(ZBS,0.0)
      SXQ(4)=HEQ
      SXQ(5)=SWEQ
      DO 120 J=1,NLAYS
      SXQ(J+5)=CMPLX(LDEP(J),0.0)
  120 CONTINUE
      CALL EXEC(15,103B,SXQ,128,IF8,0)
      END
C
C  THIS SUBROUTINE CALCULATES POTENTIALS ABOVE WATER-SEDIMENT
C  INTERFACE GIVEN POTENTIALS BELOW INTERFACE
C
      SUBROUTINE SWBC(VS,VP,SF2,CF1,CF2,XLM,MU,ALAM,LP,SIGP,
     1SIGS,PH,PHDOT,SI,SIDOT,XLMS)
      COMPLEX X1,X3,VS,VP,PH,PHDOT,SI,SIDOT,SIGP,SIGS
      COMPLEX XLM,CF1,CF2,SF2,VP2,VS2,XLM2,XLMS2,JAY
      COMPLEX A,M,E,G,F,H,C,D,XLMS
      COMPLEX Y1,Y3,Y4,SMPR
      REAL MU
      DIMENSION MU(1),ALAM(1)
      DATA JAY,PI/(0.0,1.0),3.1415926/
      PI2=PI**2
      VP2=VP**2
      VS2=VS**2
      XLM2=XLM**2
      XLMS2=XLMS*XLMS
      Y1=CMPLX(0.0,0.0)
      Y4=CMPLX(0.0,0.0)
      Y3=CMPLX(0.0,0.0)
      A=CMPLX(0.0,0.0)
      M=CMPLX(0.0,0.0)
      E=CMPLX(0.0,0.0)
      G=CMPLX(0.0,0.0)
      SMPR=CMPLX(0.0,0.0)
      F=CMPLX(0.0,0.0)
      H=CMPLX(0.0,0.0)
      C=CMPLX(0.0,0.0)
      D=CMPLX(0.0,0.0)
      A=JAY*2.*PI*VS*XLM
      M=JAY*2.*PI*VP*XLMS
      E=(ALAM(1)+2.*MU(1))/(-4.*PI2*VP2*XLM2)
```

```
      [illegible]
      F=(2.*MU(1))/(-4.*PI2*VP2*XLM*XLMS)
      H=(2.*MU(2))/(-4.*PI2*VP2*XLM*XLMS)
      C=VS/(2.*VP)
      D=XLM/(JAY*4.*VP*PI*XLMS2)
      X1=CMPLX(0.0,0.0)
      X3=CMPLX(0.0,0.0)
      X1=ALAM(2)+G*(CF2-SIGP)
      X3=ALAM(1)+E*(CF1-SIGP)
      SMPR=A*C-D*(SF2-SIGS)
      Y1=PHDOT+A*SI
      Y3=MU(2)*(PHDOT+SMPR*SI)
      Y4=X1*PH+A*H*SIDOT
      WRITE(LP,2)
    2 FORMAT(/"PH,PHDOT,SI,SIDOT-BELOW INTERFACE #1"/)
      WRITE(LP,1) PH,PHDOT,SI,SIDOT
C     WRITE(LP,1) Y1,Y3,Y4,SMPR,SF2
    1 FORMAT(4F15.7/)
C
C
C
      PHDOT=PHDOT+A*SI
      PH=(X1*PH+A*H*SIDOT)/X3
      SI=CMPLX(0.0,0.0)
      SIDOT=CMPLX(0.0,0.0)
C
C
C
      WRITE(LP,346)
  346 FORMAT(/"PH,PHDOT-ABOVE INTERFACE #1"/)
      Y1=PHDOT+A*SI
      Y4=X3*PH+A*F*SIDOT
      WRITE(LP,1) PH,PHDOT
C     WRITE(LP,1) Y1,Y4
      RETURN
      END
C
C  THIS SUBROUTINE CALCULATES POTENTIALS ABOVE INTERFACE
C  GIVEN POTENTIALS BELOW INTERFACE FOR INTERFACE SEPARATING
C  TWO ELASTIC LAYERS.
C
      SUBROUTINE SSPC(ML,VS,VP,SF1,SF2,CF1,CF2,XLM,MU,ALAM,SIGS,SI
     1PH,PHDOT,SI,SIDOT,LP,MM,XLMS)
      COMPLEX X7,VS,VP,PH,PHDOT,SI,SIDOT,A,M,E,G,F,H,C,D,XLMS,XLMS
      COMPLEX VS2,VP2,SIGP,SIGS,XLM,SF1,SF2,Y1,Y2,Y3,Y4,JAY,CF1,CF
      COMPLEX XLM2
      DIMENSION MU(1),ALAM(1)
      REAL MU
      DATA JAY,PI/(0.0,1.0),3.1415926/
      A=CMPLX(0.0,0.0)
      M=CMPLX(0.0,0.0)
      E=CMPLX(0.0,0.0)
      G=CMPLX(0.0,0.0)
      F=CMPLX(0.0,0.0)
      H=CMPLX(0.0,0.0)
      X7=CMPLX(0.0,0.0)
      C=CMPLX(0.0,0.0)
      D=CMPLX(0.0,0.0)
      Y1=CMPLX(0.0,0.0)
      Y2=CMPLX(0.0,0.0)
      Y3=CMPLX(0.0,0.0)
      Y4=CMPLX(0.0,0.0)
      PI2=PI**2
      VP2=VP*VP
      VS2=VS*VS
      XLM2=XLM*XLM
      XLMS2=XLMS*XLMS
```

```
C
C
      A=JAY*2.*PI*VS*XLM
      M=JAY*2.*PI*VP*XLMS
      E=(ALAM(ML)+2.*MU(ML))/(-4.*PI2*VP2*XLM2)
      G=(ALAM(MM)+2.*MU(MM))/(-4.*PI2*VP2*XLM2)
      F=(2.*MU(ML))/(-4.*PI2*VP2*XLM*XLMS)
      H=(2.*MU(MM))/(-4.*PI2*VP2*XLM*XLMS)
      C=VS/(2.*VP)
      D=XLM/(JAY*4.*VP*PI*XLMS2)
      Y1=PHDOT+A*SI
      Y2=-SIDOT+M*PH
      Y3=MU(MM)*(PHDOT+A*C*SI-D*(SF2-SIGS)*SI)
      Y4=(ALAM(MM)+G*(CF2-SIGP))*PH+A*H*SIDOT
      WRITE(LP,2) ML
    2 FORMAT(/"PH,PHDOT,SI,SIDOT-BELOW INTERFACE #",I3)
      WRITE(LP,1) PH,PHDOT,SI,SIDOT
C     WRITE(LP,1) Y1,Y2,Y3,Y4
    1 FORMAT(/4F15.7)
C
C
      X7=(ALAM(MM)+G*(CF2-SIGP)+A*F*M)*PH
      X7=X7+(H-F)*A*SIDOT
      X7=X7/(ALAM(ML)+E*(CF1-SIGP)+A*F*M)
      SIDOT=SIDOT-M*(PH-X7)
      IF(SIGS.EQ.CMPLX(0.0,0.0)) SIDOT=CMPLX(0.0,0.0)
      PH=X7
      X7=CMPLX(0.0,0.0)
C
C
      X7=-MU(ML)*A+MU(MM)*(-D*(SF2-SIGS)+A*C)
      X7=X7*SI+PHDOT*(MU(MM)-MU(ML))
      X7=X7/(MU(ML)*(-A+A*C-D*(SF1-SIGS)))
      IF(SIGS.EQ.CMPLX(0.0,0.0)) X7=CMPLX(0.0,0.0)
      IF(MU(ML).EQ.0.0) X7=CMPLX(0.0,0.0)
      PHDOT=PHDOT+A*(SI-X7)
      SI=X7
      IF(MU(ML).EQ.0.0) SIDOT=CMPLX(0.0,0.0)
      IF(MU(ML).EQ.0.0) SI=CMPLX(0.0,0.0)
C
C
      WRITE(LP,913) ML
  913 FORMAT(/"PH,PHDOT,SI,SIDOT-ABOVE INTERFACE # ,",I3)
      WRITE(LP,1) PH,PHDOT,SI,SIDOT
      Y1=CMPLX(0.0,0.0)
      Y2=CMPLX(0.0,0.0)
      Y3=CMPLX(0.0,0.0)
      Y4=CMPLX(0.0,0.0)
      Y1=PHDOT+A*SI
      Y2=-SIDOT+M*PH
      Y3=MU(ML)*(PHDOT+A*C*SI-(SF1-SIGS)*D*SI)
      Y4=(ALAM(ML)+E*(CF1-SIGP))*PH+A*F*SIDOT
C     WRITE(LP,1) Y1,Y2,Y3,Y4
      RETURN
      END
C
C THIS SUBROUTINE TRANSFORMS FROM THE LINEAR TO THE MAGNITUDE AND
C PHASE PLANE
C
      SUBROUTINE RNSY(SF,CF,LP,SIGS,SIGP,PH,PHDOT,SI,SIDOT,TH,THDO
     1M,MDOT,N,NDOT,GA,GADOT)
      COMPLEX PH,SI,PHDOT,SIDOT,PP,PS,PSDOT,PPDOT,GA,M,N,TH,SIGP,S
      COMPLEX MDOT,NDOT,THDOT,GADOT,SF,CF,A,CSZ1,SSZ1,CSZ2,SSZ2
      DIMENSION X(10)
      DATA PI/3.1415926/
      TH=CMPLX(0.0,0.0)
```

```
      M=CMPLX(0.0,0.0)
      GA=CMPLX(0.0,0.0)
      N=CMPLX(0.0,0.0)
      THDOT=CMPLX(0.0,0.0)
      GADOT=CMPLX(0.0,0.0)
      NDOT=CMPLX(0.0,0.0)
      PP=CMPLX(0.0,0.0)
      MDOT=CMPLX(0.0,0.0)
      PPDOT=CMPLX(0.0,0.0)
      PS=CMPLX(0.0,0.0)
      PSDOT=CMPLX(0.0,0.0)
      PP=PHDOT+PH*CF
      PPDOT=-SIGP*PH+CF*PP
      PS=SIDOT+SI*SF
      PSDOT=-SIGS*SI+SF*PS
      E=0.0
      IF(AIMAG(PP).EQ.0.0.AND.AIMAG(PH).EQ.0.0) GO TO 5
      GO TO 6
    5 IF(REAL(PP).LT.0.0.AND.REAL(PH).LT.0.0) E=3.141592654
    6 A=PH**2+PP**2
      M=0.5*CLOG(A)+(0.0,1.0)*E
      D=1.0
      IF(AIMAG(A).EQ.0.0.AND.REAL(A).LT.0.0) GO TO 9
      GO TO 10
    9 IF(AIMAG(PP).LT.0.0.AND.AIMAG(PH).LT.0.0) D=-1.0
   10 D=D*AIMAG(M)
      M=CMPLX(REAL(M),D)
      IF(REAL(PH).LT.0.0.AND.AIMAG(PH).GT.0.0)
     1M=M+(0.0,3.1415926)
      E=0.0
      IF(AIMAG(SI).EQ.0.0.AND.AIMAG(PS).EQ.0.0) GO TO 7
      GO TO 8
    7 IF(REAL(PS).LT.0.0.AND.REAL(SI).LT.0.0) E=3.141592654
    8 A=PS**2+SI**2
      N=0.5*CLOG(A)+(0.0,1.0)*E
      D=1.0
      IF(AIMAG(A).EQ.0.0.AND.REAL(A).LT.0.0) GO TO 11
      GO TO 12
   11 IF(AIMAG(PS).LT.0.0.AND.AIMAG(SI).LT.0.0) D=-1.0
   12 D=D*AIMAG(N)
      N=CMPLX(REAL(N),D)
      IF(REAL(SI).LT.0.0.AND.AIMAG(SI).GT.0.0)
     1N=N+(0.0,3.1415926)
      CSZ1=SI**2+PS**2
      IF(CSZ1.EQ.CMPLX(0.0,0.0)) N=CMPLX(0.0,0.0)
      TH=(0.,-0.5)*CLOG(((0.,1.)-PP/PH)/((0.,1.)+PP/PH))
      CSZ1=PS/SI
      IF(SI.EQ.CMPLX(0.0,0.0)) CSZ1=CMPLX(0.0,0.0)
      GA=(0.,-0.5)*CLOG(((0.,1.)-CSZ1)/((0.,1.)+CSZ1))
      CSZ1=CCOS(TH)
      SSZ1=CSIN(TH)
      CSZ2=CCOS(GA)
      SSZ2=CSIN(GA)
      MDOT=CF*(SSZ1*SSZ1-CSZ1*CSZ1)+(1.-SIGP)*SSZ1*CSZ1
      NDOT=SF*(SSZ2*SSZ2-CSZ2*CSZ2)+(1.-SIGS)*SSZ2*CSZ2
      THDOT=2.*CF*CSZ1*SSZ1-SIGP*CSZ1*CSZ1-SSZ1*SSZ1
      GADOT=2.*SF*CSZ2*SSZ2-SIGS*CSZ2*CSZ2-SSZ2*SSZ2
      IF(ISSW(4)) 4,3
    4 WRITE(LP,2)
    2 FORMAT(/"TRANSFORM TO PHASE PLANE-"
     1"TH,M,GA,N,THDOT,MDOT,GADOT,NDOT"/)
      WRITE(LP,1) TH,M,GA,N,THDOT,MDOT,GADOT,NDOT
      WRITE(LP,1) PP,PPDOT,PS,PSDOT
    3 X(1)=ATAN2(AIMAG(PH),REAL(PH))
      X(2)=ATAN2(AIMAG(PP),REAL(PP))
      X(3)=ATAN2(AIMAG(SI),REAL(SI))
```

```
      X(3)=ATAN2(AIMAG(SI),REAL(SI))
      X(4)=ATAN2(AIMAG(PS),REAL(PS))
C
C  DOUBLECHECK TO INSURE CORRECT PHASE TERM
C
      I=1
      DO 20 J=1,3
      PH=CCOS(TH)*CEXP(M)
      PP=CSIN(TH)*CEXP(M)
      SI=CCOS(GA)*CEXP(N)
      PS=CSIN(GA)*CEXP(N)
      X(5)=ATAN2(AIMAG(PH),REAL(PH))
      X(6)=ATAN2(AIMAG(PP),REAL(PP))
      X(7)=ATAN2(AIMAG(SI),REAL(SI))
      X(8)=ATAN2(AIMAG(PS),REAL(PS))
      Y1=PI/2.0
      Y3=ABS(X(I+4)-X(I))
      Y4=ABS(X(I+6)-X(I+2))
      Y5=ABS(X(I+5)-X(I+1))
      Y6=ABS(X(I+7)-X(I+3))
      IF(Y3.GT.0.00001.OR.Y5.GT.0.00001) TH=TH+Y1
      IF(Y4.GT.0.00001.OR.Y6.GT.0.00001) GA=GA+Y1
      IF(Y3.LT.0.00001.AND.Y4.LT.0.00001) GO TO 22
      GO TO 20
   22 IF(Y5.LT.0.00001.AND.Y6.LT.0.00001) GO TO 21
   20 CONTINUE
   21 CONTINUE
      IF(REAL(TH).GT.6.283185307) TH=TH-6.283185307
      IF(REAL(GA).GT.6.283185307) TH=TH-6.283183307
      PHDOT=-CF*PH+PP
      PPDOT=-SIGP*PH+CF*PP
      IF(SF.EQ.CMPLX(0.0,0.0)) SI=CMPLX(0.0,0.0)
      SIDOT=-SF*SI+PS
      PSDOT=-SIGS*SI+SF*PS
      IF(ISSW(4)) 26,25
   26 WRITE(LP,1) PH,PHDOT,SI,SIDOT
      WRITE(LP,1) PP,PPDOT,PS,PSDOT
      WRITE(LP,1) TH,GA,M,N
    1 FORMAT(4(F12.8,2X),/,4(F12.8,2X))
   25 RETURN
      END
C
C  THIS SUBROUTINE INTEGRATES MAGNITUDE AND PHASE EQUATIONS UPWARD
C
      SUBROUTINE SCSPH(SFQX,CFQX,LP,NS,NF,ZBS,DZ,NDZ,SIGP,SIGS,W1,
     1W2,W3,W4,TH,THDOT,GA,GADOT,M,MDOT,N,NDOT,PH,PHDOT,PP,PPDOT,S
     2SIDOT,PS,PSDOT,DZS,FN,L,IF1,IF2,IF3,IF4,IF14,IF15,MAG)
      COMPLEX DZ,SIGP,SIGS,TH,GA,M,N,THDOT,MDOT,GADOT,NDOT
      COMPLEX THP,GAP,MP,NP,CCSZ,CSSZ,SCSZ,SSSZ,THPDOT,GAPDOT,MPDO
      COMPLEX NPDOT,PH,PP,SI,PS,PHDOT,PPDOT,SIDOT,PSDOT
      COMPLEX W1,W2,W3,W4,SFQX,CFQX,DZS,W20,YBF(32),ZBF(32)
      DIMENSION IF10(3),IF11(3),W1(1),W2(1),W3(1),W4(1),SFQX(1),CF
      REAL MAG
      IF10(1)=2HDY
      IF10(2)=2HBF
      IF11(1)=2HDZ
      IF11(2)=2HBF
      IF10(3)=2H
      IF11(3)=2H
      N=N+MAG-7
      M=M+MAG-7
      IF(ISSW(10)) 440,441
  440 NNP=1
      GO TO 442
  441 NNP=NDZ/32
  442 CONTINUE
      THP=CMPLX(0.0,0.0)
```

```
      THP=CMPLX(0.0,0.0)
      GAP=CMPLX(0.0,0.0)
      MPDOT=CMPLX(0.0,0.0)
      NPDOT=CMPLX(0.0,0.0)
      MP=CMPLX(0.0,0.0)
      NP=CMPLX(0.0,0.0)
      THPDOT=CMPLX(0.0,0.0)
      GAPDOT=CMPLX(0.0,0.0)
      CALL EXEC(14,103B,YBF,128,IF10,L)
      CALL EXEC(14,103B,ZBF,128,IF11,L)
      N1=FN+1
      JA=FN
      A=FN-JA
      IF(NS.GE.32) NS=1
      IF(A.EQ.0.0.AND.NS.EQ.1) N1=FN
      IF(ISSW(5)) 4,9
    4 CCSZ=TH*57.295779
      SSSZ=GA*57.295779
    5 WRITE(LP,2)
    2 FORMAT(/"INTEGRATE IN PHASE PLANE-TH,M,GA,N"/)
      XNM=ZBS/FLOAT(NDZ)*FLOAT(L*32+33-NS)
      WRITE(LP,10) XNM,CCSZ,M,SSSZ,N
    9 DO 51 I=1,N1
      CALL EXEC(14,103B,CFQX,128,IF14,L)
      CALL EXEC(14,103B,SFQX,128,IF15,L)
      IF(I.EQ.N1.AND.A.NE.0.0) NF=A*32
      IF(NS.NE.1) NF=NS+NF
      IF(NF.GT.32) NF=32
   12 DO 50 J=NS,NF
      NX=33-J
      W1(NX)=CMPLX(0.0,0.0)
      W2(NX)=CMPLX(0.0,0.0)
      W3(NX)=CMPLX(0.0,0.0)
      W4(NX)=CMPLX(0.0,0.0)
      THP=TH-DZ*THDOT
      GAP=GA-DZS*GADOT
      MP=M-DZ*MDOT
      NP=N-DZS*NDOT
      CCSZ=CCOS(THP)
      CSSZ=CSIN(THP)
      SCSZ=CCOS(GAP)
      SSSZ=CSIN(GAP)
      THPDOT=-SIGP*CCSZ*CCSZ-CSSZ*CSSZ+2.*CFQX(NX)*CCSZ*CSSZ
      GAPDOT=-SIGS*SCSZ*SCSZ-SSSZ*SSSZ+2.*SFQX(NX)*SCSZ*SSSZ
      MPDOT=CFQX(NX)*(CSSZ*CSSZ-CCSZ*CCSZ)+(1.-SIGP)*CSSZ*CCSZ
      NPDOT=SFQX(NX)*(SSSZ*SSSZ-SCSZ*SCSZ)+(1.-SIGS)*SSSZ*SCSZ
      TH=TH-.5*DZ*(THDOT+THPDOT)
      M=M-.5*DZ*(MDOT+MPDOT)
      GA=GA-.5*DZS*(GADOT+GAPDOT)
      N=N-.5*DZS*(NDOT+NPDOT)
      CCSZ=CCOS(TH)
      CSSZ=CSIN(TH)
      SCSZ=CCOS(GA)
      SSSZ=CSIN(GA)
      THDOT=-SIGP*CCSZ*CCSZ-CSSZ*CSSZ+2.*CFQX(NX)*CCSZ*CSSZ
      GADOT=-SIGS*SCSZ*SCSZ-SSSZ*SSSZ+2.*SFQX(NX)*SCSZ*SSSZ
      MDOT=CFQX(NX)*(CSSZ*CSSZ-CCSZ*CCSZ)+(1.-SIGP)*CSSZ*CCSZ
      NPDOT=SFQX(NX)*(SSSZ*SSSZ-SCSZ*SCSZ)+(1.-SIGS)*SSSZ*SCSZ
      W1(NX)=TH
      W2(NX)=M
      W3(NX)=GA
      W4(NX)=N
      YBF(NX)=CCSZ*CEXP(M)
      ZBF(NX)=CMPLX(0.0,0.0)
   60 IF(NX.NE.1) GO TO 50
      CALL EXEC(15,103B,W1,128,IF1,L)
      CALL EXEC(15,103B,W2,128,IF2,L)
```

```
      CALL EXEC(15,103B,W3,128,IF3,L)
      CALL EXEC(15,103B,W4,128,IF4,L)
      CALL EXEC(15,103B,YSF,128,IF10,L)
      CALL EXEC(15,103B,ZBF,128,IF11,L)
      L1=L
      L=L-1
      NS1=NS
      NS=1
   30 IF(ISSW(5)) 7,50
    7 DO 3 JX=NS1,NF,NNP
      JY=33-JX
      Z=FLOAT(JY-1+(L+1)*32)*ZBS/FLOAT(NDZ)
      CCSZ=W1(JY)*57.295779
      SSSZ=W3(JY)*57.295779
    3 WRITE(LP,10) Z,CCSZ,W2(JY),SSSZ,W4(JY)
   10 FORMAT(F7.2,2X,8(F6.3,1X))
   70 FORMAT(/I6/)
   50 CONTINUE
   51 CONTINUE
      IF(NF.EQ.32) GO TO 6
      CALL EXEC(15,103B,W1,128,IF1,L)
      CALL EXEC(15,103B,W2,128,IF2,L)
      CALL EXEC(15,103B,W3,128,IF3,L)
      CALL EXEC(15,103B,W4,128,IF4,L)
      IF(ISSW(5)) 11,6
   11 DO 40 JX=NS,NF,NNP
      JY=33-JX
      Z=FLOAT(JY-1+L*32)*ZBS/FLOAT(NDZ)
      CCSZ=W1(JY)*57.295779
      SSSZ=W3(JY)*57.295779
   40 WRITE(LP,10) Z,CCSZ,W2(JY),SSSZ,W4(JY)
    6 NX=33-NF
      MAG=AMAX1(REAL(M),REAL(N))
      M=M-MAG+7
      N=N-MAG+7
      PH=CCOS(TH)*CEXP(M)
      PP=CSIN(TH)*CEXP(M)
      SI=CCOS(GA)*CEXP(N)
      PS=CSIN(GA)*CEXP(N)
      IF(SFQX(NX).EQ.CMPLX(0.0,0.0)) SI=CMPLX(0.0,0.0)
      PHDOT=-CFQX(NX)*PH+PP
      PPDOT=-SIGP*PH+CFQX(NX)*PP
      SIDOT=-SFQX(NX)*SI+PS
      PSDOT=-SIGS*SI+SFQX(NX)*PS
    8 NS=NF
      RETURN
      END
      END$
```

REFERENCES

(1) Pekeris, C. L., 1948, "Theory of Propagation of Explosive Sound in Shallow Water", Memoir 27, Geological Society of America.

(2) Thomson, W. T., 1950, "Transmission of Elastic Waves Through a Stratified Solid Medium", J. Appl. Phys., 21:89.

(3) Haskell, N. A., 1953, "The Dispersion of Surface Waves on Multilayered Media", Bull. Seis. Soc. Am., 43:17.

(4) Black, M. C., E. W. Carpenter, A. J. M. Spencer, 1960, "On the Solution of One Dimensional Elastic Wave Propagation Problems in Stratified Media by the Method of Characteristics", Geophys. Prosp., 8:128.

(5) Ames, W. F., 1969, Numerical Methods for Partial Differential Equations, Barnes and Noble, New York.

(6) Helmberger, D. V., 1968, "The Crust-Mantle Transition in the Bering Sea", Bull. Seis. Soc. Am., 58:179.

(7) Fuchs, K. and G. Müller, 1971, "Computation of Synthetic Seismograms with the Reflectivity Method and Comparison with Observations", Geophys. J. R. Astr. Soc., 23:417.

(8) Williams, A. O., 1976, "Discrete, Continuous, and Virtual Modes in Underwater Modes", Applied Research Laboratory of the Univ. of Texas Report 40.

(9) Bucker, H. P., 1970, "Sound Propagation in a Channel with Lossy Boundaries", JASA, 48:1188.

(10) Miller, J. F., and F. R. Ingenito, 1975, "Normal Mode Fortran Program for Calculating Sound Propagation in the Ocean", Naval Research Laboratory Report 3071.

(11) Hawker, K. E., et al., 1977, "Results of a Study of the Ocean Bottom Interaction of Underwater Sound", Applied Research Laboratory of the Univ. of Texas Report 27.

(12) Gonzalez, R. G. and K. E. Hawker, 1977, "On the Calculation of Acoustic Normal Modes Using Numerical Integration", Applied Research Laboratory of the University of Texas Report 2.

(13) Tappert, F. D., 1977, "Selected Applications of the Parabolic Equation Method in Underwater Acoustics", International Workshop on Low-Frequency Propagation and Noise, 1:155.

(14) DiNapoli, F. R., 1971, "Fast Field Program for Multilayered Media", Naval Underwater Systems Report 4103.

(15) Oppenheim, A. V., G. V. Frisk, and D. R. Martinez, 1978, "An Algorithm for the Numerical Evaluation of the Hankel Transform", Proc. IEEE, 66:264.

(16) Stickler, D. C., 1975, "Normal Mode Program with Both Discrete and Branch Line Contributions", JASA, 57:856.

(17) McKisic, J. M. and D. P. Hamm, 1976, "New Method for Normal-Mode Models of Sound Propagation in the Ocean", JASA, 59:294.

(18) Baggeroer, A. B., "A State Variable Method for Solving Acoustic Wave Equations", SACLANT ASW Research Centre Memorandum, in press.

(19) Ewing, W. M., W. S. Jardetsky, and F. Press, 1957, Elastic Waves in Layered Media, McGraw-Hill, New York.

(20) Athans, M. and P. L. Falb, 1966, Optimal Control, McGraw-Hill, New York.

(21) Kirk, D. E., 1970, Optimal Control Theory, Prentice-Hall, Englewood Cliffs, New Jersey.

(22) Kutschale, H. W., 1971, "The Integral Solution of the Sound Field in a Multilayered Liquid-Solid Half Space", Lamont-Doherty Geological Observatory Report 1.

(23) Kutschale, H. W., 1972, "Further Investigation of the Integral Solution of the Sound Field in a Multilayered Liquid-Solid Half Space", Lamont-Doherty Geological Observatory Report 6.

(24) Morse, P. M. and H. Feschback, 1953, Methods of Theoretical Physics, Volume 1, McGraw-Hill, New York.

(25) Brekhovskikh, L., 1960, Waves in Layered Media, Academic Press, New York.

(26) Baggeroer, A. B., 1970, State Variables and Communication Theory, MIT Press, Cambridge, Massachusetts.

(27) Golomb, M. and M. Shanks, 1965, Elements of Ordinary Differential Equations, McGraw-Hill, New York.

(28) Hamilton, E. L., 1974, "Geoacoustic Models of the Sea Floor" in Physics of Sound in Marine Sediments, edited by L. Hampton, Plenum, New York.

(29) Hamilton, E. L., 1971, "Elastic Properties of Marine Sediments", J. Geophys. Res., 76:579.

(30) Bender, C. M. and S. A. Orszag, Advanced Mathematical Methods for Scientists and Engineers, McGraw-Hill, New York.

(31) Dahlquist, G. and A. Björck, 1974, Numerical Methods, Prentice-Hall, Englewood Cliffs, New Jersey.

(32) Tolstoy, I. and C. S. Clay, 1966, Ocean Acoustics, McGraw-Hill, New York.

(33) Grant, F. S. and G. F. West, 1965, Interpretation Theory in Applied Geophysics, McGraw-Hill, New York.

(34) Hamilton, E. L., 1976, "Sound Attenuation as a Function of Depth in the Sea Floor", JASA, 59:528.

(35) Hamilton, E. L., 1976, "Attenuation of Shear Waves in Marine Sediments", JASA, 60:334.

(36) Brockett, R. W., 1970, Finite Dimensional Linear Systems, Wiley, New York.

(37) McGillem, C. D. and G. R. Cooper, 1971, Continuous and Discrete Signal and System Analysis, Holt, Rinehart, and Winston, New York.

(38) Hamilton, E. L., 1976, "Sound Velocity -- Density Relations in Sea-Floor Sediments and Rocks", JASA, 63:366.