T-3873

RAY-THEORETIC MODELING IN 2.5-D SMOOTHED MEDIA

by

Abdullah A. Al-Saleh

T-3873

A thesis submitted to the Faculty and the Board of Trustees of the Colorado School of Mines in partial fulfillment of the requirements for the degree of Master of Science (Goephysics).

Golden, Colorado
Date April 1, 1991

Signed: _____
Abdullah A. Al-Saleh

Approved: _____
Norman Bleistein
Thesis Advisor

Golden, Colorado
Date 3 April 1991

_____
Phillip R. Romig
Head,
Department of Geophysics

ABSTRACT

The main thrust of the thesis is a finite difference ray-tracing code which solves the ray equations for the constituents of ray trajectories, traveltime, and leading order amplitude of the ray-theoretic (WKBJ) solution of the acoustic wave equation. The primary objective behind writting the code is to calculate the ray-theoretic in-plane field in 2.5-D continuous media which are characterized by continuous physical parameters. The code will make use of the initial conditions provided at the source location to estimate the wave field along rays. It provides an alternative WKBJ modeling tool for the 2.5-D inversion codes. This alternative modeling is carried out using an additional feature in the code which allows smoothing of the original discontinuous propagation speed of the model through a smoothing parameter in the parameter file of the code. I believe that the inversion codes will produce better inversion results with smoothed models than those inversions that use ray-theoretic modeling in piecewise continuous (discontinuous) earth models. The reason is that refracted rays associated with discontinuous models exhibit great sensitivity to small perturbations along the interfaces, especially in the presence of a relatively large speed contrasts across them. The sensitivity was found to be true whether the perturbations were due to true changes in the model or due to artifacts of the modeling process (i.e. modeling of interfaces). A shifted version of a truncated 2-D Gaussian filter convolved in the spatial domain with the background propagation speed was used to

produce smoothed speed models from discontinuous ones. The smoothed models were found to be effective in diminishing ray trajectories sensitivity as well as the amplitude sensitivity. Further more, they provide more uniform distribution of ray trajectories. This was confirmed by comparing the two types of results obtained from smoothed and discontinuous models.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my advisor Norman Bleistein for his help and encouragement during my study at Colorado School of Mines. His enthusiasm to explore science and to search new ideas is a non-stopping source of research potential. This source has been the supply of many important steps during the progress of this thesis. In addition, I would also like to thank Professors Frank Haddsell and Ronald Knoshaung for serving on my committee. Also, I like to thank Dr. Jack Cohen for his helpful comments.

I am grateful for the continuing support provided by Saudi ARAMCO company at Dhahran, Saudi Arabia during the four years 1978-1990. The computer facility at School of Mines, recently, have made the research work more attractive and much more productive. This new environment have made my work much easier to handle recently.

Warm thanks go to my parents, Abdulrhman and Munirah, who want to see me happy at all moments of my life. Special thanks go to my brother, Yahya, for his care and encouragement. Finally, I would like to thank my wife Omm Abdulrhman for her great patience and continuing support.

1. INTRODUCTION.

This thesis describes a finite difference code for solving the ray system for the constituents of in-plane ray trajectories, travel time and leading order amplitude of the ray theoretic or WKBJ solution to the constant density acoustic wave equation in a 2.5-D continuous media. 2.5-D media are those media in which there is no speed variation in the transverse direction perpendicular to the line of observations. Geologically, this implies that the dip in this out-of-plane direction is zero (Figure 1.1). For this ray tracing problem the speed propagation must be continuous through out the entire model. The system requires, in addition, a continuous second derivative of the speed. In practice, this is not true because the second speed derivative as well as the first speed derivative, are not continuous at the edges of the speed cells. However, we hope that smoothing, implemented by the code, will minimize the effect of these discontinuities so they can be ignored, especially in the neighborhood of the model interfaces.

This project was motivated by a desire to have a WKBJ package for smoothed propagation speeds for application to inversion codes. The reason is that I believe they will produce better inversion results than are presently obtained with ray theoretic modeling in piecewise continuous (discontinuous) earth models. Modeling experiments that were carried out comparing the two types of results confirm this

conjecture. Ray trajectories show great sensitivity to small perturbations in interfaces with relatively large speed contrasts across them.

The wave field in a background medium is required by most migration/inversion codes. Ray-theoretic modeling can be used to obtain this background wave field. This modeling technique is based on the ray method which assumes a high frequency wave field. This approximate method provides a good estimation of the field when the wavelength of the wavefield emitted from the source is much smaller than the characteristic dimensions of the medium (Cerveny et. al., 1972), typically by a factor of three or more (Bleistein, 1984). In 2.5D media, it was found to be in good agreement with observations and with asymptotic expansion of exact solutions, in those cases for which an exact solution is available (Bleistein N., 1984 ; Dong W., 1989). Any discrepancies are usually attributed to diffraction effects or to special regions where the high frequency assumption fails, such as near pinchouts.

Conventional ray tracing for imaging purposes is usually carried out by tracing rays from each output point in the subsurface to every point at the observation surface (sources/receivers). This would require a heavy computational load if the number of output points is excessively large, $O(N^4)$, where N is the number of output points along one direction. Gray (1986), suggested an alternative method in which rays are traced from the surface points to the subsurface in a "top-down" ray tracing procedure. The field is then evaluated at the output points by interpolation using the nearby rays.

This method predicts that the cost of ray tracing reduces to $O(N_x N_{ray} N_z)$ operations, where N_{ray} is the number of rays traced from each depth, and N_x and N_z are number of output points along x and z axes respectively. This cost is comparable to the operation count for Kirchhoff summation. In addition, as suggested by Gray, the method can accommodate large velocity variations more satisfactorily. The above tracing method were implemented through a ray-theoretic modeling program which was used to generate the simple tracing examples shown latter in this thesis.

The code is a fast ray tracing program due to two reasons. One reason is the implementation of the above tracing approach suggested by Gray (1986). The other reason is due to an adaptive stepping algorithm used by the code when rays are traced from a depth level to another (Press et. al., 1988). The code also has some disadvantages. One of which is the relatively long computer time spent in smoothing the speed when the sampling intervals are chosen to be small. This is due to the costly two dimensional convolutional operation. The other disadvantage is that the code does not continue tracing beyond the zero value of the vertical slowness, an issue that should be addressed in future.

Ray tracing, in general, was extensively studied in discrete models as well as in models with no discontinuities (continuous). Smith et. al. (1986) have stated that neither the discrete boundary case nor the continuous case accurately describes the situation faced in tomographic reconstruction of biological tissues. The reason is that,

on the one hand the body contains many discrete boundaries that mark an abrupt change in the propagation speed of waves. On the other hand, we do not know a priori the type, location, and angle of interfaces within the medium of interest. I believe that a similar situation also exist in seismic inverse problems.

They also used and applied an appropriate low pass filtering technique, and reduced the sampled discrete boundary to a continuous refractive index function. In this way the index of refraction is not singular at the interface and rays can be traced continuously to the next layer. In addition, they have shown that when the index of refraction has constant but different speed values across the interface, the refraction angle predicted by integrating the angular form of the ray bending equation over the smoothed boundary yields the same ray bending as predicted by Snell's Law.

The model in practical situations is subdivided into a network of speed cells to reduce the complexity of real speed distributions to simpler ones. The distribution within each cells is approximated by analytic functions that are computed based on the values of the speed at the grid points. Cerveny (1989) mentioned that the computed travel times are not as influenced by the fictitious edges in interfaces caused by a piece-wise linear interpolation as the ray amplitudes do. Implementation of splines removes these fictitious edges from interfaces and suppresses globally the oscillations in the velocity distribution. However, in some cases a standard spline algorithm cannot remove the oscillation fully, and could generate very unpleasant local oscillations in the velocity itself or in its first derivative (Cerveny, 1989).

Cerveny suggests also using splines with smoothing or splines with tension to overcome the problem. Langan et. al. (1985) used a smoothing function which averages over three speed cells. In this way they attempt to smooth the speed in the model rather than smoothing the interfaces. This method is similar to our smoothing approach for which I use a gaussian filter to smooth the speed rather than averaging over three cells only as I will describe in chapter 3. Chapman and Drummod (1982) remarked that in order to obtain reliable amplitudes, the speed must be a smooth function of the coordinates of the medium, so as to prevent the formation of spurious critical points, caustics, and triplications (Madrid, 1985).

In chapter 2, I will derive the ray system necessary to generate the in-plane traveltimes as well as in-plane ray trajectories in 2.5-D media. It will also be shown that two new auxiliary quantities are required in order to find the in-plane amplitude as well. These two quantities will be derived in the last section of the next chapter. A description of the smoothing filter to smooth the speed propagation of the model will be given in the second section of chapter 3. A brief discussion of the ray tracing algorithm in the code will be given in the third section of chapter 3. This includes an outline of how the code is used to generate ray trajectories outputs as well as amplitude outputs. Examples of ray trajectories will be shown later in the last section of chapter 3. The amplitude discussion as well as some amplitude outputs will be postponed to the end of the last section in chapter 3. These examples illustrate the practical importance of sensitivity problem usually encountered in 'ray tracing'

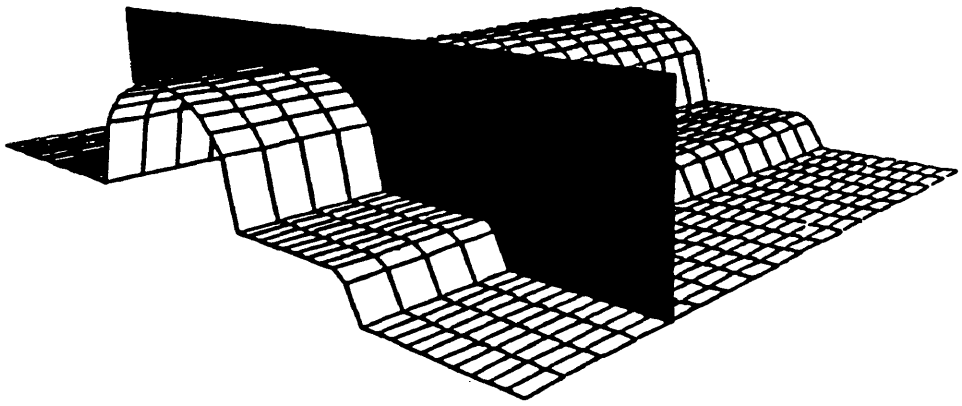

Figure 1.1: The 2.5D geometry. The dip is zero in direction perpendicular to the line of observation.(courtesy of Docherty.)

modeling of wave propagation.

2. THE RAY SYSTEM.

2.1 Traveltime Ray System

Wave equation is the equation of motion which models the propagation of the wave field in an acoustic, isotropic, inhomogeneous media. The constant density wave equation in the frequency domain is

$$\nabla^2 u(\mathbf{x}, \omega) + \frac{\omega^2}{c^2(\mathbf{x})} u(\mathbf{x}, \omega) = 0 , \qquad (2.1.1)$$

where $\mathbf{x}=(x, y, z)$ is the spatial rectangular coordinates, ω is the angular frequency, $c(\mathbf{x})$ is the propagation speed, and $u(\mathbf{x}, \omega)$ may represent either pressure or velocity scalar potentials. I seek a WKBJ or ray-theoretic solution (Bleistein 1988) of equation (2.1.1) of the form

$$u(\mathbf{x}, \omega) = A(\mathbf{x})\, e^{i\omega\tau(\mathbf{x})} , \qquad (2.1.2)$$

where $A(\mathbf{x})$ is the amplitude, and $\tau(\mathbf{x})$ is the phase function of the traveling wave.

Our objective is to model the in-plane propagation of the wave field in a 2.5-D continuous media. These media are characterized by structural variations that are essentially 2-D (cylindrical in three dimensions) while the propagation is 3-D. In such

media, the in-plane propagation of a wave in three dimensions can be described totally in terms of in-plane calculations similar to those in 2-D models (Bleistein, 1986, Bleistein et al., 1987). This is also true for the amplitude of the wave field (the 2.5-D amplitude).

When the above ansatz is used as an approximation to the full solution of the wave equation, equation (2.1.1) can be reduced to a system of first order ordinary differential equations (Bleistein 1984) that involves the relevant ray quantities for approximating the acoustic wave field in the medium. This approximate solution will be accurate when the source wavelet is considered to be a high-frequency wavelet. That is to say, the dominant wavelength in the wavelet must be small enough when compared to the natural length scales of the problem for the formal solution to approximate the exact solution.

Substituting equation (2.1.2) into equation (2.1.1) and then equating the coefficients of ω^2 and ω to zero, we obtain the eikonal equation

$$(\nabla\tau)^2 = \frac{1}{c^2(\mathbf{x})}, \tag{2.1.3}$$

and the transport equation

$$2\nabla\tau\cdot\nabla A + \nabla^2 A = 0. \tag{2.1.4}$$

Choosing z as the running parameter over the rays, the traveltime ray system is

$$\frac{d\mathbf{D}}{dz} = \frac{d}{dz}\begin{bmatrix} x \\ \sigma \\ p \\ \tau \end{bmatrix} = \frac{1}{q}\begin{bmatrix} p \\ 1 \\ ss_x \\ s^2 \end{bmatrix}, \quad q = \sqrt{s^2 - p^2}. \tag{2.1.5}$$

where $\mathbf{D}$ is the reduced form of the vector $(x, \sigma, p, q, \tau)^T$, σ is a running parameter along the rays, needed for amplitude calculations, $p = \partial\tau/\partial x$ and $q = \partial\tau/\partial z$ are horizontal and vertical slownesses respectively, and $s = s(\mathbf{x})$ is the slowness magnitude defined by $1/c(\mathbf{x})$. We choose the positive sign of the square root for q since z is increasing downward. The initial values of the ray quantities at the source location, denoted by the vector $\mathbf{D}_0$, are independent of the initial depth so

$$\mathbf{D}_0(\beta) = \begin{bmatrix} x_0 \\ \sigma_0 \\ p_0 \\ \tau_0 \end{bmatrix} = \begin{bmatrix} x_0 \\ 0 \\ s_0 \sin\beta \\ 0 \end{bmatrix}, \quad q_0 = s_0 \cos\beta. \tag{2.1.6}$$

A derivation of equation (2.1.5) is included in appendix A. The solution of the system (2.1.5)-(2.1.6) represents part of the full solution given by equation (2.1.2) since it only provides traveltimes while incomplete amplitude information are involved.

The computed traveltimes are identical to the in-plane 2.5-D traveltimes (Bleistein 1984). However, we need the amplitude as well. In our case, the in-plane 2.5-D amplitude, derived in appendix B, can be expressed in terms of quantities found by a standard 2-D ray tracing

$$A(z,\beta) = \frac{1}{4\pi}\sqrt{\frac{c(z_0)\sec\beta}{\sigma(z,\beta)\,|\partial x/\partial\beta|}} \tag{2.1.7}$$

where $\partial x/\partial\beta$ is a new auxiliary quantity which is not included in the system (2.1.5) and is also a function of z and β. In order to find the in-plane amplitude, it is necessary to find $\partial x/\partial\beta$ in addition to the above system in equation (2.1.5). Adding $\partial x/\partial\beta$ to equation (2.1.5) requires further extension to the system. The rest of this chapter is devoted to deriving this extension. Note also that this formula can not be used to evaluate the amplitude for rays with horizontal initial slowness vector since the amplitude becomes singular for these rays at the source location.

2.2 A Result From Ordinary Differential Equations Theory.

We describe here the underlying ordinary differential equation theory for determination of the derivatives of unknowns in a system of equations with respect to a parameter. In our application, we require $\partial x/\partial\beta$ to be a function of (z,β), where z is the independent variable and β is the parameter. The quantity $\partial x/\partial\beta$ is the missing ray component that must be added to the vector **D** in order to compute the in-plane 2.5-D amplitude along ray trajectories. The unknown $\partial x/\partial\beta$, will be treated now as a new component in the extended version of the vector **D**, denoted by $\overline{\mathbf{D}}$.

In order to include the quantity $\partial x/\partial\beta$ to the vector **D**, we must first find an expression for its derivative with respect to z. In addition, it is necessary that we can interchange the order of z and β derivatives. Since the parameters z and β are independent variables along rays, as mentioned before, we can carry out this interchange without altering the final output of differentiation. This assumption of independence allows us to express the z-derivative of $\partial x/\partial\beta$ in terms of other ray quantities. Before we do that we note that β will be treated as a parameter (is constant when z is changing) and hence the partial derivative can be replaced by a total derivative. The expression can now be written as follows

$$\frac{d}{dz}\left[\frac{dx}{d\beta}\right] = \frac{d}{d\beta}\left[\frac{dx}{dz}\right] = \frac{d}{d\beta}\left[\frac{p}{q}\right] = \frac{1}{q}\frac{dp}{d\beta} - \frac{p}{q^2}\frac{dq}{d\beta} \qquad (2.2.1)$$

The quantity $dq/d\beta$ can be expressed in terms of $dp/d\beta$ and $dx/d\beta$ using the eikonal equation (2.1.3)

$$p^2 + q^2 = s^2(\mathbf{x}). \qquad (2.2.2)$$

This expression is linear in $dp/d\beta$ and $dx/d\beta$

$$\frac{dq}{d\beta} = \frac{ss_x}{q}\frac{dx}{d\beta} - \frac{p}{q}\frac{dp}{d\beta}. \qquad (2.2.3)$$

Substituting (2.2.3) in (2.2.1) yields

$$\frac{d}{dz}\left[\frac{dx}{d\beta}\right] = \frac{s^2}{q^3}\frac{dp}{d\beta} - ss_x\frac{p}{q^3}\frac{dx}{d\beta}. \qquad (2.2.4)$$

The above expression involves the β-derivative of the horizontal slowness p. This forces us to compute and add $dp/d\beta$ as well as $dx/d\beta$ to the vector $\mathbf{D}$, in order to determine $dx/d\beta$ along the rays. Rather than deriving these derivatives now in an ad hoc way by differentiating each component alone, we recall a result from ordinary differential equations theory which enables us to derive the β-derivatives of all ray quantities of the vector (x, σ, p, q, τ) in a matrix form and to provide the final expression of the ray system in a vector notation. This gives easier look at the unknowns and equations that are computed in the code to find traveltimes and

amplitudes of the field. In addition, it provides easier way to extend the system for the case of 3-D ray tracing problem.

To obtain these derivatives in a systematic way, we first derive the result for a general system of first order ordinary differential equations and then apply the result to our specific system in equation (2.1.5) in the following section. We start by introducing a general vector of N dependent unknown variables denoted by $\mathbf{Y} = (y_1, y_2, \cdots, y_N)^T$ along with the independent variable z, which corresponds to z in the system (2.1.5). Let the vector $\mathbf{Y}$ and z satisfy the following ordinary differential equations in the following vector notation

$$\mathbf{Y}' = \mathbf{f}(z, \mathbf{Y}), \quad \mathbf{Y}(z_0) = \boldsymbol{\eta} \tag{2.2.5}$$

where $'$ denotes d/dz, $\mathbf{f}=(f_1, f_2, \cdots, f_n)^T$ is a continuous vector function of the (n+1)-dimensional vector $(z, \mathbf{Y})$, $\boldsymbol{\eta}=(\eta_1, \eta_2, \cdots, \eta_n)$ is the initial conditions vector of $\mathbf{Y}$, and z_0 is the initial value of the variable z.

The vector $\mathbf{Y}$ depends on the initial vector $\boldsymbol{\eta}$, so we can write

$$\mathbf{Y} = \mathbf{Y}(z, \boldsymbol{\eta}) \tag{2.2.6}$$

Assume now, that the initial vector $\boldsymbol{\eta}$ depends on another independent variable which is assumed to be constant when z is changing, and let us denote the new parameter by the symbol β, which corresponds to β in equation (2.1.6).

Differentiating (2.2.6) with respect to β yields

$$\mathbf{Y}_\beta = \frac{\partial \mathbf{Y}}{\partial \beta} = \frac{\partial \mathbf{Y}}{\partial \eta_j}\frac{\partial \eta_j}{\partial \beta} = \mathbf{Y}_\eta \cdot \frac{d\boldsymbol{\eta}}{d\beta}, \quad j = 1, \ldots, N, \tag{2.2.7}$$

where

$$\mathbf{Y}_\eta = \left[\frac{\partial y_i}{\partial \eta_j}\right] \quad i, j = 1,2,\ldots,N . \tag{2.2.8}$$

To include the vector $\mathbf{Y}_\beta$ in the system of equation (2.2.5), we must find the z-derivatives of the vector $\mathbf{Y}_\beta$. By differentiating the vector $\mathbf{Y}_\beta$ with respect to z we can write

$$\frac{d\mathbf{Y}_\beta}{dz} = \frac{d\mathbf{Y}_\eta}{dz}\frac{d\boldsymbol{\eta}}{d\beta}. \tag{2.2.9}$$

In the above equation, the only problem left is to find $d\mathbf{Y}_\eta/dz$ in a matrix form. A result that can be found in Coddington & Levinson (1964) in which the initial conditions vector $\boldsymbol{\eta}$ depends on the parameter β, $\boldsymbol{\eta}(\beta)$, will enable us to express $d\mathbf{Y}_\eta/dz$ in terms of $\mathbf{Y}_\eta$. This result states that the matrix $\mathbf{Y}_\eta$ satisfies the following linear system

$$\frac{d\mathbf{Y}_\eta}{dz} = \mathbf{M}(z)\,\mathbf{Y}_\eta, \qquad \mathbf{Y}_\eta(z_0, \boldsymbol{\eta}) = I_{N\times N}, \tag{2.2.10}$$

where

$$\mathbf{M}(z) = \left[\frac{\partial f_i}{\partial y_j}\right] \qquad i,j = 1,2,...,N \ ,$$

and $I_{N\times N}$ is the $N\times N$ identity matrix.

Equation (2.2.10) says that the z-derivative of the vector $\mathbf{Y}_\eta$ is given by the vector $\mathbf{Y}_\eta$ multiplied from left by the matrix $\mathbf{M}(z)$. Implementing the above result in equation (2.2.9) yields

$$\frac{d\mathbf{Y}_\beta}{dz} = \mathbf{M}(z)\,\mathbf{Y}_\eta \frac{d\eta}{d\beta} = \mathbf{M}(z)\,\mathbf{Y}_\beta. \tag{2.2.11}$$

Finally, we can write the modified system for the general case, were the vector $\mathbf{Y}$ is now extended to the vector $\bar{\mathbf{Y}} = (\mathbf{Y}, \mathbf{Y}_\beta)^T$, as follows

$$\bar{\mathbf{Y}}' = \frac{d\bar{\mathbf{Y}}}{dz} = \frac{d}{dz}\begin{bmatrix} \mathbf{Y} \\ \\ \mathbf{Y}_\beta \end{bmatrix} = \begin{bmatrix} \mathbf{f}(z, \mathbf{Y}) \\ \\ \mathbf{M}(z)\,\mathbf{Y}_\beta \end{bmatrix}, \tag{2.2.12}$$

where the initial conditions are given by

$$\bar{\mathbf{Y}}_0 = \begin{bmatrix} \eta(\beta) \\ \\ \eta_\beta(\beta) \end{bmatrix} = \begin{bmatrix} \mathbf{Y}(z_0, \beta) \\ \\ \mathbf{Y}_\beta(z_0, \beta) \end{bmatrix} \tag{2.2.13}$$

Equations (2.2.12)-(2.2.13) represent the necessary and sufficient system (in vector notation) to find the $\mathbf{Y}_\beta$ components along with the old $\mathbf{Y}$ components. The

general result in these equations will be applied to the system (2.1.5) in the next section.

2.3 Amplitude Ray System.

The ray system in equation (2.1.5) of the previous section, consists of five dependent unknowns and one independent variable, namely z. One of these dependent unknows is updated, alternatively, using the eikonal equation as in equation (2.1.5). However, if we want to use the result of last section we must consider the full vector (x, σ, p, q, τ). This is due to the fact that we can not construct the matrix $\mathbf{M}(z)$ unless using the above vector. With this in mind, the general form of the matrix $\mathbf{M}(z)$ for our case becomes

$$\mathbf{M}(z) = \begin{bmatrix} \frac{\partial f_1}{\partial y_1} & \frac{\partial f_1}{\partial y_2} & \frac{\partial f_1}{\partial y_3} & \frac{\partial f_1}{\partial y_4} & \frac{\partial f_1}{\partial y_5} \\ \frac{\partial f_2}{\partial y_1} & \frac{\partial f_2}{\partial y_2} & \frac{\partial f_2}{\partial y_3} & \frac{\partial f_2}{\partial y_4} & \frac{\partial f_2}{\partial y_5} \\ \frac{\partial f_3}{\partial y_1} & \frac{\partial f_3}{\partial y_2} & \frac{\partial f_3}{\partial y_3} & \frac{\partial f_3}{\partial y_4} & \frac{\partial f_3}{\partial y_5} \\ \frac{\partial f_4}{\partial y_1} & \frac{\partial f_4}{\partial y_2} & \frac{\partial f_4}{\partial y_3} & \frac{\partial f_4}{\partial y_4} & \frac{\partial f_4}{\partial y_5} \\ \frac{\partial f_5}{\partial y_1} & \frac{\partial f_5}{\partial y_2} & \frac{\partial f_5}{\partial y_3} & \frac{\partial f_5}{\partial y_4} & \frac{\partial f_5}{\partial y_5} \end{bmatrix},$$

$$= \begin{bmatrix} 0 & 0 & 1/q & -p/q^2 & 0 \\ 0 & 0 & 0 & -1/q^2 & 0 \\ (ss_x)_x/q & 0 & 0 & -ss_x/q^2 & 0 \\ (ss_x)_z/q & 0 & 0 & -ss_z/q^2 & 0 \\ 2ss_x/q & 0 & 0 & -s^2/q^2 & 0 \end{bmatrix}, \qquad (2.3.1)$$

where $(..)_x$, $(..)_z$ denote $\partial/\partial x$, $\partial/\partial z$ respectively, and the vector $\mathbf{f}$ is given by

$$\mathbf{f} = \frac{1}{q} \begin{bmatrix} p \\ 1 \\ ss_x \\ ss_z \\ s^2 \end{bmatrix}. \qquad (2.3.2)$$

The two unknowns $\partial\sigma/\partial\beta$ and $\partial\tau/\partial\beta$ are not involved in the computation of ray amplitude. Therefore, if we disregard them in the extended system we will still be able to compute the ray amplitude completely. Applying the general result of last section to the system (2.1.5) and disregarding the unnecessary ray quantities leads to an expression for the vector $d(\mathbf{D}_\beta)/dz$ given as

$$\frac{d\mathbf{D}_\beta}{dz} = \begin{bmatrix} \dfrac{dx}{d\beta} \\ \\ \dfrac{dp}{d\beta} \end{bmatrix} = \begin{bmatrix} \dfrac{1}{q}\dfrac{dp}{d\beta} - \dfrac{p}{q^2}\dfrac{dq}{d\beta} \\ \\ \dfrac{(ss_x)_x}{q}\dfrac{dx}{d\beta} - \dfrac{ss_x}{q^2}\dfrac{dq}{d\beta} \end{bmatrix}, \tag{2.3.3}$$

where $\mathbf{D}_\beta$ is the reduced version of the vector $(x_\beta, \sigma_\beta, p_\beta, q_\beta, \tau_\beta)$. Extending the vector $d\mathbf{D}/dz$ by the vector $d\mathbf{D}_\beta/dz$ and using equation (2.2.3), yields a new vector equation in terms of the new extended vector $\bar{\mathbf{D}} = (\mathbf{D}, \mathbf{D}_\beta)^T$ given by

$$\frac{d\bar{\mathbf{D}}}{dz} = \frac{d}{dz}\begin{bmatrix} \mathbf{D} \\ \\ \mathbf{D}_\beta \end{bmatrix}, \tag{2.3.4}$$

$$= \frac{d}{dz}\begin{bmatrix} x \\ \sigma \\ p \\ \tau \\ \dfrac{dx}{d\beta} \\ \\ \dfrac{dp}{d\beta} \end{bmatrix} = \frac{1}{q}\begin{bmatrix} p \\ 1 \\ ss_x \\ s^2 \\ \dfrac{s^2}{q^2}\dfrac{dp}{d\beta} - ss_x\dfrac{p}{q^2}\dfrac{dx}{d\beta} \\ \\ \dfrac{ss_x p}{q^2}\dfrac{dp}{d\beta} + \dfrac{(ss_{xx} - s_x{}^2p^2)}{q^2}\dfrac{dx}{d\beta} \end{bmatrix}$$

The above vector equation along with the alternative formula to compute $q(z, \beta)$ in equation (2.1.5) represents the final form of the ray system that is used in the code

to compute the travel time, phase, and amplitude needed to find the in-plane high frequency wave field in any 2.5D inhomogeneous medium.

3. RAY MODELING IN 2.5-D SMOOTHED MEDIA.

3.1 Introduction

In practice, actual subsurface media are described in terms of piecewise continuous models. Such a description may not necessarily represent a correct assumption and may not reflect the actual nature of the subsurface. Physical parameters can have a smooth transitional behavior from one geological lithology to another as was found for some sedimentary boundaries. In addition, the details of interfaces usually introduced in the model are often contaminated with artifact noise caused either by erroneous picks of the control points (interface points) or by the method of modeling a continuous reflector from discrete points (e.g. spline interpolation).

In either case, the noise will appear as small perturbations along the interfaces which usually cause a major practical problem when implementing ray methods to estimate the propagating field. The output of the modeling (the computed field) will represent a field that is contaminated with noise as a result of the sensitivity of ray response to the artifact details as will be shown in later examples. Sometimes these perturbations cause rays to deviate significantly and generate artifact gaps that occupy large areas in the model which are not accessible by a reasonable density of rays.

These gaps can become considerably large for some geological settings in the model and prevent satisfactory wave field modeling, as will be show using simple examples.

Specifically, if important geological structures exist within these gaps, rays will miss these structures and the field cannot be reliably modeled in these areas by the ray method. This situation in turn has disadvantages that can place some limitations on the performance of the inversion process. The limitations are related to the fact that the inversion process needs reasonable and fair estimates of the wave field in the area in order to provide a satisfactory imaging of the subsurface. Within these artifact gaps, the computation of the field can lead to unsatisfactory imaging or no imaging at all within a region that could be vital to the interpreter.

In this chapter I will show one ray tracing output generated by our code to illustrate the correctness of the code algorithm. Also many outputs will be given to illustrate the importance of the sensitivity problem associated intrinsically with the ray method when used in either modeling or inversion problems. In addition, I will provide some smoothed outputs to show the effectiveness of 2-D Gaussian smoothing in reducing the impact of this problem. An example of an actual model conducted by Marathon Oil Company will be considered to illustrate the sensitivity problem encountered in practice. I will also provide some tabulated data to illustrate the degree of accuracy of computations attained by the code. A brief discussion of the filter used to smooth the speed will be given in the second section. Finally, a description of the code will also be included in the third section.

3.2 Smoothing Filter.

Removing errors (e.g. perturbations produced by interpolation) such that the modified speed distribution would still accurately depict the discontinuous model is called smoothing (LaFara, 1973). Smoothing two-dimensional data through designing two-dimensional filters, primarily in the frequency domain, has been investigated by many workers. These designs have been considered by Hu and Rabiner (1972) for both IIR (infinite impulse response) and FIR (finite impulse response) two-dimensional filters in wavenumber domain. Two-dimensional convolution filtering as applied to gravity and magnetic maps was discussed by Byerly (1964). Huang (1972) gave a brief discussion in the design of two-dimensional windows for filtering two-dimensional data. He established a result which enables one to obtain good two-dimensional windows from good one-dimensional windows.

All attempts above aimed to design the smoothing filter in the wavenumber domain (using 2-D Fourier transforms) rather than in the spatial domain (using convolution). We will avoid designing our filter in the wavenumber domain for a number of reasons. Most important is the presence of discontinuities on the speed function along each interface in the model. This will correspond to rippling effects in the wavenumber domain that might have unfavorable impact over the filtered speed distribution (smoothed) and hence ray tracing.

Working in the spatial domain enables us to perform smoothing through a two-dimensional convolution between the model speed distribution and the smoothing filter. This has the advantage of being easy to code (four "do loops" at most) and at the same time we avoid any possible significant distortion that might appear if we used transforms methods in the design process (e.g. Fourier transforms). The disadvantage here is the relatively longer computer times associated with convolution using serial type computers. However, convolution is a suitable mathematical operation in parallel computing machines (e.g. Crays). Further more, only one pass over the model is required, so that the computer cost is a relatively small part of the total cost of modeling the wave field.

Since we do require that the modified speed to be as close as possible to the original speed function, we consider only circular symmetric two-dimensional low-pass filters (so not to deform the model). A good hint in choosing the filter is simply to recognize that the number of continuous derivatives possessed by a given continuous function can be viewed as a measure of the smoothness of this function. Therefore, the higher the number of continuous derivatives possessed by the function, the smoother is the function.

Also, there is a proportional relationship between the smoothness of a given function and the compactness of its Fourier transform. This is to say, the smoother the function, as measured by the number of continuous derivatives, the more compact its transform, or the faster it dies away with increasing wavenumber (Bracewell 1965).

That another also, mentioned that the two-dimensional Gaussian distribution is a circular symmetric filter that has an infinite number of continuous derivatives so that the filter can be considered as smooth as possible. Likewise, its transform is as compact as possible. In fact, its transform is another Gaussian which dies faster than any negative integral power of wavenumber.

This indicates that the rate of rejection (the rate at which higher frequencies are attenuated) of this filter is higher than the rate of a filter with any negative integral power of wavenumber, k^{-n} were n is a positive integer and $k=\sqrt{k_x{}^2+k_z{}^2}$ is the wavenumber. This has the effect of removing most of high frequency content in the speed function very fast compared to most other filters. We consider this as a nice property associated with Gaussian filters and a suitable filtering effect that we need for removing the small high frequency perturbations effectively. Jumps in the speed function across interfaces are converted to transitions that link speed values between each adjacent pair of layers in a smooth continuous fashion. The width of these transitions (or the distance they occupy across interfaces) depends on the radial length of the filter as well as on the speed sampling intervals.

Depending on the dominant wavelength in the source wavelet, assume that the 2-D Gaussian filter is sampled to k sampling grid points. If the numbers of speed grid points are M_x and M_z along both the offset and depth directions respectively, the cost of the smoothing operation is $O(kM_xM_z)$. This cost is smaller than the cost of

ray tracing because it involves only multiplications and additions of k points. Ray tracing on the other hand requires solving a total of six coupled first order ordinary differential equations at N_z points $N_{ray}N_x$ number of times, $O(N_{ray}N_xN_z)$, where N_x and N_z are number of output points along x and z directions respectively, and N_{ray} is the number of rays. If we let $N=N_x=N_z$ and $M=M_x=M_z$, then N is usually greater than M and hence the cost of ray tracing is usually greater than smoothing. The relation between the dominant wavelength and the speed cell size should be mentioned here. As a rule of thumb, the wavelength of the wavelet should be bigger than the speed cell size which is a feasible assumption. The reason is that, otherwise the speed cell will be treated as an obstacle rather than a small portion of the speed distribution which may introduce a degradation to the imaging stage. Bleistein (personal communication) suggested at least four cells per wavelength.

The general form of a 2-D continuous Gaussian distribution is,

$$g(x,z) = A\ e^{-(x^2+z^2)}\ , \tag{3.2.1}$$

where x and z are the spatial coordinates and A is a constant.

This expression extends over the unbounded domain in the (x,z)-plane. For a practical implementation we need to truncate the filter at some distance from the origin. This distance should be related somehow to the dominant wavelength in the seismic data. It is well known that the ray method is only valid when each

characteristic dimension of the problem, denoted by L, is large compared to the wavelength λ, (Bleistein, 1984). Following Bleistein, we choose the filter radius to be three-halves of the dominant wavelength in the seismic data. This distance is at our disposal because it is one of the flags in the parameter file of the code. It is this parameter that determines the degree of smoothing we apply to the speed function in the model. So, if we denote this radial distance in the (x, z)-plane by $a = 3\lambda/2$, we can express the truncated filter in polar coordinates as follows,

$$f(\rho) = A\, e^{-\frac{\rho^2}{a^2}} H(a-\rho) \,, \tag{3.2.2}$$

where $H(a-\rho)$ is the Heaviside step function, $\rho = \sqrt{x^2+z^2}$, and $a = 3\lambda/2$.

This truncation produces a jump at $\rho = a$ of magnitude $1/e$. This jump causes a rippling effect in the wavenumber plane (k_x, k_z) and introduces a distortion effect into the smoothed version of the speed. The rippling effect has been confirmed by transforming the filter into the wavenumber domain and displaying the Fourier version of the filter into a 3-D prospective. The plot shows clearly the unfavorable ripples (Gibbs phenomena) along the tails of the filter in both k_x and k_z directions. Since we were not successful in extracting the filter from its raw Fourier data in the correct order, I will not show the plot here. I believe that the plot was sufficient to give us a fair insight of the rippling behavior of the filter which might have unfavorable impact over ray tracing.

The way I will handle the discontinuity is by shifting the filter vertically down with an amount equal to the jump at $\rho = a$. Introducing this shift in equation (3.2.2) yields

$$f(\rho) = A\ [\ e^{-\frac{\rho^2}{a^2}} - e^{-1}\ ]\ H(a-\rho)\ , \tag{3.2.3}$$

and in discrete form,

$$f_{ik} = f(i\Delta x, k\Delta z) = A\left[e^{(i^2\Delta x^2 + k^2\Delta z^2)} - e^{-1}\right]\ , \tag{3.2.4}$$

where $i = -n_x, ..., n_z$, $k = -n_z, ..., n_z$ such that $n_x =$ greatest integer less than $(a/\Delta x)$, and $n_z =$ greatest integer less than $(a/\Delta z)$.

The normalized version of equation (3.2.4) is the final form of the Gaussian filter used in the code to smooth the speed of a model. Smoothing will be performed by convolving this normalized version with the sampled speed function in the spatial domain. A typical filter of the normalized version is shown in Figure 3.1. In the next the section I will give some idea about the code's important features as well as how the code works.

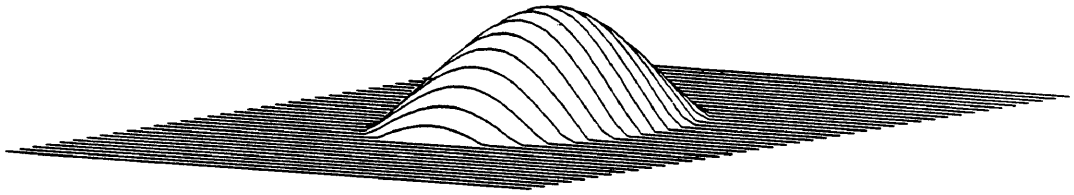

Figure 3.1: A plot of a typical Gaussian filter as used in the code.

3.3 Code Description.

The ray tracing algorithm in the code may be split into two phases of computations. Phase I starts by reading the model parameters provided by an external file (the parameter file). At this stage, the input model must consists of layers with constant earth parameters separated by arbitrary smooth interfaces across which these constants may change. The parameters will be used to build the background model for the inversion process. They include information about the number of interfaces in the model, number of control points (stored in a different file, the model file), propagation speeds (represented by a sequence of constant numbers), and the parameters which specify the spatial boundaries of the model.

The next step within this phase, will be a cubic spline fit to the reflectors using the control points (interface points) provided by the model file in order to guarantee continuous functions along the reflectors. At the control points , the spline interpolator will provide a perfect fit to the background interfaces while at other points it will be questionable. It will be shown later by a number of examples that the spline interpolator (at least the one we are using) provides a mathematical process that depends sensitively on the locations of these control points. Using the continuously interpolated reflectors, the speed is then sampled accordingly based on a rectangular grid geometry with sampling intervals specified by the user in the parameter file. The

rectangular grid could be made a square grid if we choose $\Delta x = \Delta z$. In our case, to make the smoothing filter a symmetric 2-D filter, the sampling intervals, Δx and Δz, must be equal.

The speed distribution within each grid cell depends entirely on the speed values at the four corners of the cell. The dependence is based on a 2-D bilinear interpolation that allows each corner point of the cell to contribute linearly to the interpolated speed according to its distance from the point along the ray trajectory. Following the sampling stage, speed is stored as a two dimensional array of data for a later recall and usage.

The code, next, reads the smoothing parameters from the parameter file and the smoothing stage starts by constructing the smoothing filter based on the filter radius specified in the parameter file. The filter is then convolved with the sampled speed function to obtain the smoothed version of the speed. The length of the filter is introduced according to the dominant wavelength in the seismic data to be imaged. We note that the code may read the radius parameter from the parameter file depending on whether or not we want to smooth the propagation speed. If the smoothing parameter is active, the filter radius must be specified in order to carry out the smoothing, otherwise no value for the radius need to be specified.

Phase II of the computations is the ray tracing stage which represents the main core of the code and which consumes most of the computational time spent during the

```
circle                          : name of file containing interfaces.
1                               : number of interfaces in the model.
10000. 12000.                   : model propagation speeds.
0. 2000.                        : model boundaries along x axis.
20.                             : speed sampling interval along x axis.
0. 2000.                        : model boundaries along z axis.
20.                             : speed sampling interval along z axis.
180.                            : dominant wavelength in seismic data.
y                               : optional flag for smoothing.
stl                             : file name of output unsmoothed speed.
stls                            : file name of output smoothed speed.
stl                             : file name of input unsmoothed speed.
stls                            : file name of input smoothed speed.
n                               : flag for 3-D plot of unsmoothed speed.
n                               : flag for 3-D plot of smoothed speed.
1                               : number of take-off angle ranges.
-32.                            : starting take-off angle.
32.                             : final take-off angle.
.5                              : take-off angle increment.
0.                              : initial ray parameter value.
0.                              : initial traveltime value.
0.                              : initial angular derivative of ray offset.
1000. 100.                      : source offset and source depth.
850.                            : maximum tracing depth allowed.
4.                              : depth increment.
4.                              : initial adaptive depth increment.
.0001                           : minimum depth increment allowed.
200. 1800.                      : left and right allowed tracing offsets.
1.e-10                          : minimum alowed vertical slowness.
1.e-3                           : accuracy tolerance.
850.                            : depth of the computed amplitude.
y                               : rays plot flag using unsmoothed speed.
y                               : rays plot flag using smoothed speed.
```

Figure 3.2: Typical parameter file as used in the code.

execution. At this stage, the code starts to read the ray parameters which control the initial conditions of the wave field at the source location. This data include the starting location (source or receiver) of the rays, number of take-off angle ranges, maximum and minimum angles within each angle range, extreme left and right tracing offsets that a ray can reach in the model provided that a non-zero value for the vertical slowness did not occur, maximum depth that a ray can reach provided that a non-zero value for the vertical slowness did not occur, depth increment of tracing, etc. Figure 3.2 shows a typical parameter file as used in the code.

Therefore, given the source location with a set of field initial values along with a specified take-off angle, rays are shot until one of three things may happen. The ray keeps bending until the vertical slowness becomes zero, the ray reaches the maximum left or right tracing offsets, or the ray reaches the maximum tracing depth specified in the parameter file. For all these three cases, the code will stop tracing and move to the next ray by increasing the take-off angle using the angular increment specified in the file. The code allows more than one angle range to be considered for tracing the rays from a single shot. By breaking a single angular range into two ranges, different ray densities in adjacent sectors can be achieved. I find this parameter to be a convenient option that can save time and storage when a large number of rays are needed and when a dense set of take-off angles in a particular direction is suggested.

The ray tracing routine reads initial ray location and travel direction from the file " raypar " (the parameter file name) and computes initial values of p, q, and $\partial p/\partial\beta$.

A Fourth order Runge-Kutta method is used to compute the new ray quantities at the new depths in the model. As the ray marches from one depth to another, tracing is carried out in an adaptive stepping procedure where the ray quantities have to satisfy a success criterion before they are accepted as good values at the new depth level. This criterion involves a given error tolerance provided by a flag in the parameter file. The tolerance specifies the degree of accuracy desired in computing the new ray quantities. When this accuracy is reached, the program accepts the new quantities and proceeds to treat them as initial conditions for the next depth subinterval (adaptive depth increment) within the original fixed depth increment.

The most important part of the code is a combined performance of two subroutines that together constitute the segment in the code which is responsible for the adaptive tracing behavior as rays march from one depth to another. These subroutines were taken from Press et. al. in (1987). The first subroutine specifies a new depth level using the depth increment in the parameter file. It prescribed this new depth and the initial values at the previous depth and passes this information to the next subroutine. The second subroutine tries to march rays (within the depth increment) using an adaptive increment found by an empirical formula (Press et. al., 1978). The ray marching continues until the success criterion is satisfied. If the criterion is not satisfied the adaptive increment is decreased again (also according to the same empirical formula) until the criterion is again satisfied. When the prescribed depth is reached, the subroutine will return the new values to the first subroutine so to

be used for a new marching of rays. When there is no success in reaching the prescribed depth, the code switches to the next ray.

Figure 3.3 shows a simplified flow chart of the code algorithm. The top box includes slowness initial angles, starting depth, and initial ray quantities. The information is passed to a another subroutine which performs stepping in depth to advance the ray quantities to deeper levels into the model. After a depth level is reached by previous steppings, the adaptive procedure is applied again to determine the solution at a new depth level. During each sublevel trial, the trial subroutine passes the sublevel depth and its ray quantities to the adaptive stepping subroutine where the data is tested against a success criterion.

The adaptive stepping is based on achieving the final depth, within one depth interval, using two different computational paths. Along one path, the depth increment is divided into two equal increments and ray quantities are then computed at the final depth by double stepping. For the other path, the final depth is achieved using a single full depth increment where ray quantities are computed just once. The difference between the two values obtained by these two different paths is computed and scaled by the starting values of ray quantities along with their associated derivatives. The maximum difference is computed and divided by the prescribed accuracy tolerance specified in the parameter file. If the division output is greater than one, the criterion is not satisfied and the increment is reduced again in order to satisfy the criterion again. If the criterion is satisfied, the new values at the successful sublevel are accepted and a

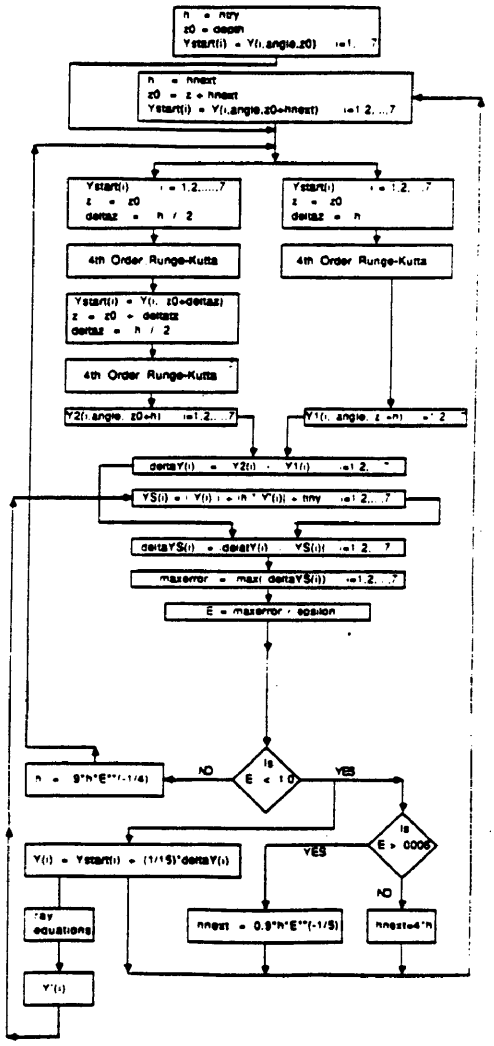

Figure 3.3: A flow chart of the algorithm used in the code.

new depth sub-increment is proposed for a deeper marching within the fixed depth increment. The process continues until the terminal depth of the fixed increment is reached with successful values.

The accuracy of the routine was tested by comparing the code output for a linear depth dependent speed model of the form,

$$c(\mathbf{x}) = 1 + 10z, \tag{3.3.1}$$

and the analytical solution obtained using the same model. For this type of model, rays trajectories are circles. Figure 3.4 shows a plot of rays using the above speed distribution. Tables 3.1 and 3.2 illustrate a comparison between the code output and the analytic results for this type of model. An accuracy of five decimal places agreement between the two outputs was found to be the case at some depth levels. At other depth levels the accuracy reduces to four decimal places and even three.

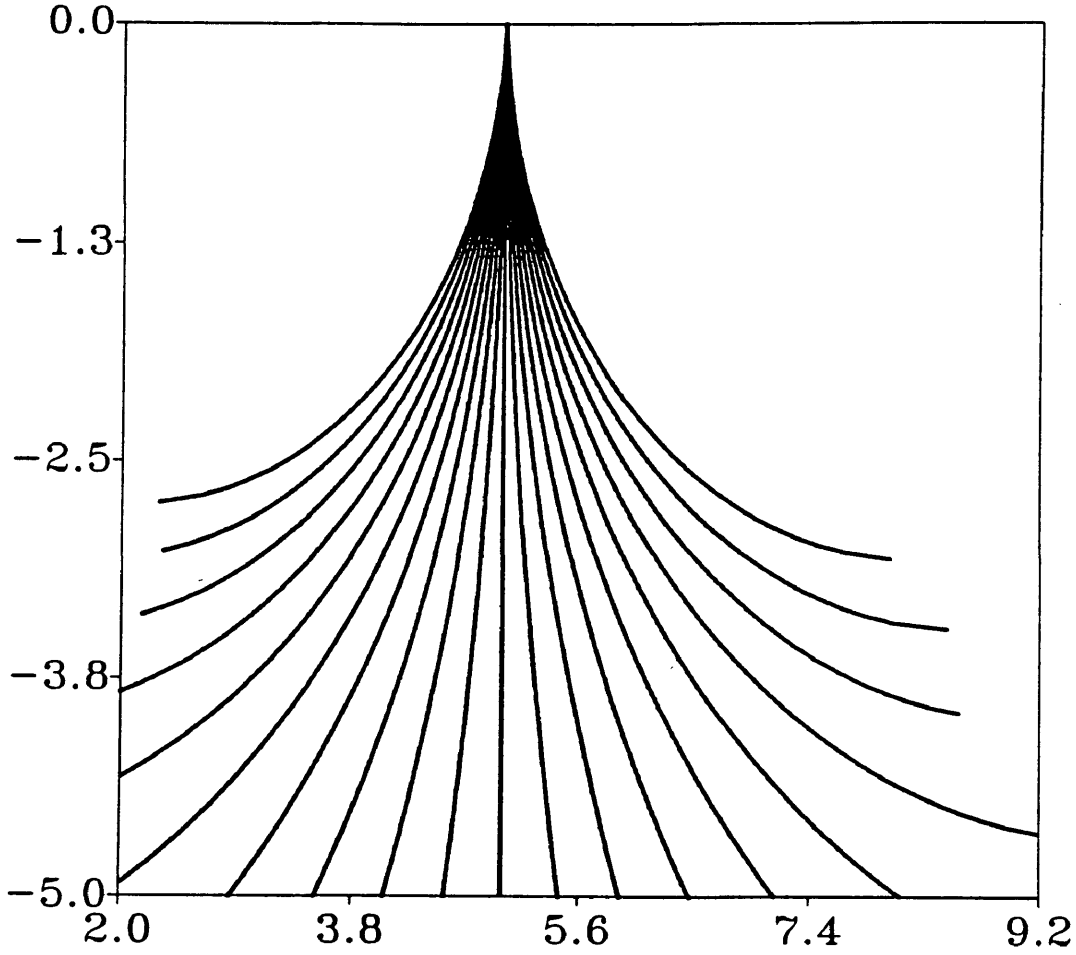

Figure 3.4: A plot of rays using the speed function c = 1 + 10 z. In this case, rays represents parts of circles.

Table 3.1: This table compares traveltimes computed by the code to traveltimes obtained by the analytic solution using the speed distribution $c = 1 + 10\ z$. The take-off angle in this case is -1.8 degrees and the initial time and depth were both zeros..

DEPTH	ANALYTIC FORMULA	CODE OUTPUT
1.88	.310024	.310025
1.92	.312595	.312595
1.96	.315149	.315149
2.00	.317689	.317690

Table 3.2: This table compares $dx/d\beta$ computed by the code to $dx/d\beta$ obtained by the analytic solution using the speed distribution $c = 1 + 10\ z$. The take-off angle is also -1.8 degrees and the initial time and depth were both zeros.

DEPTH	ANALYTIC FORMULA	CODE OUTPUT
1.88	28.1138	28.0138
1.92	29.8119	29.7118
1.96	31.6041	31.5147
2.00	33.5317	33.4316

3.4 Ray Trajectory Examples.

A ray-theoretic modeling output using a smoothed version of a discontinuous speed can be obtained using relevant parameters that are provided by a parameter file. The file can read any sampled 2-D speed function directly from an external file. This flexibility permits the use of any general 2-D speed background in place of the stratified constant speeds model often used in practice. This option can be chosen by shutting down the smoothing parameter through a flag provided in the parameter file and the code will proceed to read the arbitrary speed from an external file.

Throughout the modeling process using smoothed versions of practical models, one can learn much about WKBJ modeling in general. An important example is the problem connected to the high sensitivity of ray trajectories to small artifact perturbations along interfaces. It will be shown by various examples that ray trajectories show great sensitivity to these small perturbations introduced by the interface spline interpolator. This is true whether the perturbations were due to true changes in the model or due to artifacts of the modeling process (e.g. cubic spline).

Figure 3.5 is a model of a perfect horizontal reflector. The cubic spline interpolator provides a perfect fit to the reflector since all control points (points used by spline interpolation to construct interfaces) are at the same depth. This should minimize the presence of any artifacts that may be introduced by interpolation. Speeds

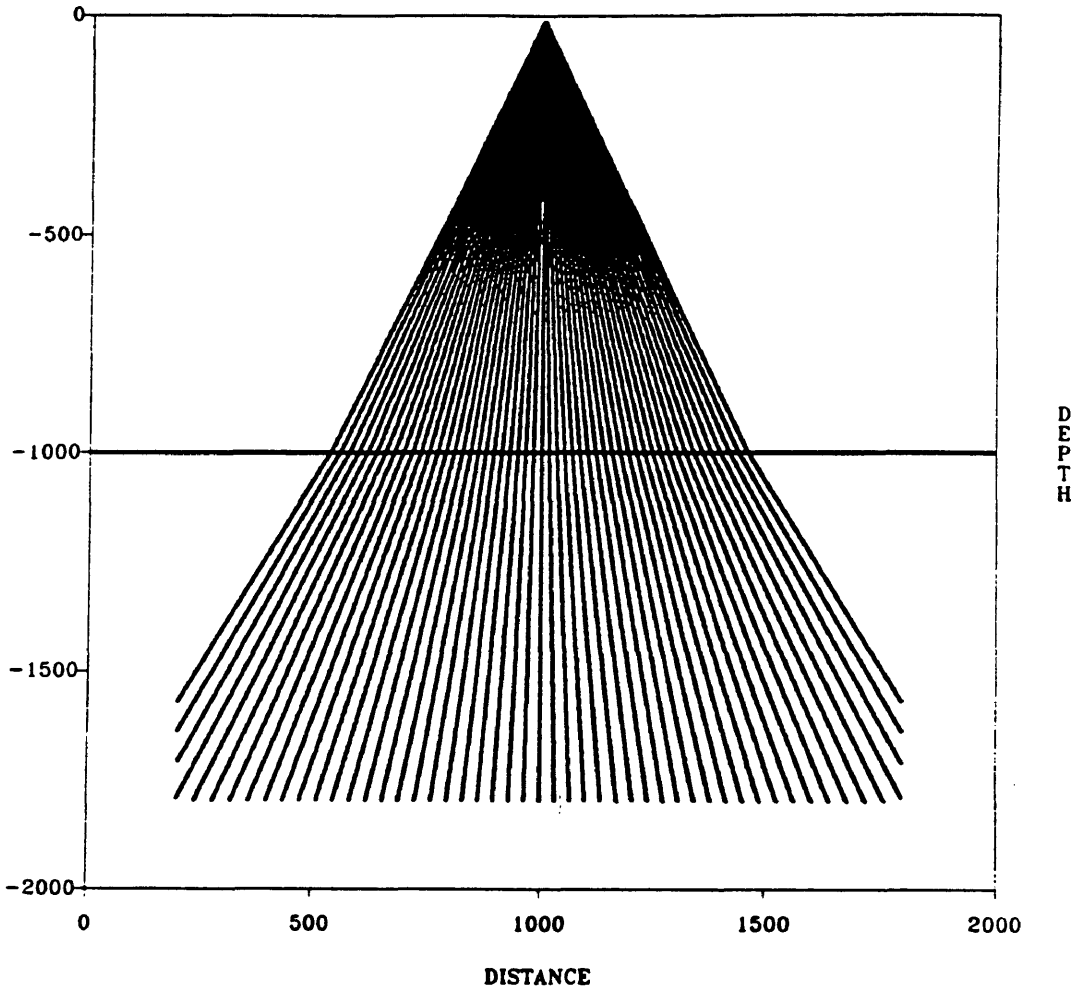

Figure 3.5: A plot of rays through a horizontal interface. The cubic spline interpolator provides a perfect fit through control points. In this case, the presence of artifacts is minimized. This illustrates ability of rays to access near vertical regions when artifacts are absent.

used here were 10000 ft/sec and 12,000 ft/sec. Ray take-off angles range between -25.0 and 25.0 degrees with an increment of 1.0 degree. The source is located at 10 ft deep and 1000 ft offset. Rays here tend to behave normally as they cross the reflector. All rays in this case penetrate successfully to near vertical regions below the reflector as they refract across. It should be noted here that the speed jump across the reflector is not a perfect step function. Rather it is a linear ramp of width $\Delta z = 20ft$ that connects the two values of the speed at each layer in a linear fashion according to a 2-D bilinear interpolation (Press et. al 1988).

In order to see the effect of a small perturbation along the reflector on rays distribution, a small semi-circular feature was introduced at the center of the reflector. The semi-circle looks like a wedge more than a circular shape. This is due to the size of speed sampling intervals which were coarse (20ft) as compared to the semicircle size. The ratio of the feature to a typical model dimensions $(O(10^4))$, is $O(2\times10^{-3})$. This ratio is very small and should not change the medium characteristics significantly. On the other hand, the feature causes a considerable deflection of the central rays as seen in Figure 3.6.

At greater depths than the one shown in the last figure, the separation between the central rays increases and the sensitivity becomes an apparent problem, since the inaccessible regions become larger in size at greater depths. The region can become large enough to include important subsurface structures below the reflector which could be missed during the modeling process. In addition, the computations of ray

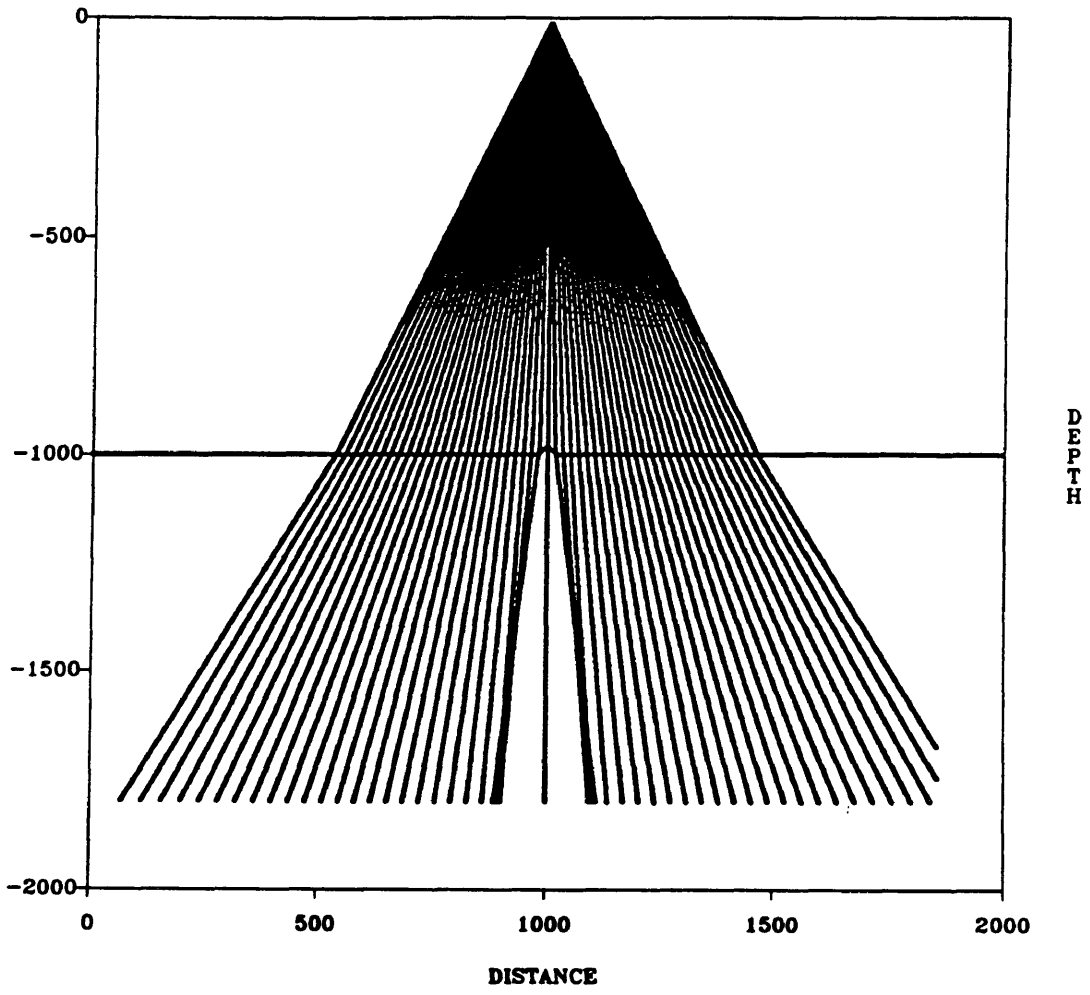

Figure 3.6: Plot of rays across the perturbed reflector. Bending of the central rays across the feature illustrates rays sensitivity due to small details along the interface. The feature produced a ray gap below the reflector. This ray gap is inaccessible using only the above rays (except the zero incidence ray).

amplitudes will add another dimension to the problem since it is well known that ray amplitudes are highly sensitive to the model small details (Cerveny, 1985).

After applying smoothing using a Gaussian filter of radius 120 ft the perturbation almost disappeared and the interface becomes nearly a perfect horizontal zone. In this case, the discontinuity (the ramp) has been replaced by a smooth transition zone between the layers. Figure 3.7 illustrates the effect of applying smoothing on the behavior of the central rays. The impact of small perturbation on rays has been considerably diminished after smoothing and rays tend to have a good access to the ray gap in Figure 3.6.

The model of Figure 3.6 consist of a small perturbation that we have introduced to illustrate the sensitive response of rays to these fine details. We now consider an example in which a number of these small perturbations are present along the interface. To do so we consider a similar model to Figure 3.6 with a bigger semi-circle having a radius of 80 as shown in Figure 3.8. At this point we are not interested in the semi-circle. Rather we are interested in the small details along the horizontal part of the reflector to illustrate their effect on rays trajectories. Size of semi-circle will cause spline interpolator to show these artifacts in latter examples. In this case the shape of the feature is getting closer to a perfect semi-circle on the scale of the sampling interval, 20 ft. Rays here tend to behave in a fashion similar to that in the last example and the only difference is that a larger number of rays are deviated because of the larger scale of the anomaly.

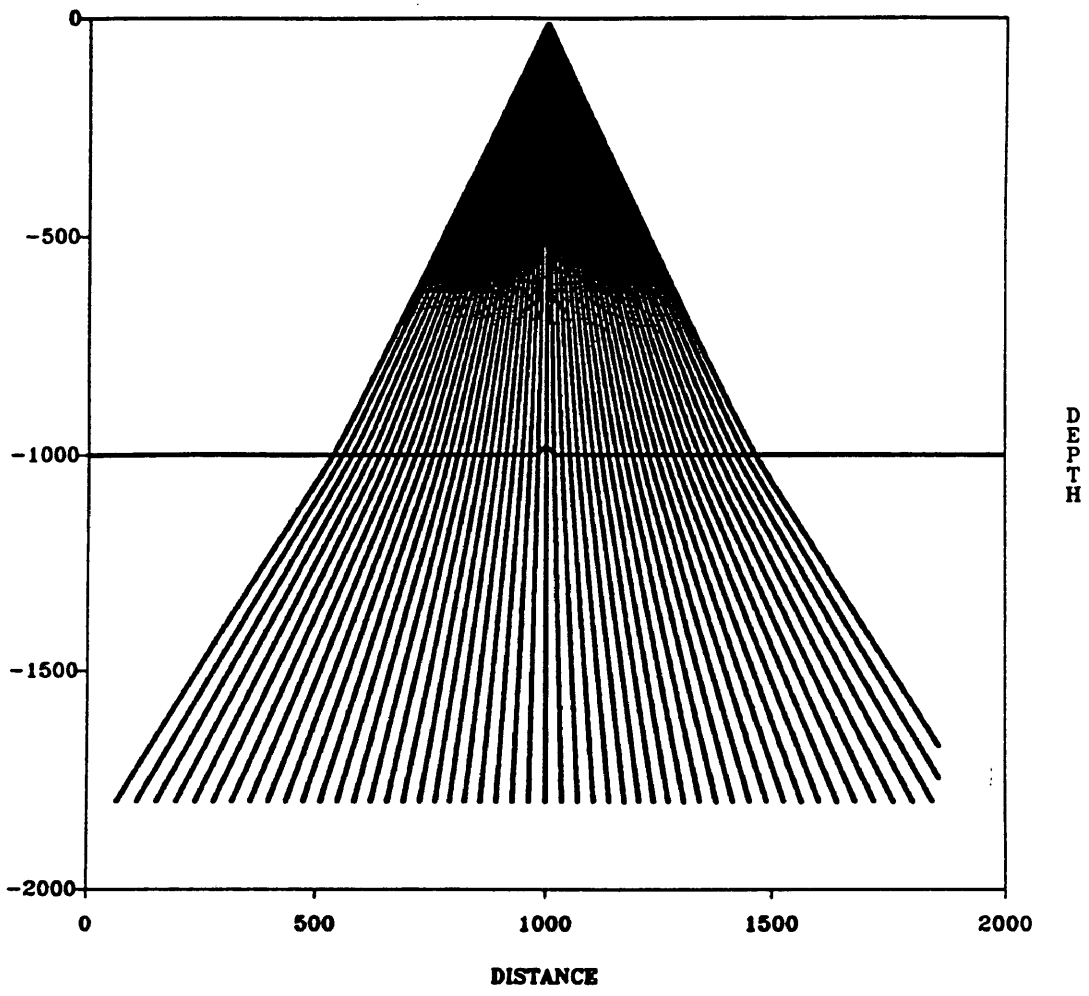

Figure 3.7: Rays here tend to have a good access to the ray gap in Figure 3.6. Gaussian smoothing was effective in removing the perturbation effect.

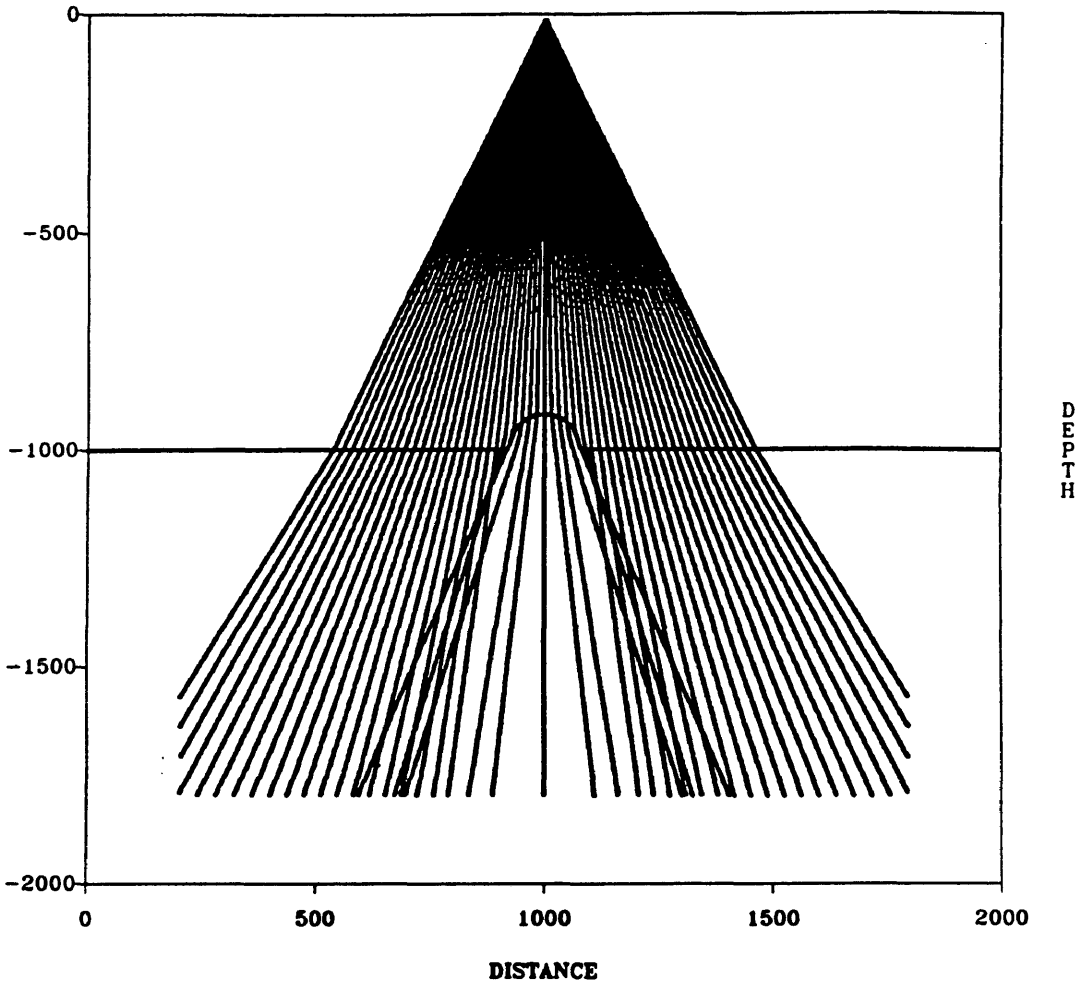

Figure 3.8: Plot of rays across an interface with a semi-circle of radius 80 ft. Across horizontal sides, rays behavior does not reflect significant presence of artifacts.

Figure 3.9 shows the speed distribution at a sampling intervals of 20 ft. Figure 3.10 shows the smoothed version of Figure 3.9. It is apparent here, that the Gaussian filter tends to minimize the perturbations in a similar way to a weighted sums since the filtering is performed via convolution. Compare small features along the semi-circle in both cases. Figure 3.11 illustrates the effect of smoothing over rays previously shown in Figure 3.8 when a Gaussian filter of radius 120 ft is used. In this case, ray bending due to interface curvature was reduced and rays refract less and remain more vertical. This is one of the effects that must be accepted in the modeling process when smoothing is applied. Note the more uniform ray distribution below the semi-circle.

It should be noted here that our objective in considering a similar model is only to cause the interpolator (cubic spline) to produce a number of small perturbations by either changing the speed sampling intervals or by changing the number of control points along the interface. Figure 3.12 shows a similar model to that of Figure 3.9 but now with speed sampling intervals of 5.0 ft. The small hidden perturbations in Figure 3.9 start to appear now in Figure 3.12 as we use finer sampling intervals. Figure 3.13 illustrates the effect of refining the speed sampling intervals on ray distribution. This indicates a high potential for artifacts presence when sampling intervals are refined. In other words, it illustrates the significance of choosing the speed sampling intervals when tracing the rays. Figure 3.14 illustrates the effect of smoothing in this case. Gaussian filtering here was effective in minimizing the artifact distribution of rays which can be seen when comparing the two figures.

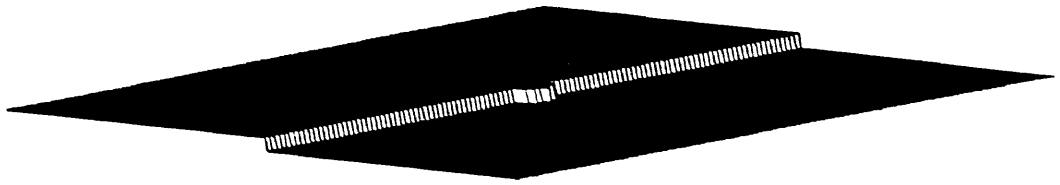

Figure 3.9: A plot of the discretized speed distribution of Figure 3.8. Note the prominent presence of irregularities along the middle feature.

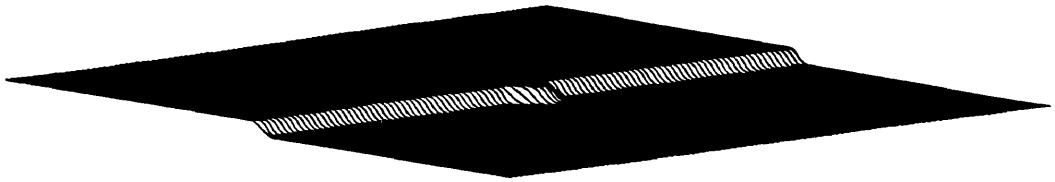

Figure 3.10: This is the same model as in Figure 3.9 but after smoothing. In this case, the filter radius was 60 ft. Perturbations in Figure 3.9 were minimized especially along the semicircle.

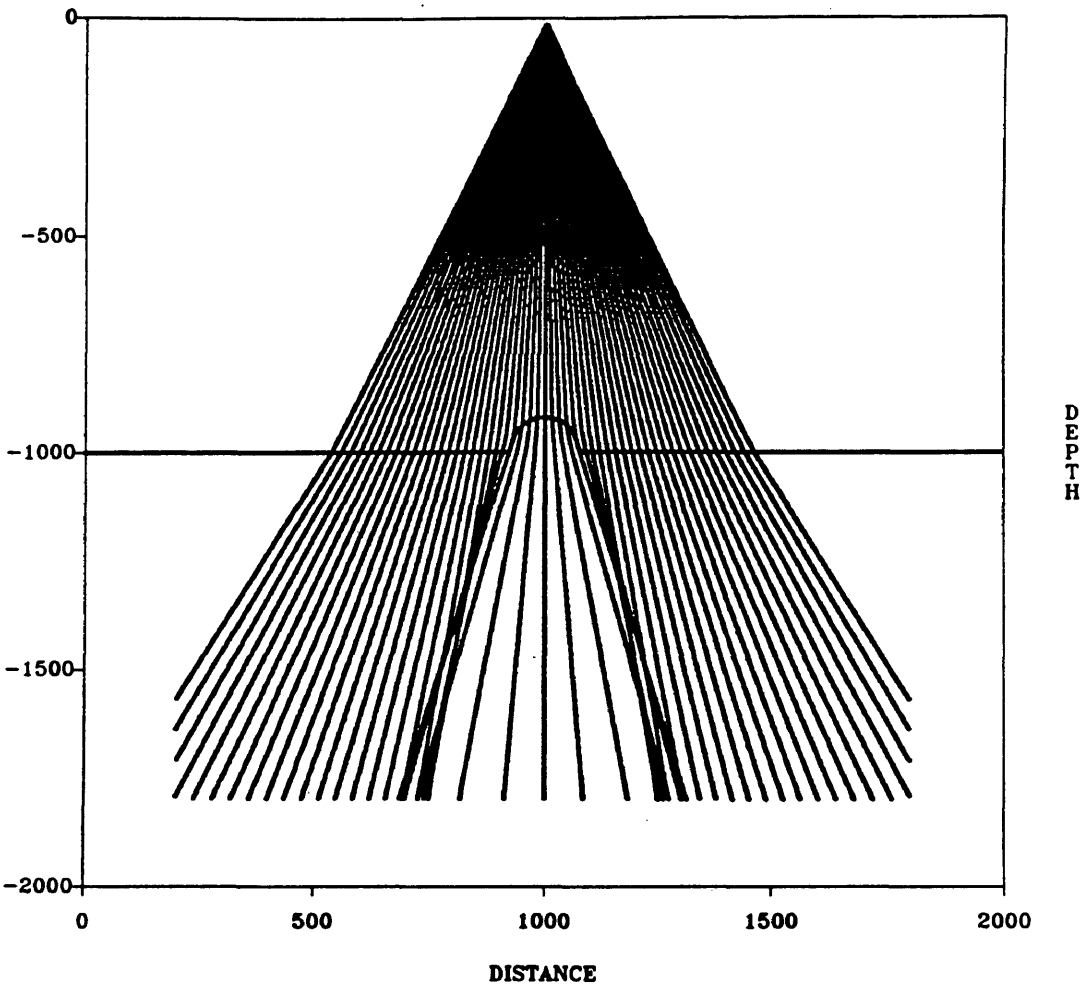

Figure 3.11: Rays using the model shown in Figure 3.10. Rays here have the tendency to be less sensitive to the interface curvature. This is one of the smoothing side effects that can not be avoided in smoothed ray tracing.

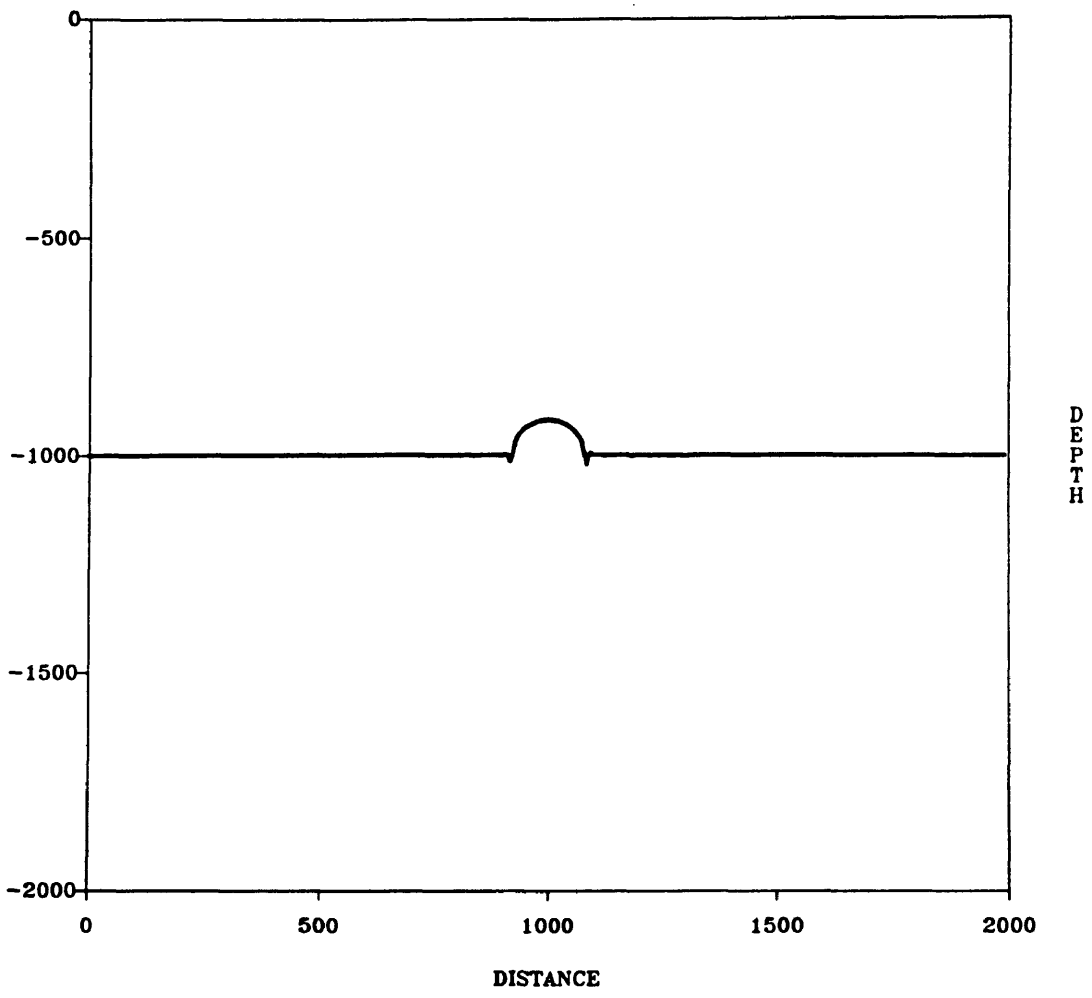

Figure 3.12: This is the same model as in Figure 3.9 after refining the speed intervals from 20 ft to 5 ft. More artifacts has been introduced to the model in this case especially along the horizontal part.

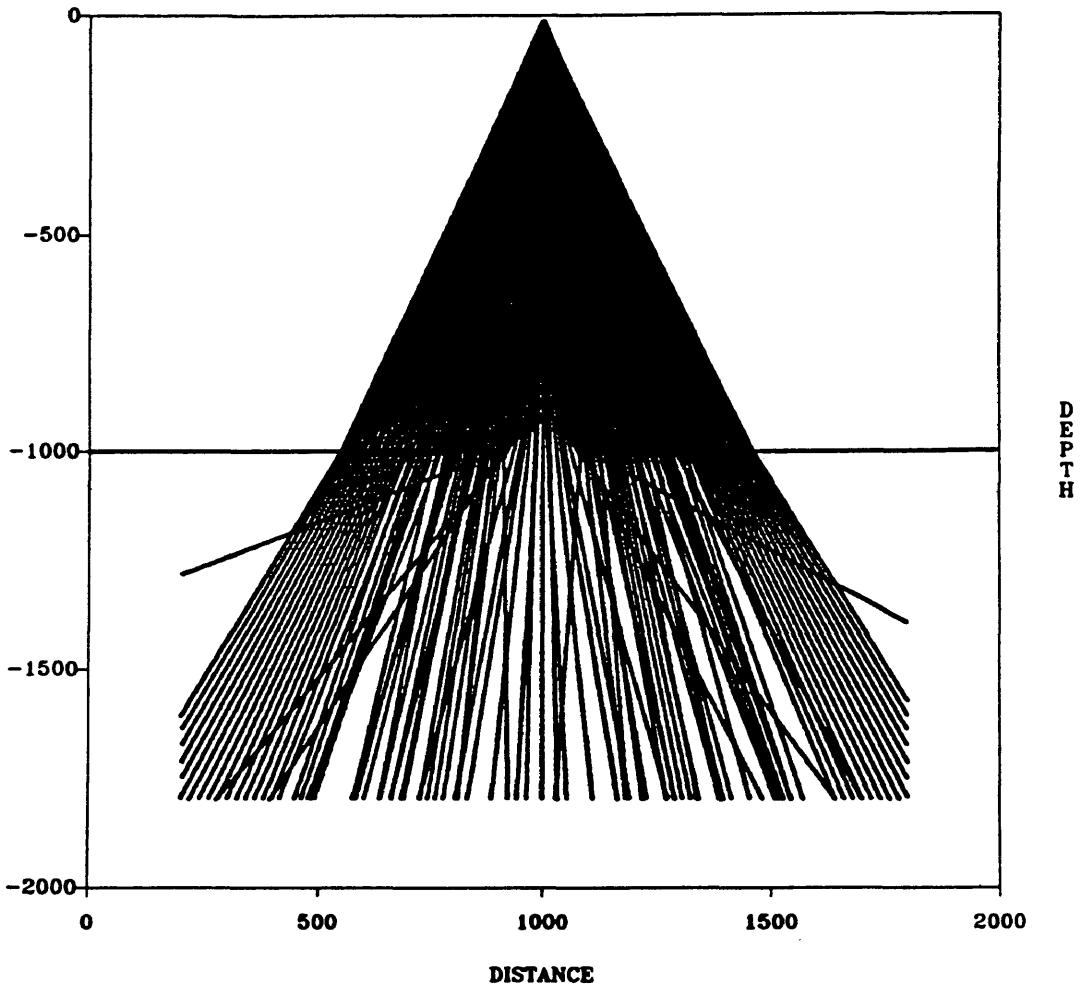

Figure 3.13: Rays traced using the speed distribution in Figure 3.12. Angle increment used here is 0.5 degrees. Rays exhibits great sensitivity due to the artifacts along the interface.

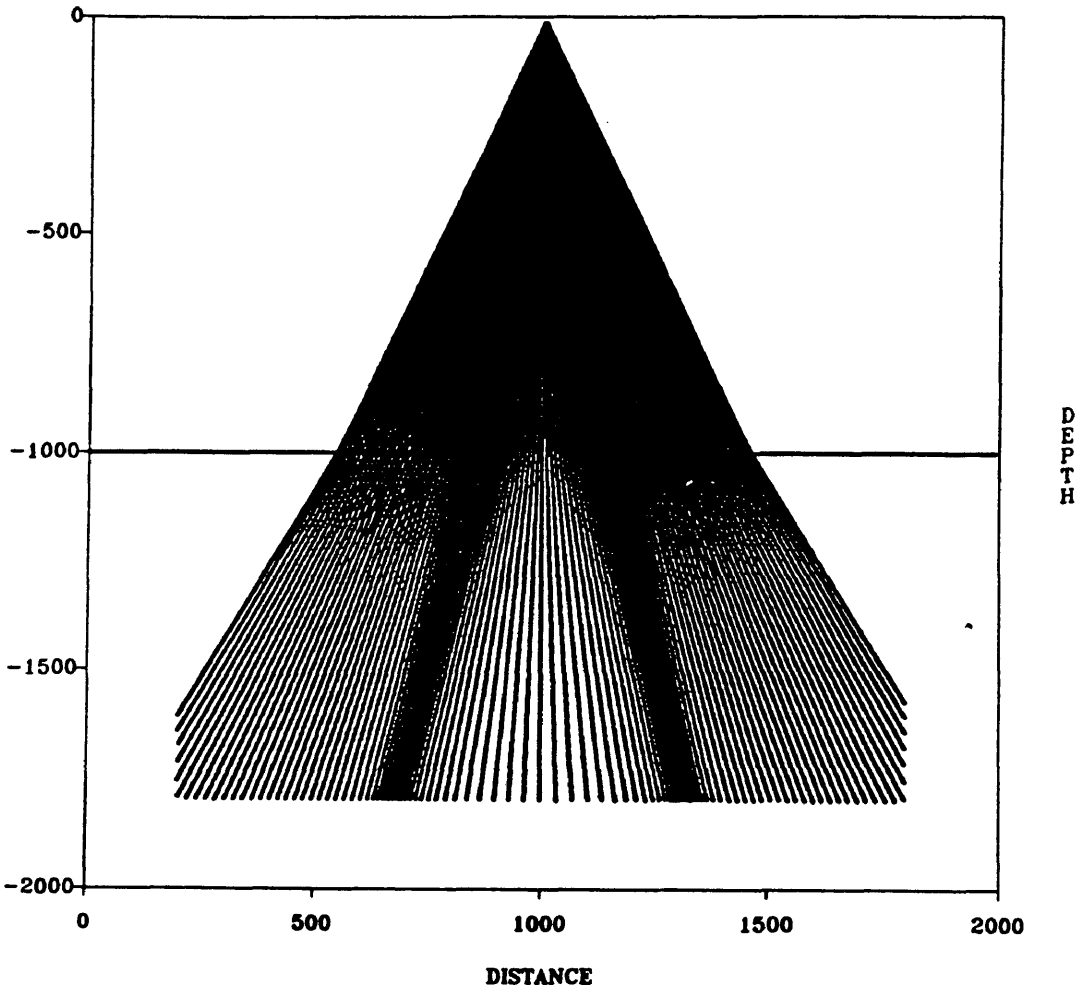

Figure 3.14: A plot of rays using a smoothed version of Figure 3.13. Angle increment is 0.5 degrees. Rays in this case are more uniformly distributed below the interface.

Figure 3.15 is a plot of the same model shown in Figure 3.8 but with fewer control points along the interface. This figure is an example of a poor interpolator performance because the number of control points has been reduced from 217 to 64. The small irregularities along the reflector can easily be seen when the figure is viewed from the plane of the paper. The figure also depicts the ray distribution in this case. One can see that the distribution of rays is much more erratic than in Figures 3.8 and 3.11, demonstrating the sensitivity to irregularities in the interface description. Figure 3.16 shows the same rays after smoothing was applied with radius of 60 ft. To appreciate more the significance of the sensitivity problem, see Figure 3.17 where rays are traced to greater depths in the model. Note the relatively large ray gaps that can be introduced when these irregularities are present along the interface. Figure 3.18 shows again smoothing effect when a Gaussian filter of radius 60 ft is applied.

Figure 3.19 represents a scaled down laboratory simulation of a real seismic survey (physical model). The data for this model was provided to us by Marathon Oil Company. A source located at 10 ft depth with offset 16,000 ft has been used to generate an acoustic wave. Velocities used in the model are (top to bottom) 19000, 16000, 21000, 15000, and 19000 ft/sec. The WKBJ modeling using the above propagation speeds is shown in Figure 3.20. Ray behavior shows a high sensitivity due to small irregularities along interfaces. Part of these small perturbations are due to true changes of the model and part are due to the artifacts introduced by the interpolator. Take off angles range in this case from -25.0 to 25.0 degrees using 1.0

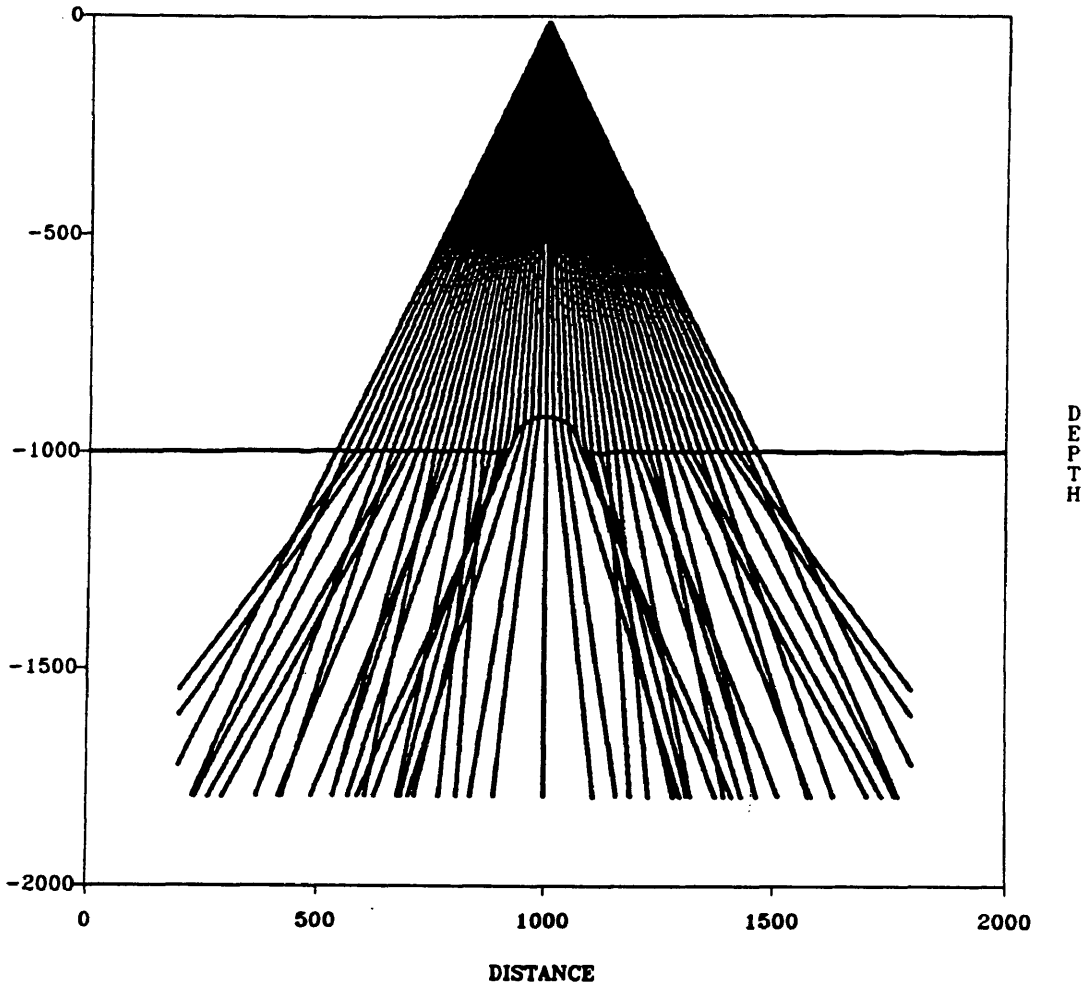

Figure 3.15: A highly sensitive ray behavior due to the presence of perturbations caused by a poor performance of spline interpolation. This is indicated by the many crossings of the refracted rays.

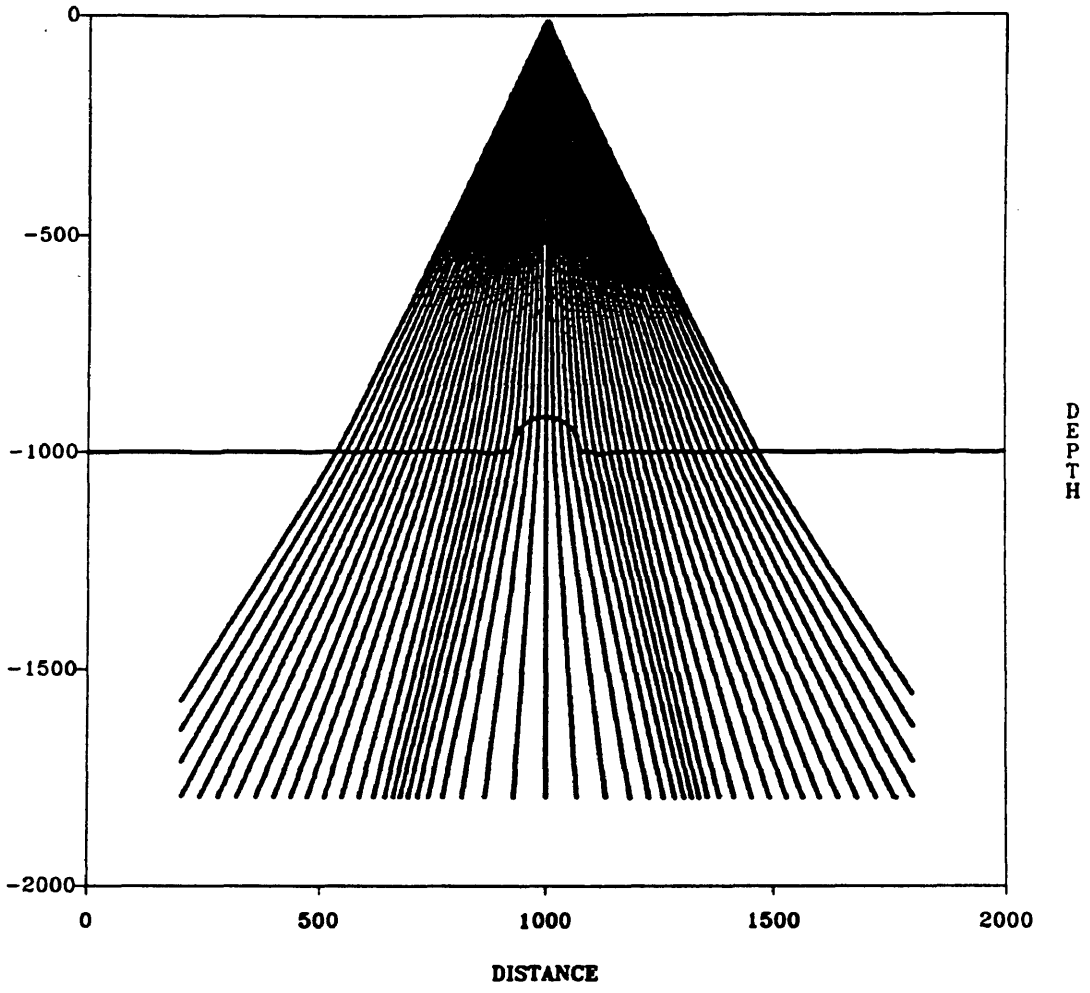

Figure 3.16: Plot of rays using a smoothed version of Figure 3.15. The filter radius used in this case was 60 ft. Note the uniform distribution of rays below the interface. This tracing could be a more reliable estimation of the field for inversion purposes than tracing in Figure 3.15 may do. Crossings in this case has been removed.

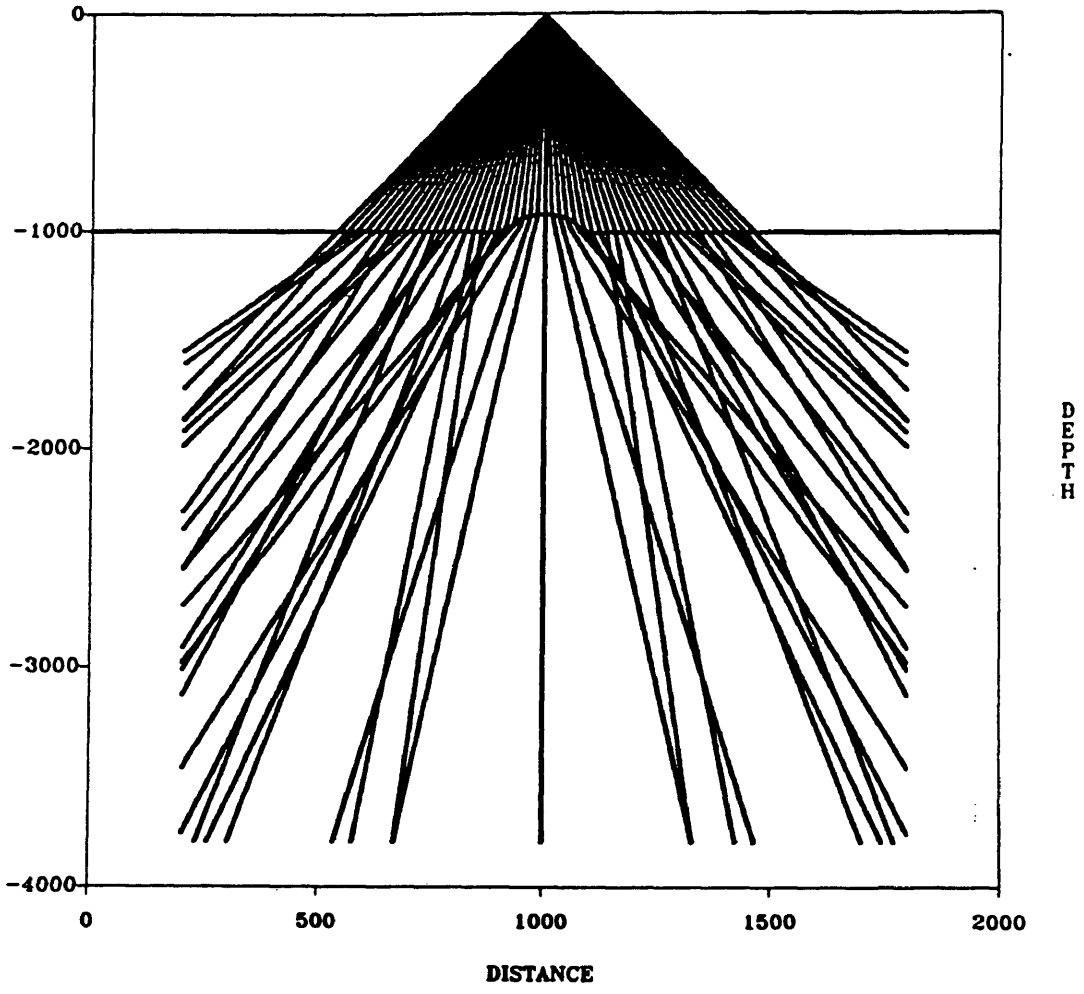

Figure 3.17: At greater depths the sensitivity becomes an apparent problem. Note the relatively large ray gaps introduced by the small artifacts along the interface.

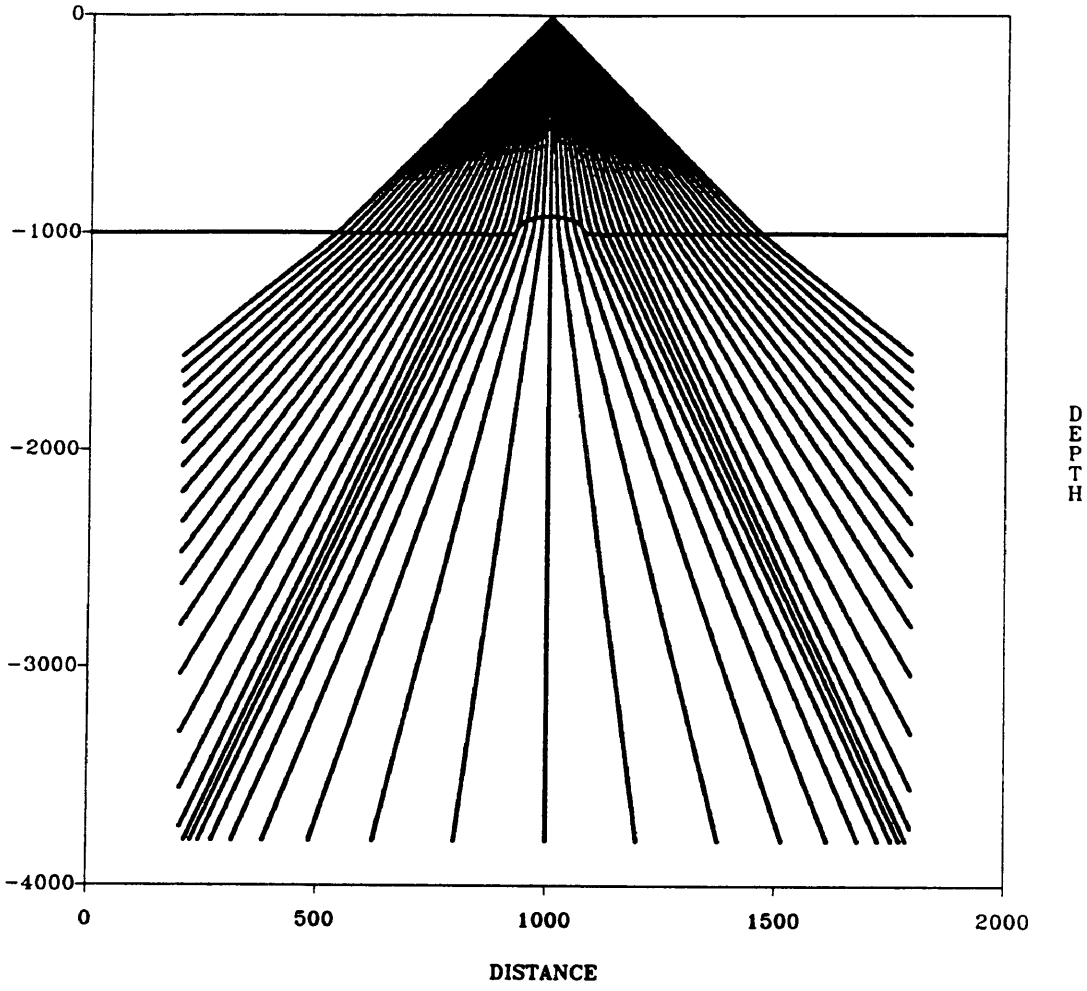

Figure 3.18: This is a plot of rays using the smoothed version of Figure 3.17. A more uniform ray distribution has been attained. The size of ray gaps in Figure 3.17 is minimized after smoothing.

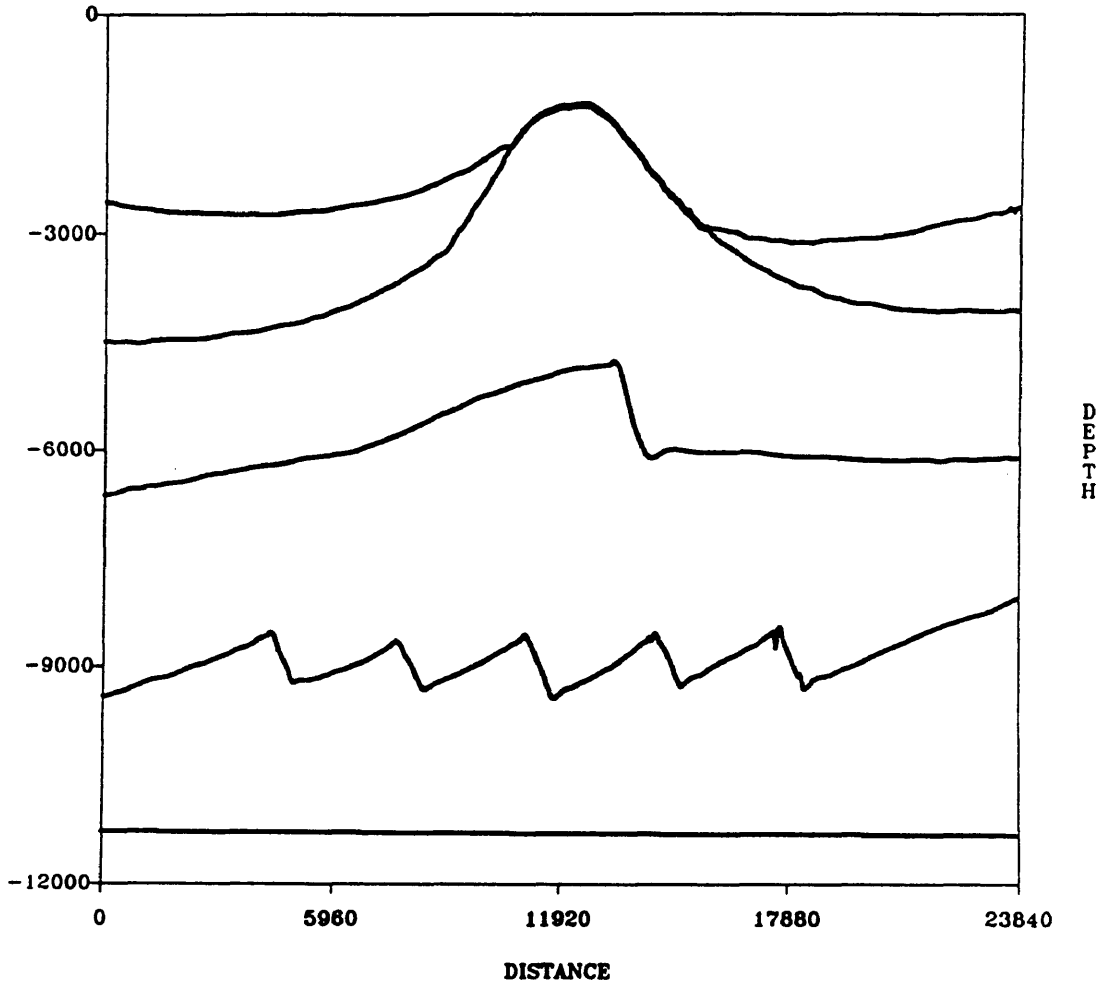

Figure 3.19: Marathon Oil Company model. Large number of artifacts can be introduced in practical models.

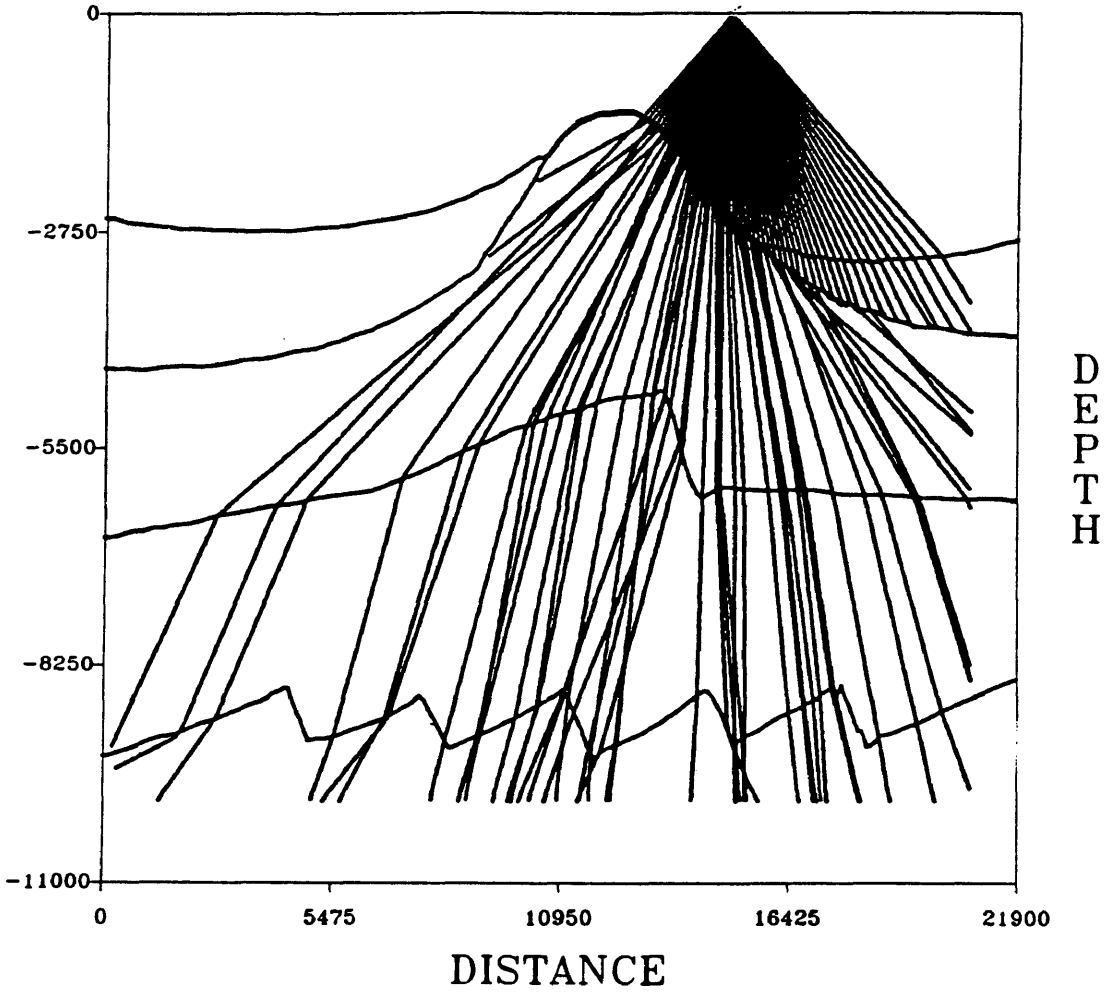

Figure 3.20: A plot of rays using unsmoothed Marathon model. The source is located at depth 10 ft and offset 16000 ft. Rays behave sensitively due to the presence of a large number of artifacts along interfaces.

degree increment.

The first sawtooth from the left and the second sawtooth from the right were not illuminated sufficiently by rays which could cause unreliable estimation of the field within that region. Also note that the presence of the fault along the third reflector has forced the middle rays to deflect considerably because of the sudden increase in the incidence angle along the reflector as the take-off angle change. This is due to a sudden change in the local normal along the reflector. Figure 3.21 shows ray trajectories after smoothing was applied. Smoothing here has produced a more uniform distribution of rays, especially over the second sawtooth from the right. The sawtooth in the extreme left was not illuminated sufficiently because the range of take-off angles here was not big enough to cover the extreme left area of the model.

We look now at the (side) effects that could arise when applying smoothing prior to ray tracing. One of these effects is connected to high speed contrasts across interfaces. High speed contrasts represent natural barriers that usually prevent near vertical penetration of rays (except for small incidence angles). Figure 3.22 illustrates this fact where a simple experiment was carried out using two different sets of speeds. This tracing experiment was conducted to see (practically) how rays react to two different speed contrasts across the same reflector as well as the effect of smoothing on this reaction. The first speed set is 10000 ft/sec and 12,000 ft/sec for top and bottom layers respectively, while the second set is 10000 ft/sec and 20,000 ft/sec for the same layers. This simple experiment consists of two experiments superimposed in one plot.

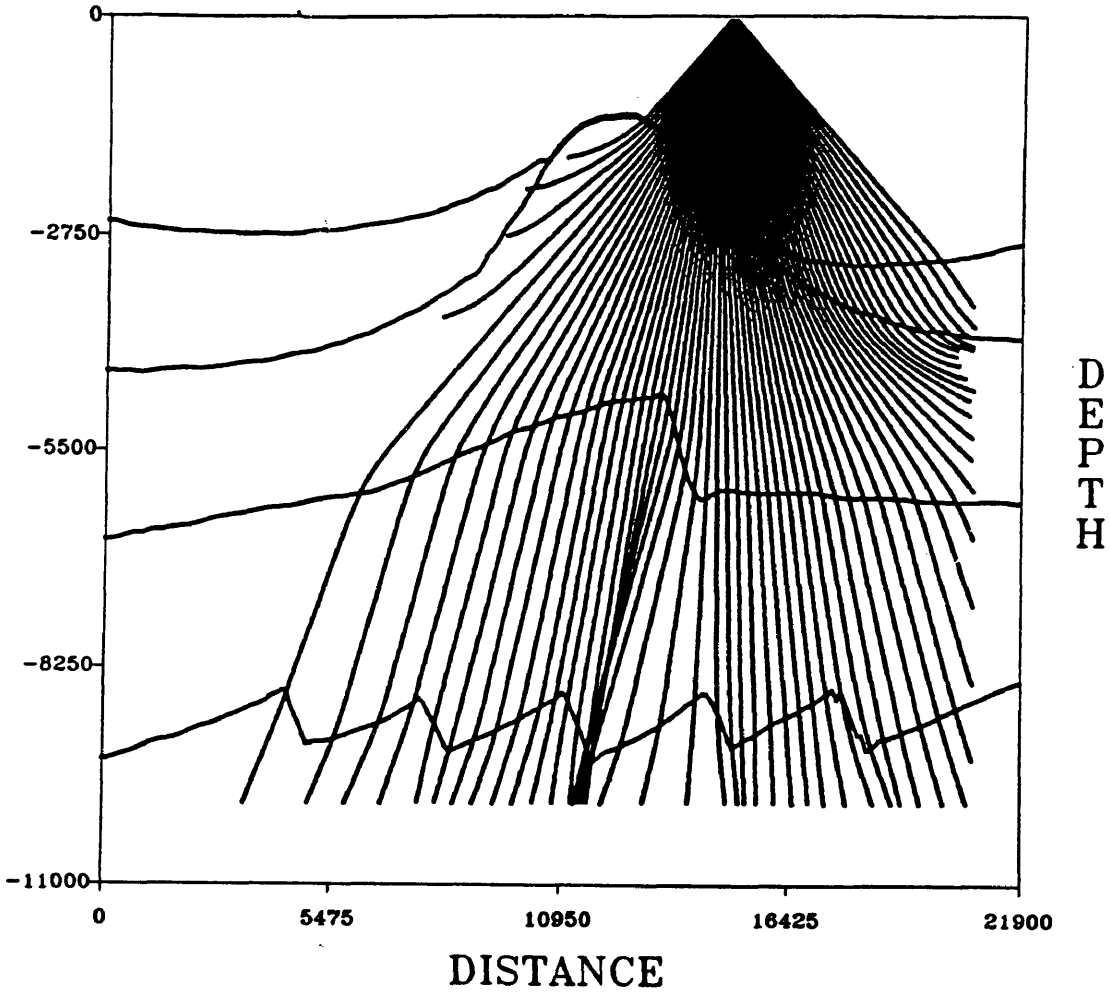

Figure 3.21: A plot of rays using the smoothed version of the Marathon data. The radius of the filter used for smoothing was 120 ft. Smoothing was effective in producing more uniform ray distribution than in Figure 3.20.

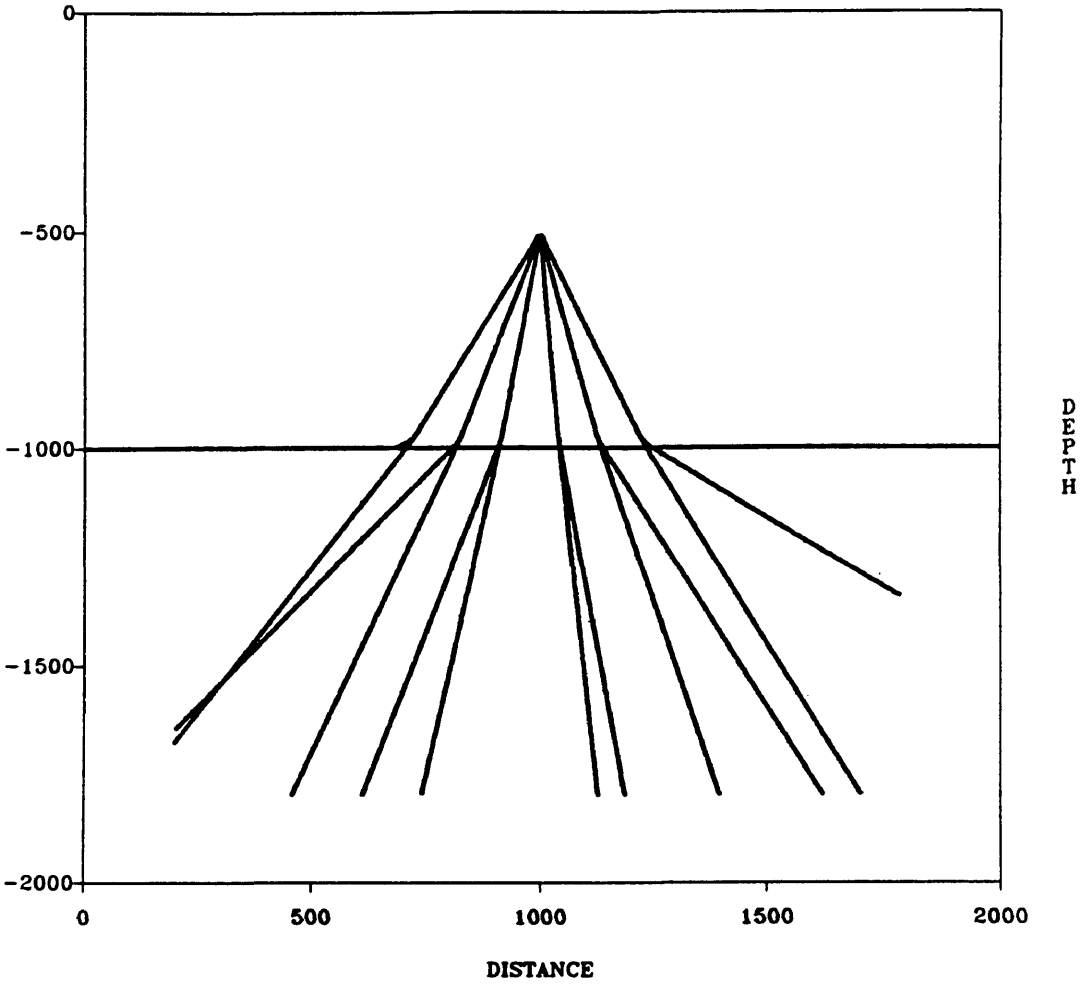

Figure 3.22: A plot of rays using two different speed distributions. In the top layer, rays from the two experiments overlap because the same speed has been used. Rays in the bottom layer discriminate because different propagation speeds has been encountered. This shows that high speed contrasts represent natural barriers that prevent near vertical penetration especially when incidence angle is large.

Note that the rate at which rays approach critical incidence as a function of the take-off angle depends on the speed contrast across the interface. The higher the contrast, the higher is this rate.

Figure 3.23 shows the smoothing effects on rays across the smoothed interface. They tend to penetrate more smoothly to the lower layer than they do in the example above. The discrimination angle in the lower layer between rays of the same incidence angle seems to decrease especially at higher incidence angles. This is due to the simple observation that smoothed rays tend to penetrate more towards the local normal than unsmoothed rays do. Figure 3.24 shows this smoothing effect in a more pronounced depiction where we can see clearly the smoothed ray has penetrated towards the vertical more than the unsmoothed one has. When the source is closer to the interface, the difference is more apparent as shown in Figure 3.25. Smoothing seems to weaken these natural barriers produced by high speed contrasts and allows more ray penetration towards vertical directions than unsmoothed models do.

A second side effect is that large sudden changes in interface slope (e.g. faults) may cause significant deflection of refracted rays. Figure 3.26 shows a faulted horizontal reflector where the central rays across the fault deflected significantly from their directions at incidence because of the high speed values at the left side of the fault. Such a deflection causes some pitfalls in modeling the structure below the interface because the deflected rays produced a ray gap that may result in an unreliable field estimation there. Figure 3.27 shows how smoothing forced the deflected rays to

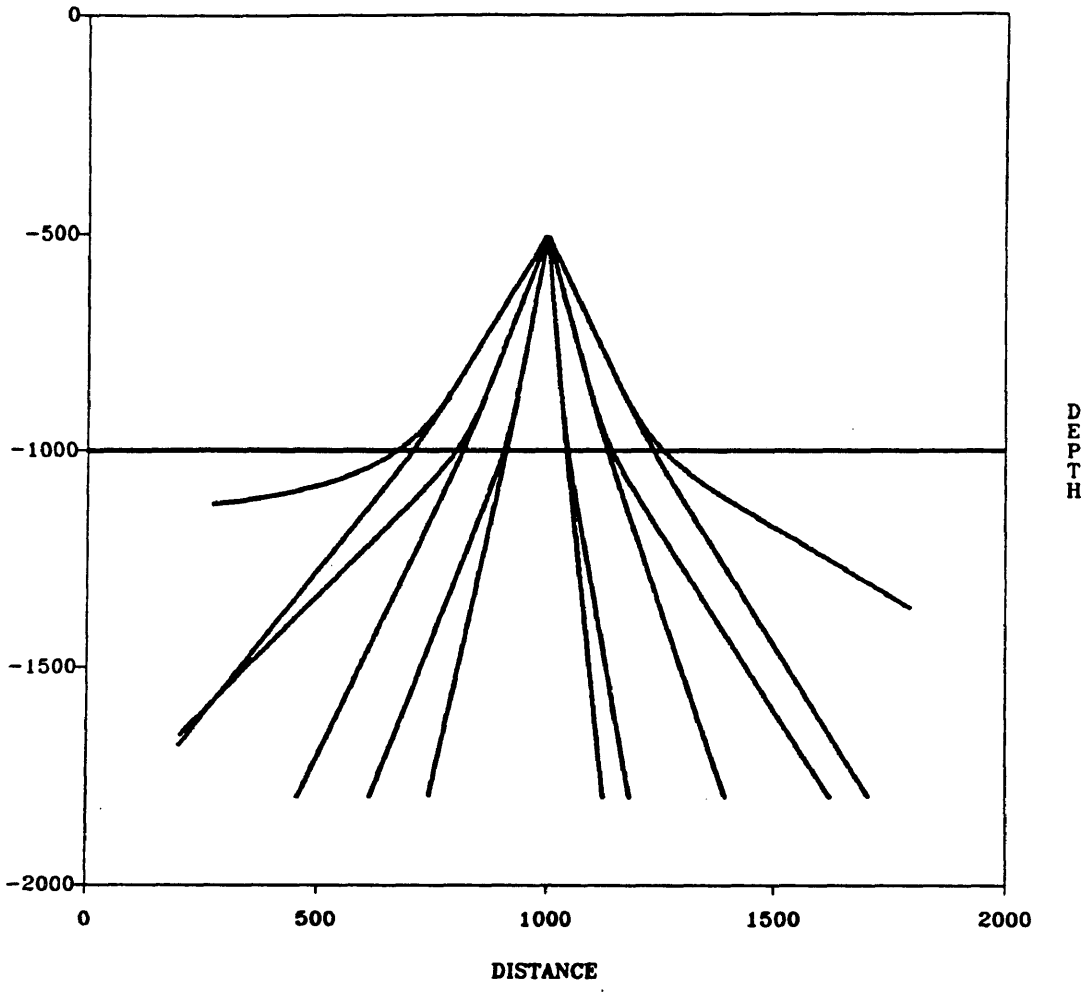

Figure 3.23: These rays are traced using smoothed version of Figure 3.22. Rays in this case penetrate smoothly to the underlain layer. The discrimination angle is decreased by smoothing. Note also the critically refracted ray at the extreme left of Figure 3.22 has been allowed here to penetrate.

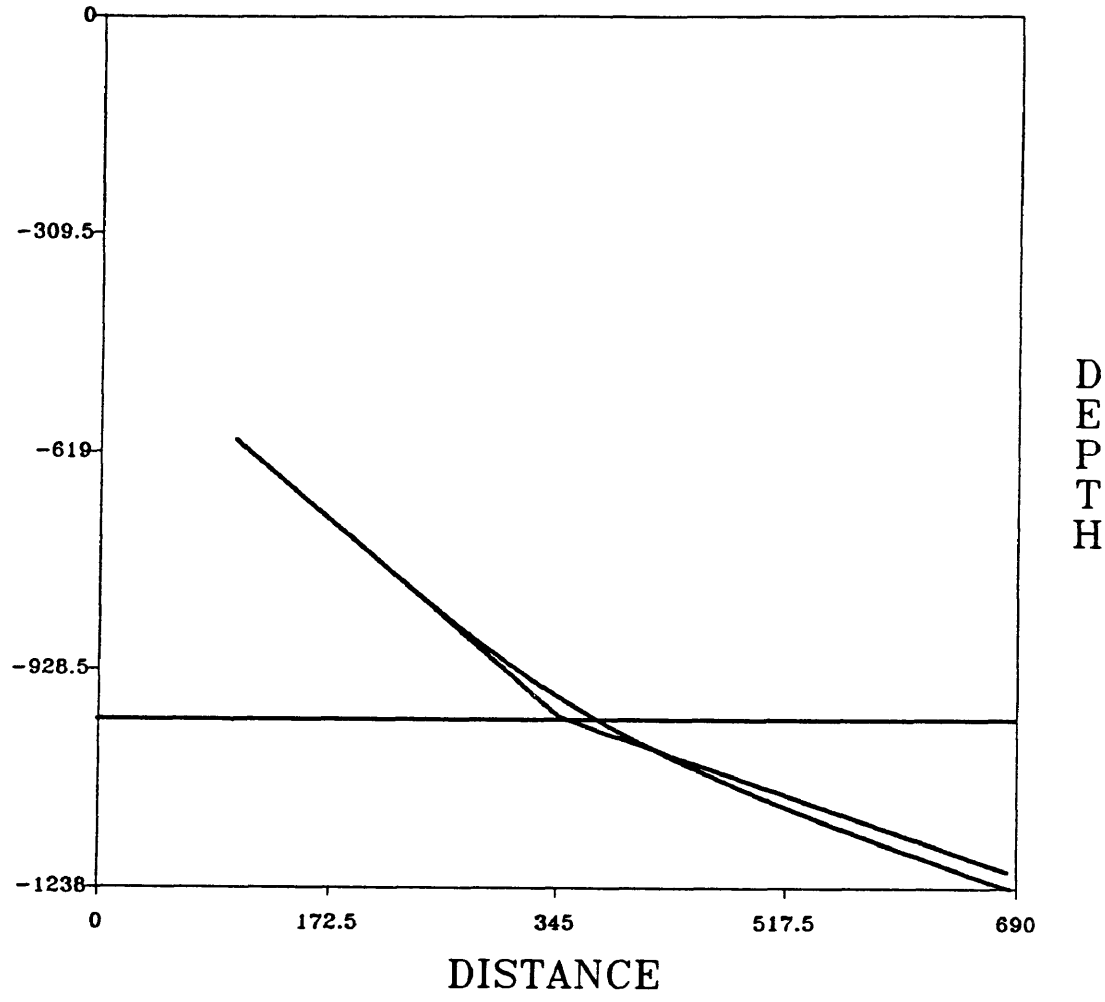

Figure 3.24: Plot of two rays having the same take-off angle. One ray in this case were traced using a discontinuous speed (unsmoothed ray) and the other using the smoothed version (smoothed ray). Smoothed ray tend to refract as it approachs the transition zone. At the interface, unsmoothed ray refracts with a bigger refraction angle allowing the smoothed ray to have more penetration towards vertical directions.

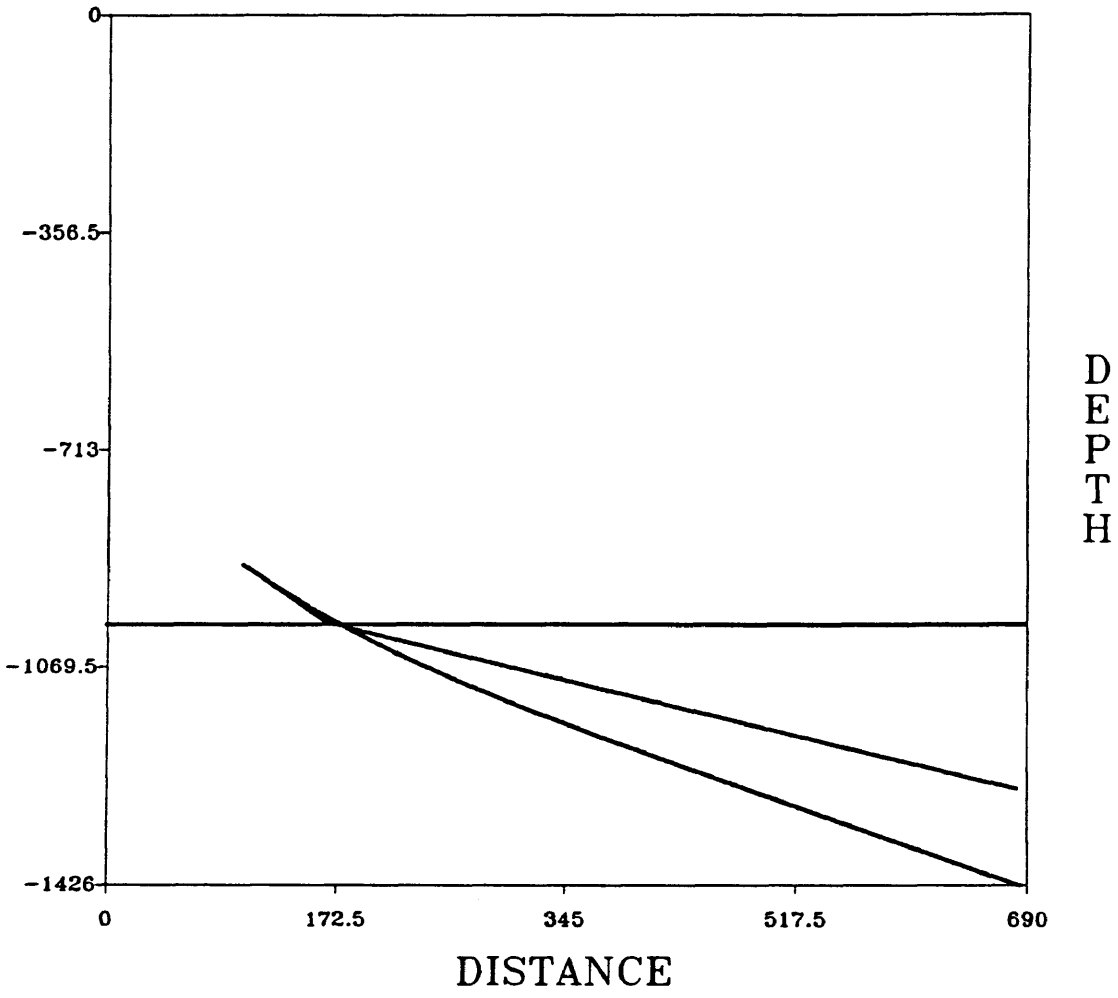

Figure 3.25: This plot illustrates that a more discrimination is attained when the source is close to the interface.

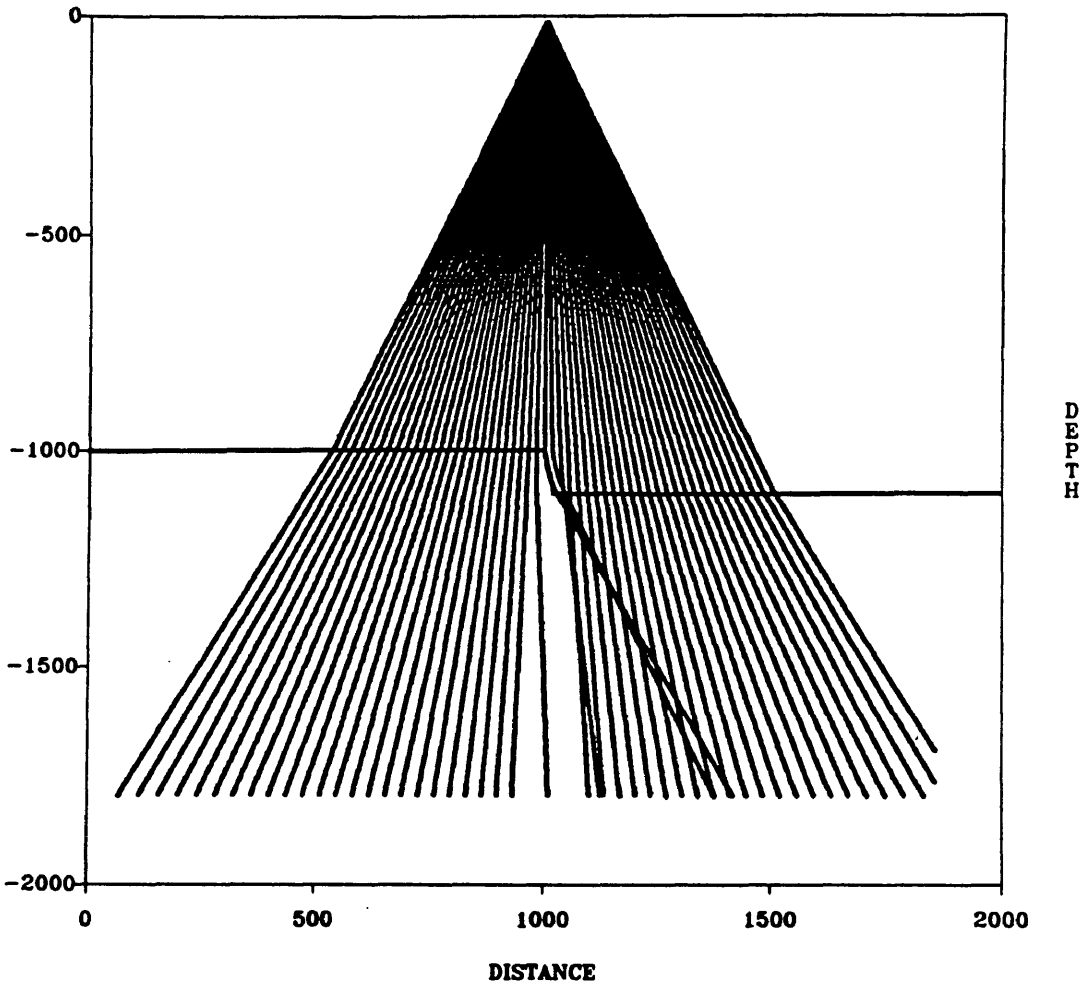

Figure 3.26: A plot of rays through a faulted horizontal reflector. Source is located at depth 10 ft and offset 1000 ft. Angles range from -25.0 degrees to 25.0 degrees using one degree increment. Central rays across the fault are significantly deflected because of a large sudden change in incident angle with respect to change in take-off angle.

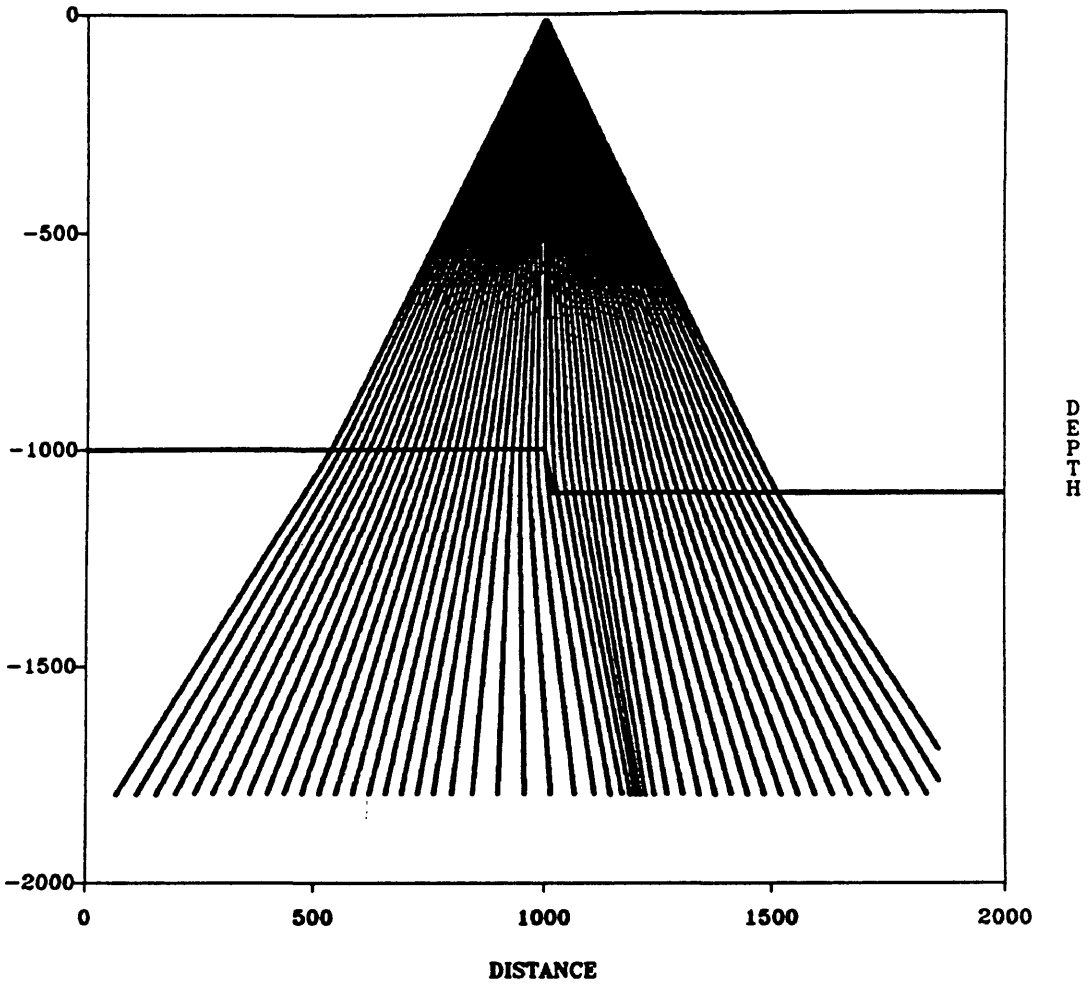

Figure 3.27: Smoothing in this case was effective in filling the ray gap of Figure 3.26. Also a more uniform ray distribution is attained than in the last figure.

fill the ray gap and provide a more uniform ray distribution.

It should be noted here that the details along interfaces are not the direct cause of the sensitivity problem. Rather, they are the indirect cause. It is the type of grid geometry adopted for sampling the speed function which is directly connected to the sensitivity. In the case of rectangular grid geometry, the shape of the sampled speed depends greatly on the fine details along interfaces. I believe that this tied connection for this particular grid geometry is primarily responsible for the rays sensitivity problem in this case.

This can be seen more clearly if we recall that the speed cells along the interfaces can take different shapes depending on which of the four cell corners has advanced into the underlain layer. Figures 3.28 and 3.29 show two different shapes that a cell can take along interfaces. Now, which corner will advance into the underlying layer depends highly on the specific details that exist along the interfaces. Regular geometry is characterized by fixed distances between grid points in any speed cell in the model, while neither gaps nor overlaps are allowed between the cells. When a regular grid geometry (rectangular grid) is considered, I believe that the frequency at which these cells flip from one shape to another along interfaces, as the take-off angle changes, is the direct cause of rays sensitivity in our particular tracing problem.

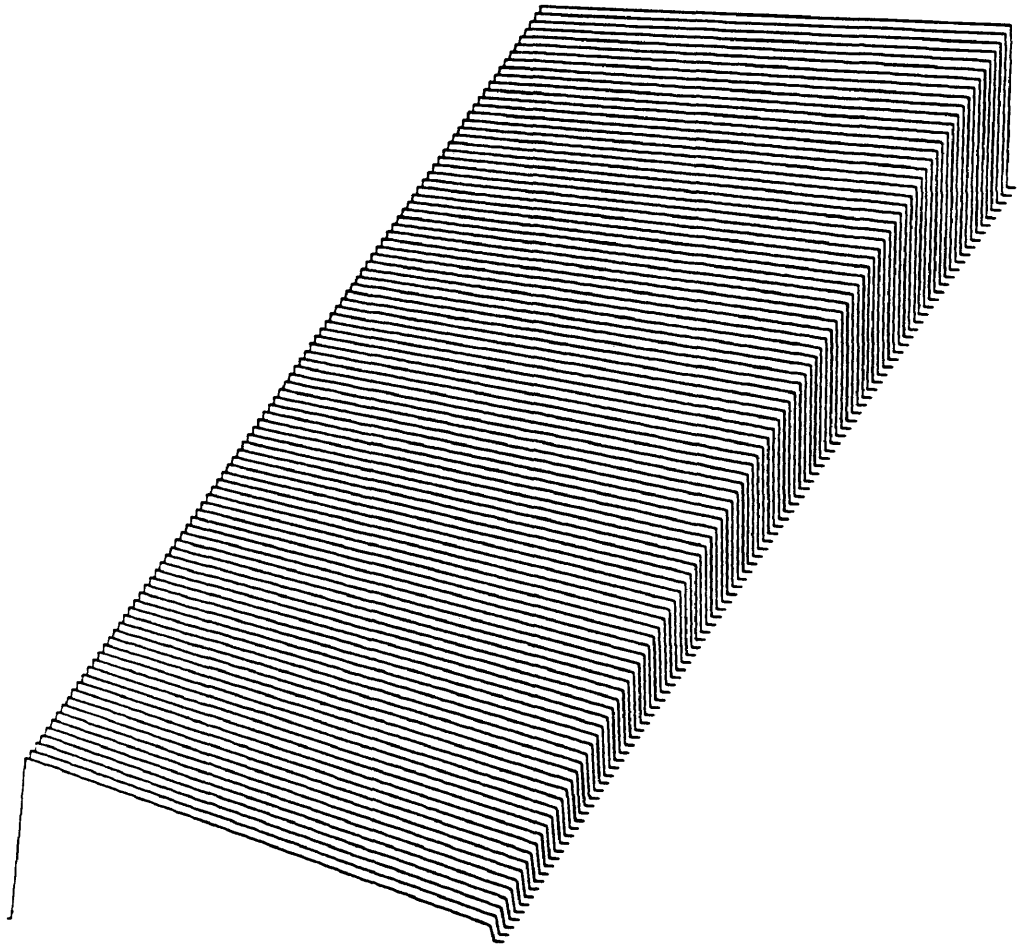

Figure 3.28: Plot of one type of speed cells. In this type only one corner has advanced into the underlain layer.

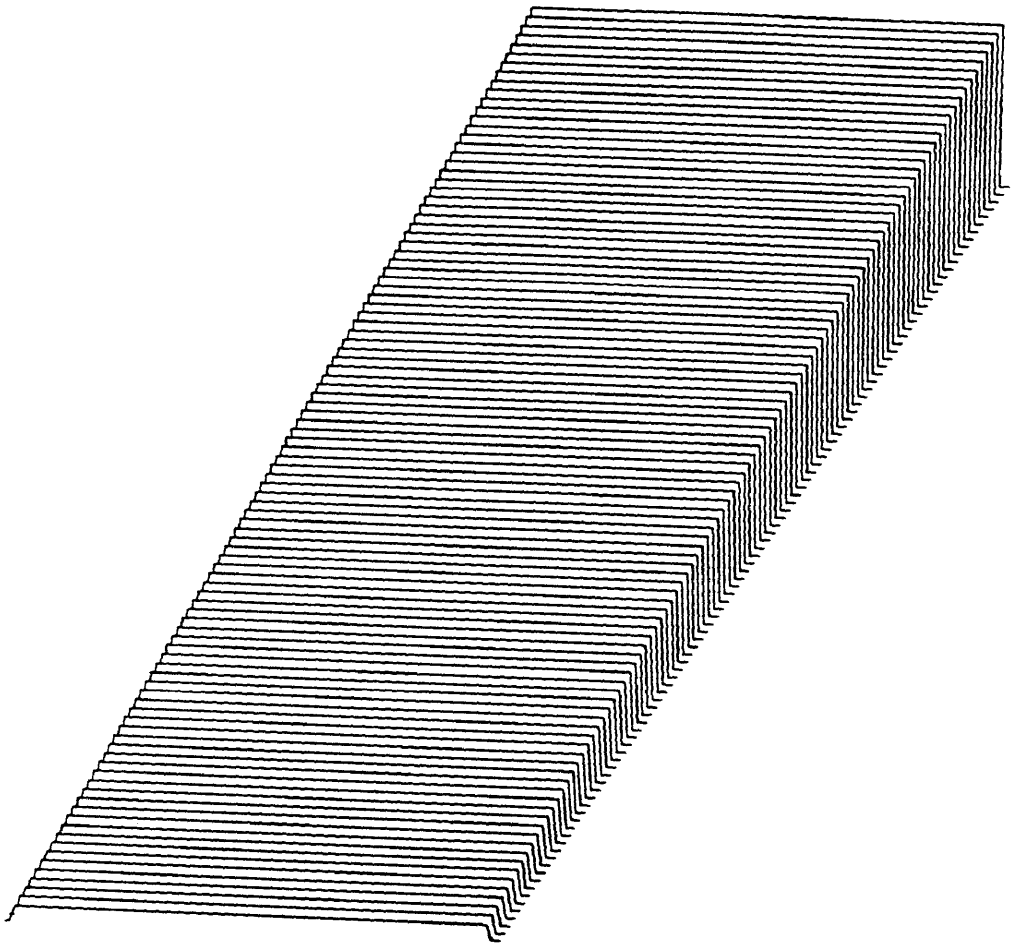

Figure 3.29: Plot of another type of speed cells. Two corners have advanced into the underlain layer.

3.5 Ray Amplitude Examples.

Equation (2.1.7) can be used to compute the amplitude in two ways. Either by interpolating the amplitude at certain locations between rays in the model or by evaluating the amplitude at a specific depth along the rays themselves. For this discussion, I have chosen to evaluate the amplitude along the rays to avoid the additional errors that arise from interpolation. In addition, it is straightforward to evaluate the amplitude along the rays in the code. So, at a certain depth, I evaluate all parameters and variables that appear in equation (2.1.7) for a range of take-off angles and then plot the amplitude versus offset along the profile.

The model on Figure 3.9 was chosen to illustrate the amplitude behavior after the refraction across the interface. Figure 3.30 shows a plot of the amplitude versus offset at depth 1100 ft with two sharp peaks at about 900 ft and 1100 ft offsets. The plot here is symmetric since the model of Figure 3.9 is symmetric and the source is located above the middle of the semi-circle. These two peaks indicate the presence of two caustics associated with the refraction of the rays by the semi-circle along the interface as indicated in Figure 3.31. Figure 3.32 shows the same amplitude after applying Gaussian smoothing with filter of radius 270.0 ft. The smoothing in this case removed the two peaks and allowed the amplitude to be a smooth function of offset. If we are interested in the semi-circle then it is difficult to say something about its presence from

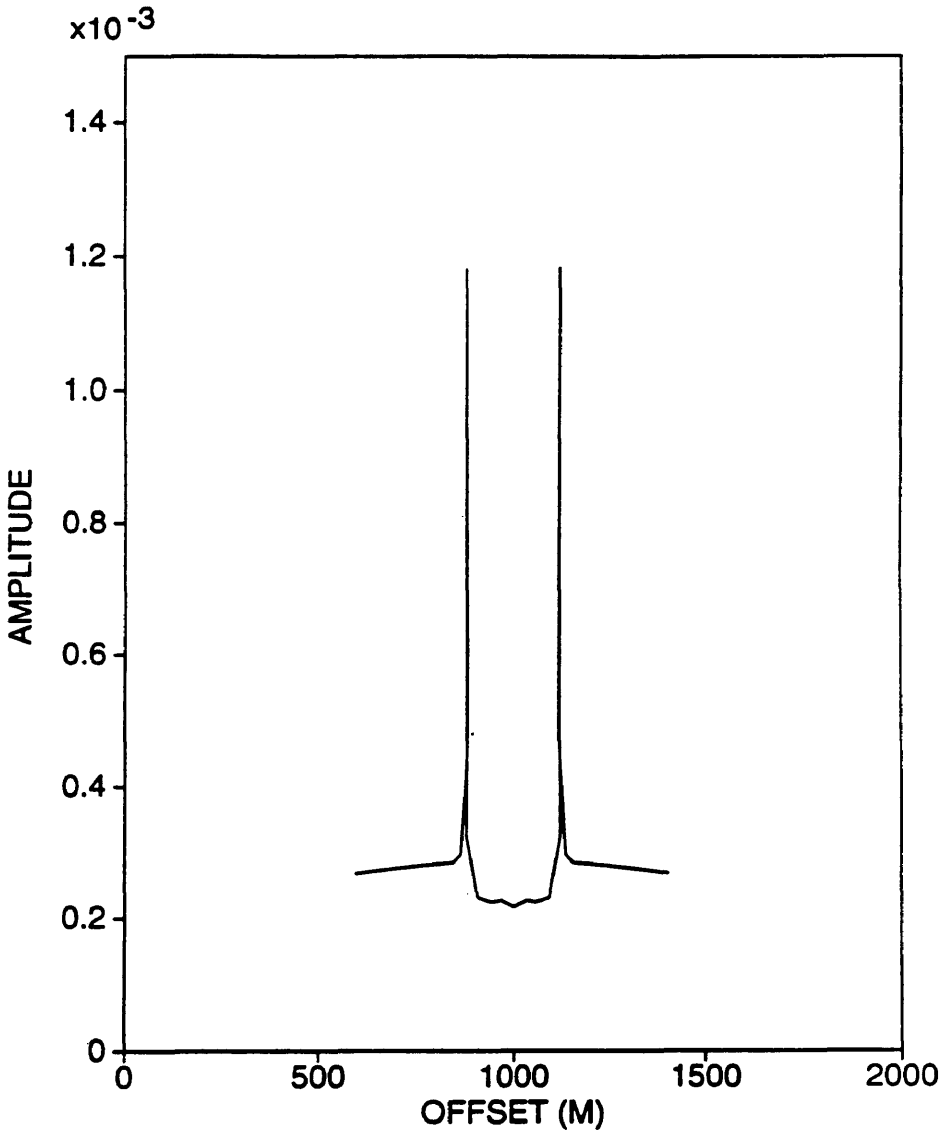

Figure 3.30: Plot of the amplitude fo Figure 3.9. The amplitude in this case was plotted at depth 1100 ft with angles range from -20.0 to 20.0 degrees at an increment of 1.0 degree. The plot shows two sharp peaks at about 900 ft and 1100 ft offsets. These two peaks reflect the presence of two caustics associated with the refraction by the semi-circle.

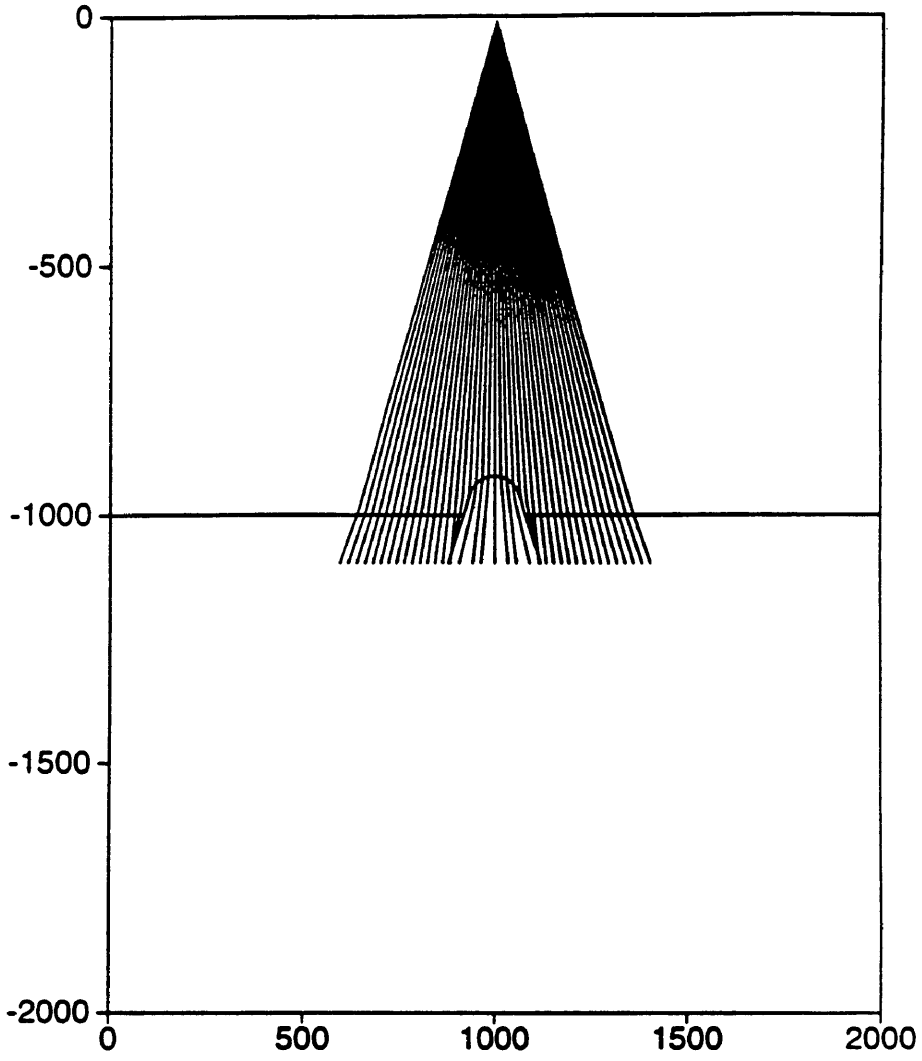

Figure 3.31: Rays associated with the amplitude in Figure 3.30. The two places were the rays cross each other are the locations were the cusps of the two caustics are positioned. In addition, they are the same locations were the two peaks occur in Figure 3.30.

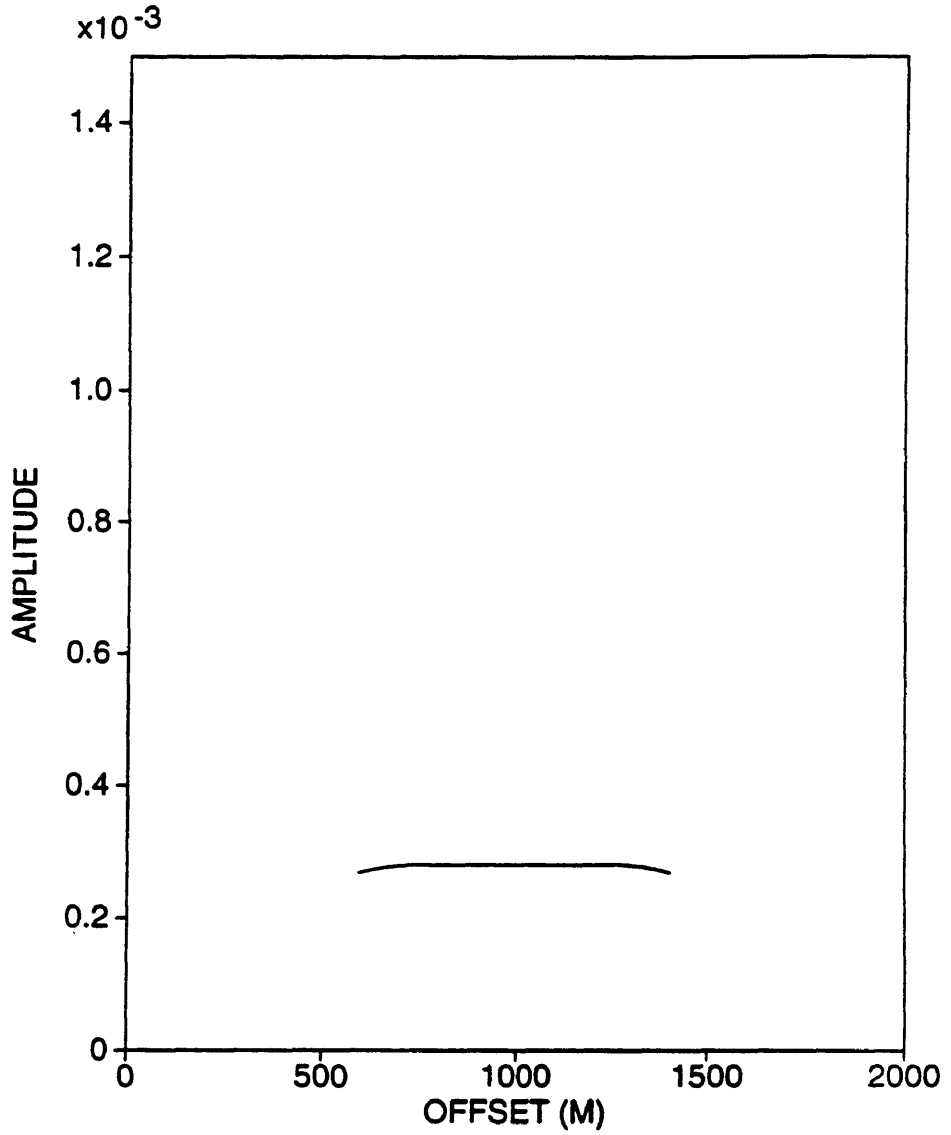

Figure 3.32: The smoothed amplitude version of Figure 3.30. The radius of the Gaussian filter was 270.0 ft.

just looking at the smoothed version of the amplitude plot. This is again one of the side effects of smoothing on amplitude calculations that will be encountered in modeling problems when the ray method is implemented after smoothing is applied to the speed distribution. These effects should be expected because the discontinuities (ramps in our case) of speed propagation have been replaced by smooth transitions after smoothing was applied.

The amplitude sensitivity appears to be more pronounced than the sensitivity of ray trajectories, as mentioned earlier by Cerveny (1985), and as suggested by the code outputs that will be soon below. To illustrate this, I use a simple model similar to that of Figure 3.8 but with different number of control points. The interface in this case was constructed using 73 control points (217 points were used in Figure 3.8). These points are 50.0 ft apart along the horizontal flanks except the points between offsets 900.0 ft and 920.0 ft and offsets 1080.0 ft and 1100.0 ft which are 10.0 ft apart. The points within the semi-circular feature were chosen to be 5.0 ft apart. The plot of the model in this example is shown in Figure 3.33 which clearly shows the artifacts along the semi-circle as well as along the horizontal flanks near the semi-circle edges.

The amplitude before the refraction takes place should be a concave down curve since more distance is travelled towards the sides of the model and since a uniform speed is encountered in the upper layer. This is because of more decay of amplitude due to geometrical spreading. This decay is illustrated by Figure 3.34 where the amplitude was computed along the rays at a depth of 850.0 ft using speeds of 10000.0

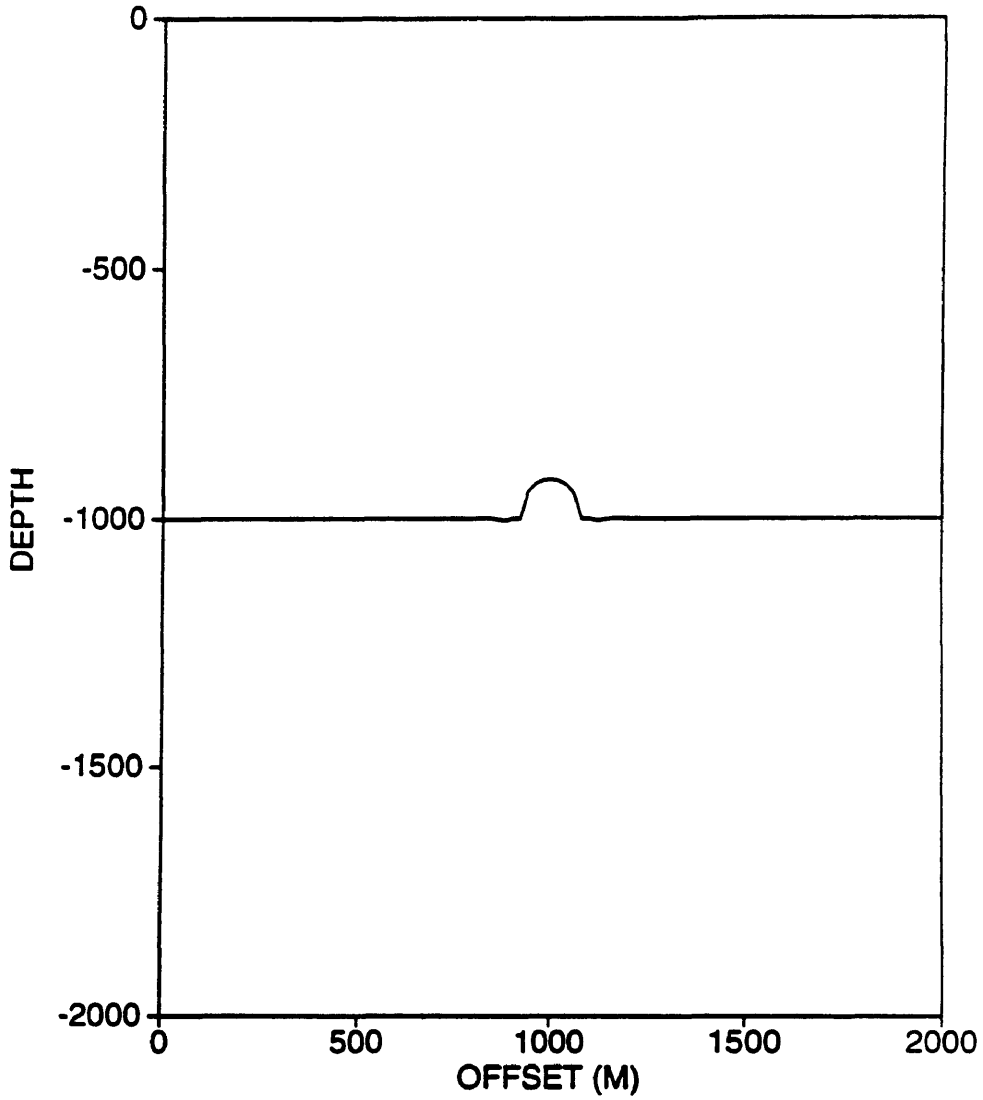

Figure 3.33: A simple model that consists of a single interface constructed using 73 control points. These points were 50.0 ft apart along the horizontal flanks except the closest four points to the semi-circle edges were they are 10.0 ft apart. Within the semi-circle, the points are 5.0 ft apart.

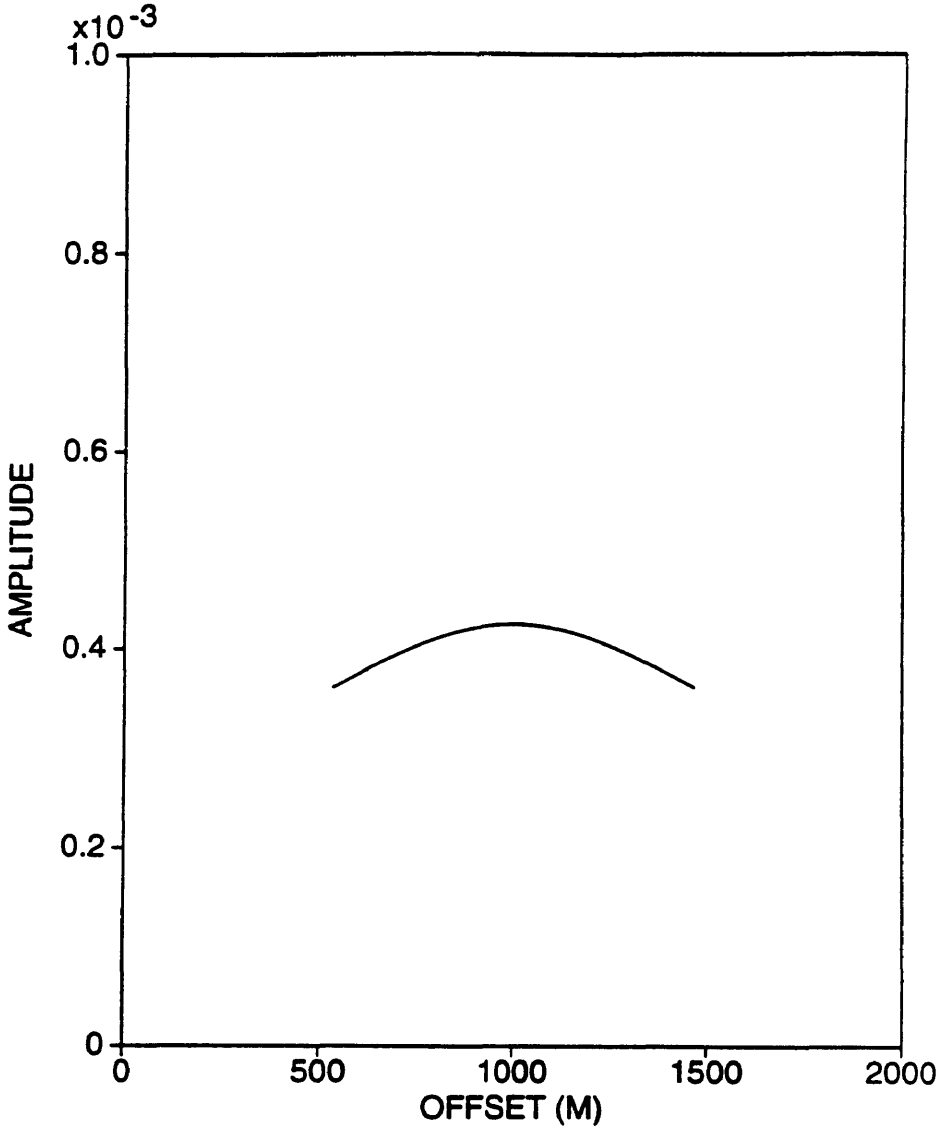

Figure 3.34: The amplitude distribution along the rays using the model in Figure 3.33 at depth 850.0 ft. The source is located at 1000.0 ft offset and 100.0 ft depth.

ft/sec and 12000.0 ft/sec for the upper and lower layers respectively. The amplitude is then computed at depth 1150.0 ft, after the refraction of rays took place, and plotted again as shown in Figure 3.35. This plot illustrates a great sensitivity in amplitude behavior due to the presence of small perturbations along the horizontal flanks of the interface as well as the semi-circle. These perturbations are difficult to see along the interface but their presence is clearly indicated by the amplitude behavior as well as the crossings of the associated ray trajectories as shown in Figure 3.36. Note that each peak in Figure 3.35 corresponds to some rays crossing (caustic) in Figure 3.36. The presence of these caustics was confirmed by connecting the amplitude points as a function of increasing take-off angle and it was found that the amplitude curve indeed intersects with itself indicating the presence of caustics.

The amplitude plot in Figure 3.35 should not show such a sensitive behavior if the model was a perfect semi-circle with perfectly horizontal flanks. The rapid fluctuation of amplitude behavior reflects the great impact of these small perturbations on the refracted field below the interface. These perturbations are easily recognized in the above simple model by the rays behavior. However, these perturbations are hard to distinguish when we deal with real models were rays behavior become a vague indicator of the perturbations presence because reflectors in real models are not necessary flat and so the curvatures of the interfaces enter the computation process. To appreciate the problem of amplitude sensitivity more, I refined the angle increment from 1.0 degree to 0.5 degree and plotted the amplitude again as shown in Figure 3.37

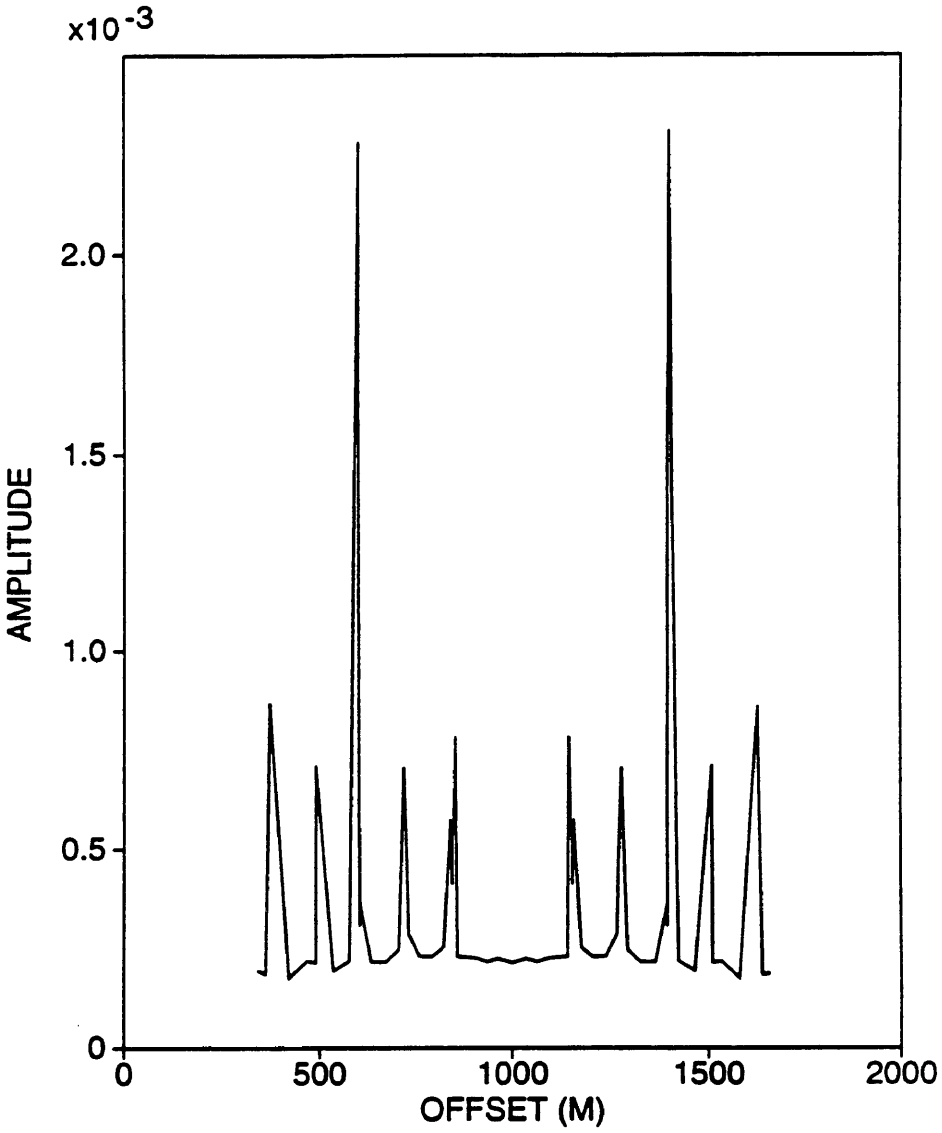

Figure 3.35: Amplitude distribution at depth 1100.0 ft using the model in Figure 3.33. Great sensitivity associated with the interface artifact perturbations.

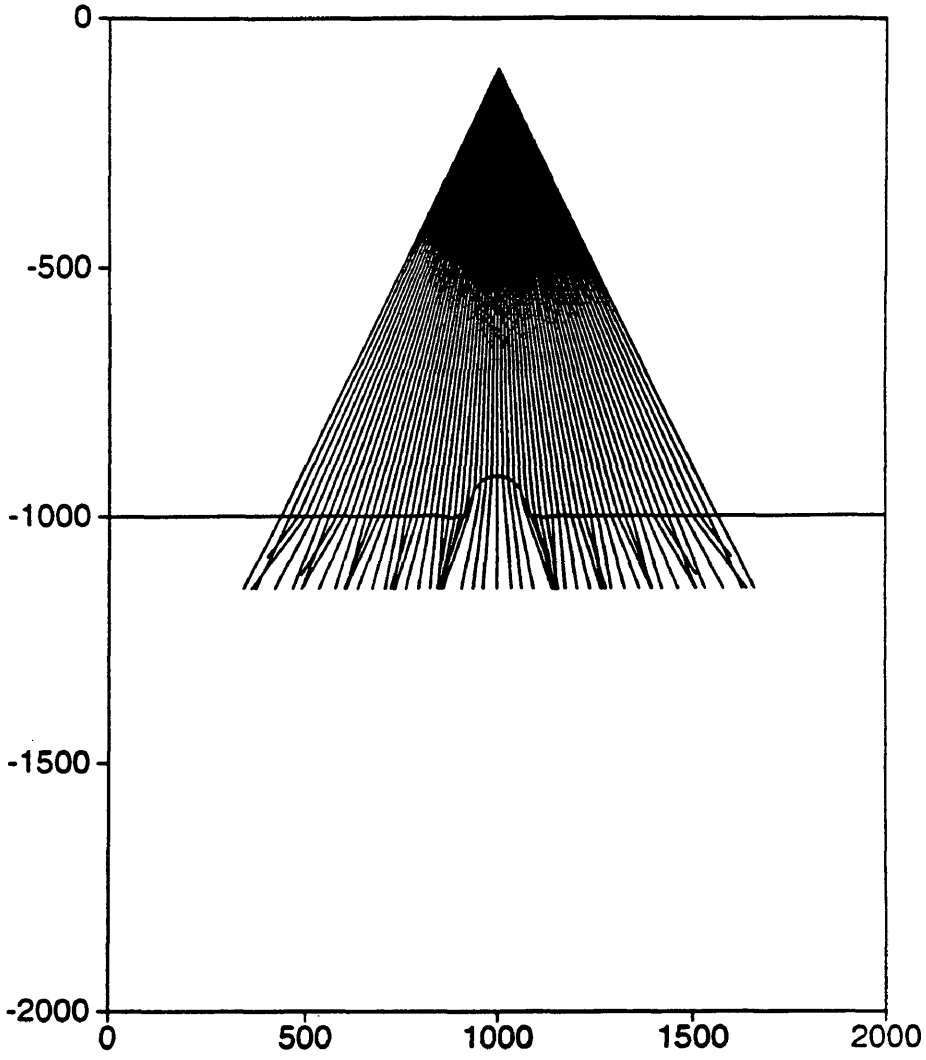

Figure 3.36: Ray trajectories associated with Figure 3.35. The source is located at offset 1000.0 ft and depth 100.0 ft. Each sharp peak in Figure 3.35 is caused by some ray crossing above caused by the artifacts along the interface.

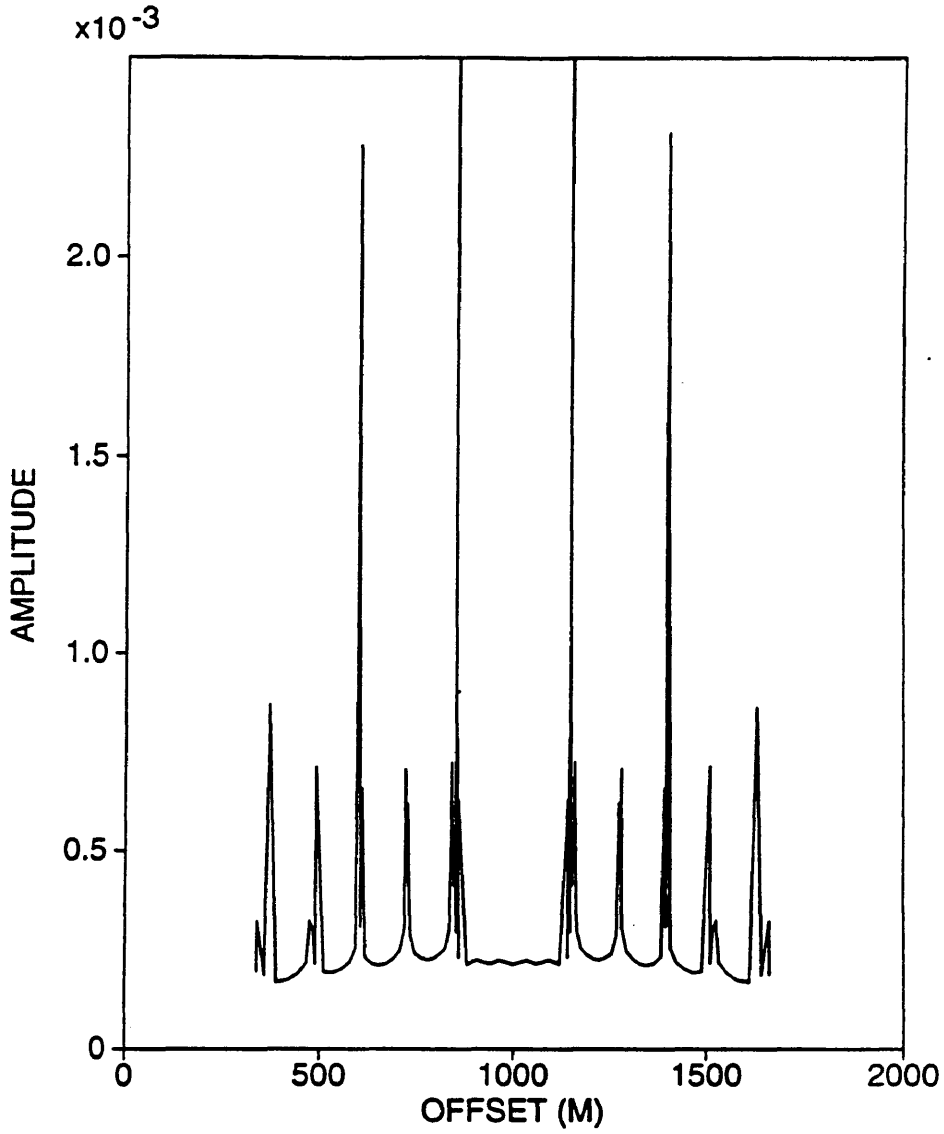

Figure 3.37: Amplitude plot of Figure 3.35 after refining the take-off angle from 1.0 degree to 0.5 degree. This illustrates the problem of sensitivity encountered in ray tracing modeling.

with the corresponding ray trajectories in Figure 3.38.

Figure 3.37 illustrates clearly the sensitivity problem encountered when using the ray method to estimate the amplitude. Figure 3.39 shows a plot of the amplitude after smoothing was applied with a filter of radius 270.0 ft. Smoothing here was effective in removing these rapid fluctuations but at the same time it removed almost all possible indications of the semi-circular presence along the interface. However, we are not interested at the moment in the semi-circle because the main objective here is to remove these fluctuations.

Last examples for amplitude calculations will be considered now using the Marathon Oil Company model shown in Figure 3.20. These examples are computed at four different depths in the model from the same source location and for the same take-off angle range. These depths are at 5000.0, 7000.0, 9000.0, and 11000.0 ft. The source for these plots is located at depth 10 ft and offset 16000 ft. The take-off angles in this case range between -50 to 50 degrees with increment of 1 degree. The speeds chosen for the Marathon model were 11750.0, 15750.0, 22410.0, 15750.0, 19900.0, and 20000.0 ft/sec from top to bottom.

Figure 3.40 shows the amplitude at depth 5000.0 ft were two sharp peaks, at about offsets of 12300.0 ft and 15500.0 ft, indicate two caustics due to the combined geometry of the first and second interfaces in the model. The rest of the fluctuations, I believe, are due to the artifact perturbations along the first and second interfaces. These

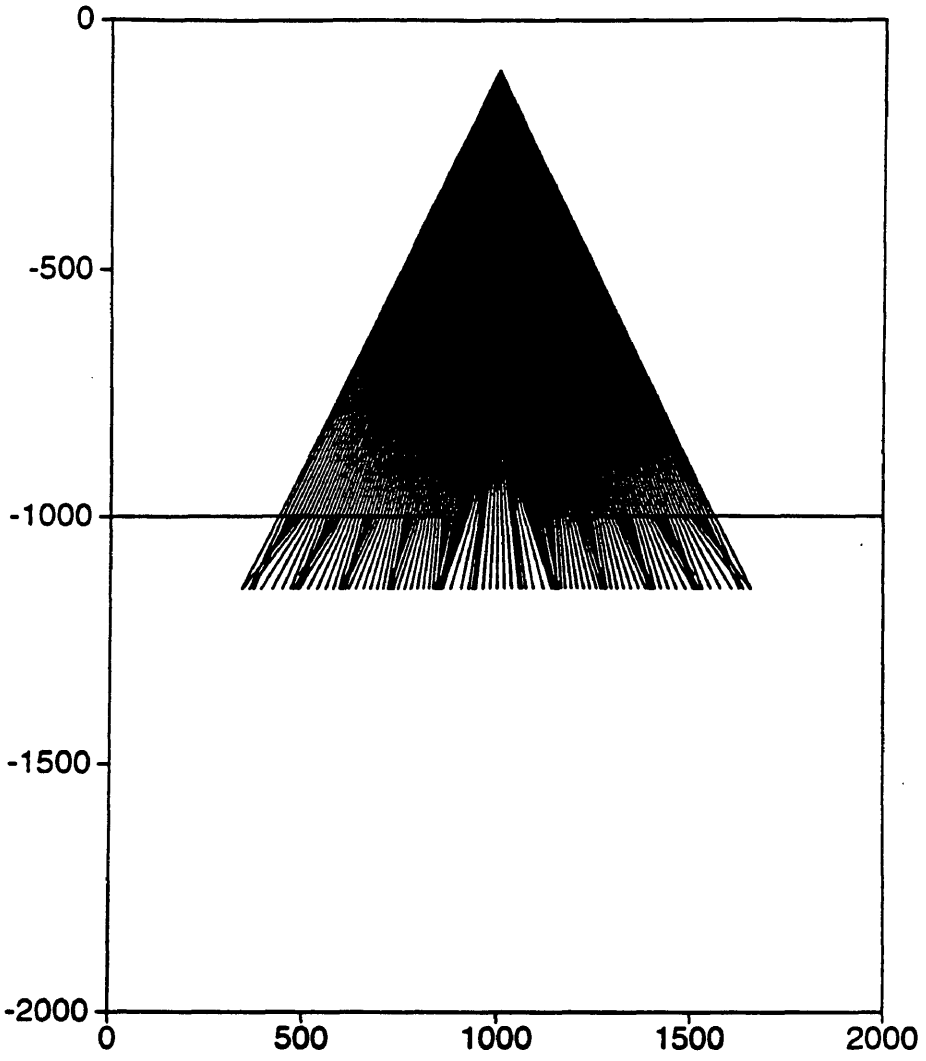

Figure 3.38: Ray trajectories associated with Figure 3.37.

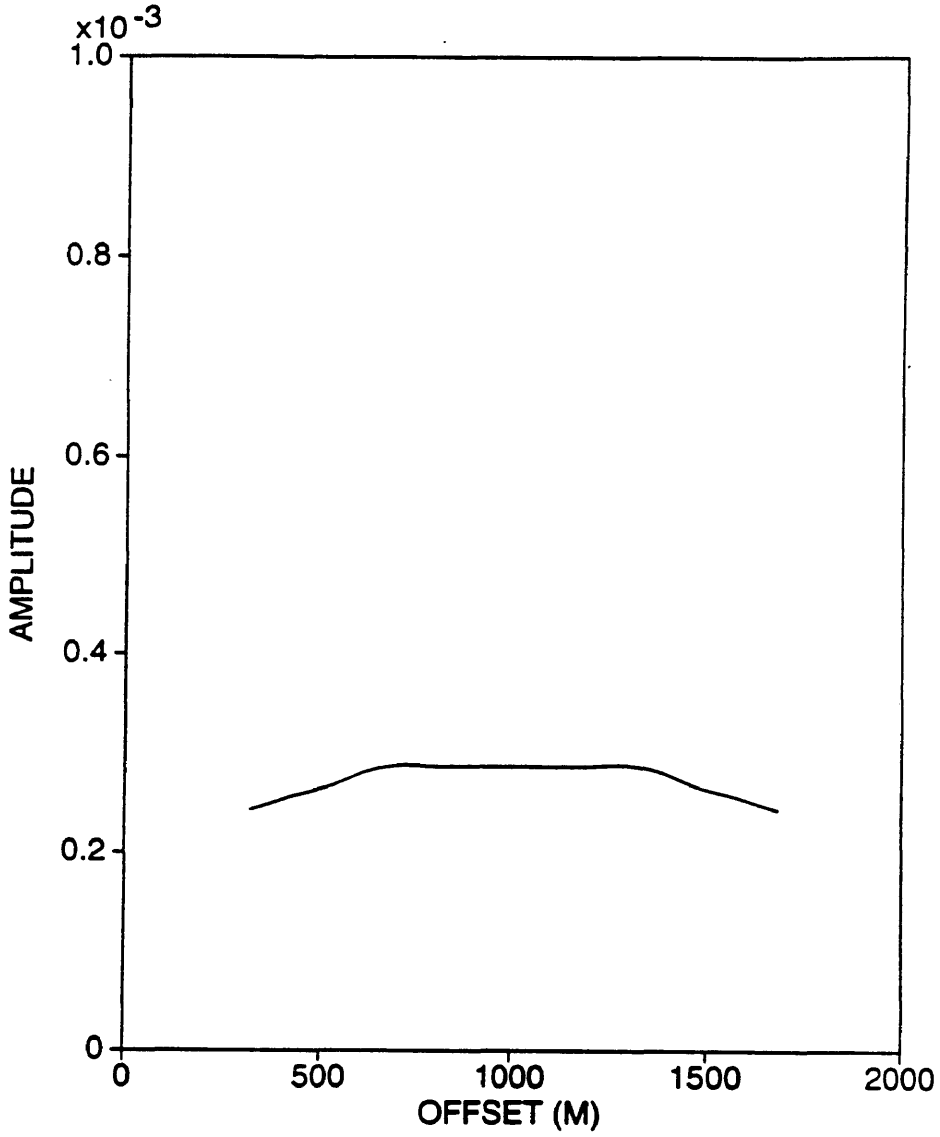

Figure 3.39: Smoothed version of the amplitude in Figure 3.37. The radius of the filter is 270.0 ft while the plotting scale is .001 instead of .005 as in Figures 3.35 and 3.37 to show the general trend of the amplitude. The fluctuations were effectively removed in this case while the general trend of the amplitude is still preserved.

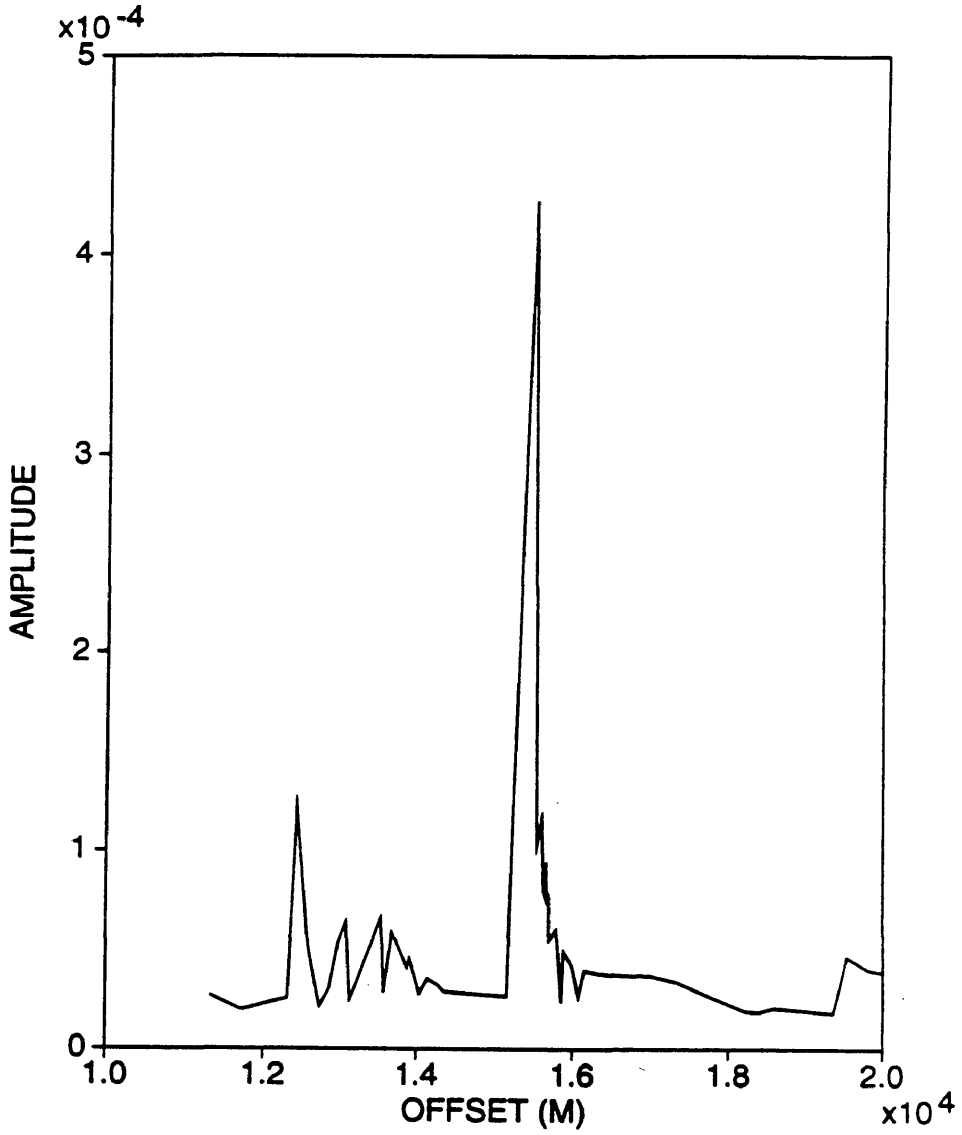

Figure 3.40: Ray-theoretic amplitude at depth 5000.0 ft using the Marathon Oil Company model with speeds 11750.0, 15750.0, 22410.0 ft, 15750.0, 19900.0, and 20000.0 ft/sec. Many fluctuations due to artifacts along the first and second interface.

two peaks tend to weaken as we go to greater depths as a result of geometrical spreading and as suggested by the Figures 3.41, 3.42, and 3.43 were the amplitudes are computed at depths 7000.0, 9000.0, and 11000.0 ft respectively. These plots confirms the sensitivity problem encountered in practice associated with the ray-theoretic amplitudes. Applying smoothing with filter radius of 1200.0 ft, to Figure 3.41, 3.42, and 3.42 the results are shown in Figures 3.44, 3.45, 3.46 respectively. The amplitude fluctuations were effectively removed while the average value of the amplitude is still preserved in the smoothed version.

Interface artifacts seems to have greater influence on the ray-theoretic amplitudes more than they do on the ray trajectories even though they have strong influence on both. It is important to minimize this influence if meaningful information is to be extracted from the amplitude of the wave field as well as the phase function (traveltime). 2-D Gaussian type filters seems to be effective in reducing this influence in most of the cases that we considered. It is still not clear whether the smoothed amplitude will provide a more correct estimation to the true amplitude of the wave field or not. However, since our purpose is to have more suitable amplitudes for imaging purposes, the aim here is to suppress the effect of these artifacts to provide better imaging. Smoothing the speed of the model using 2-D Gaussian filters was found to be effective in minimizing this problem.

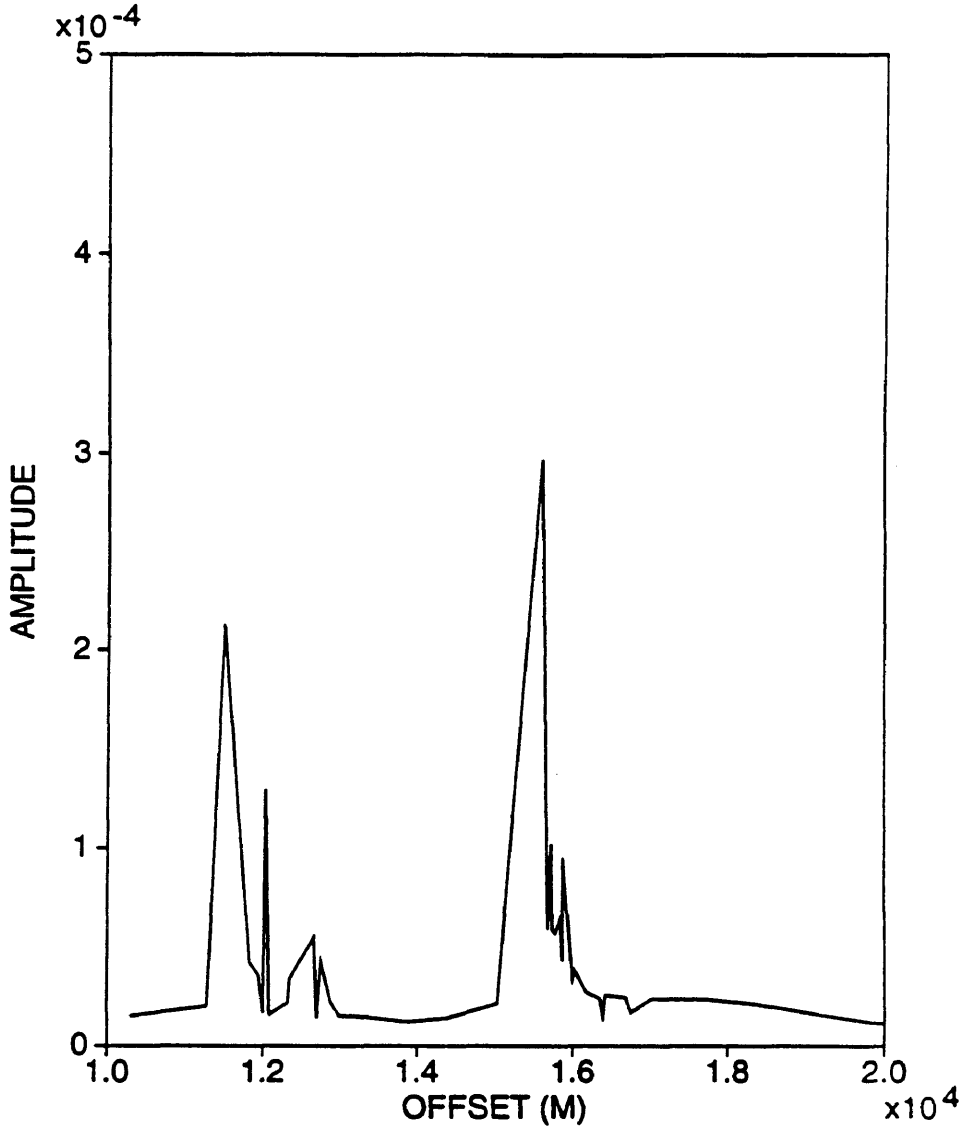

Figure 3.41: Ray-theoretic amplitude at depth 7000.0 ft using the Marathon Oil Company model with speeds 11750.0, 15750.0, 22410.0 ft, 15750.0, 19900.0, and 20000.0 ft/sec. Many fluctuations due to artifacts along the first, second, and third interface.

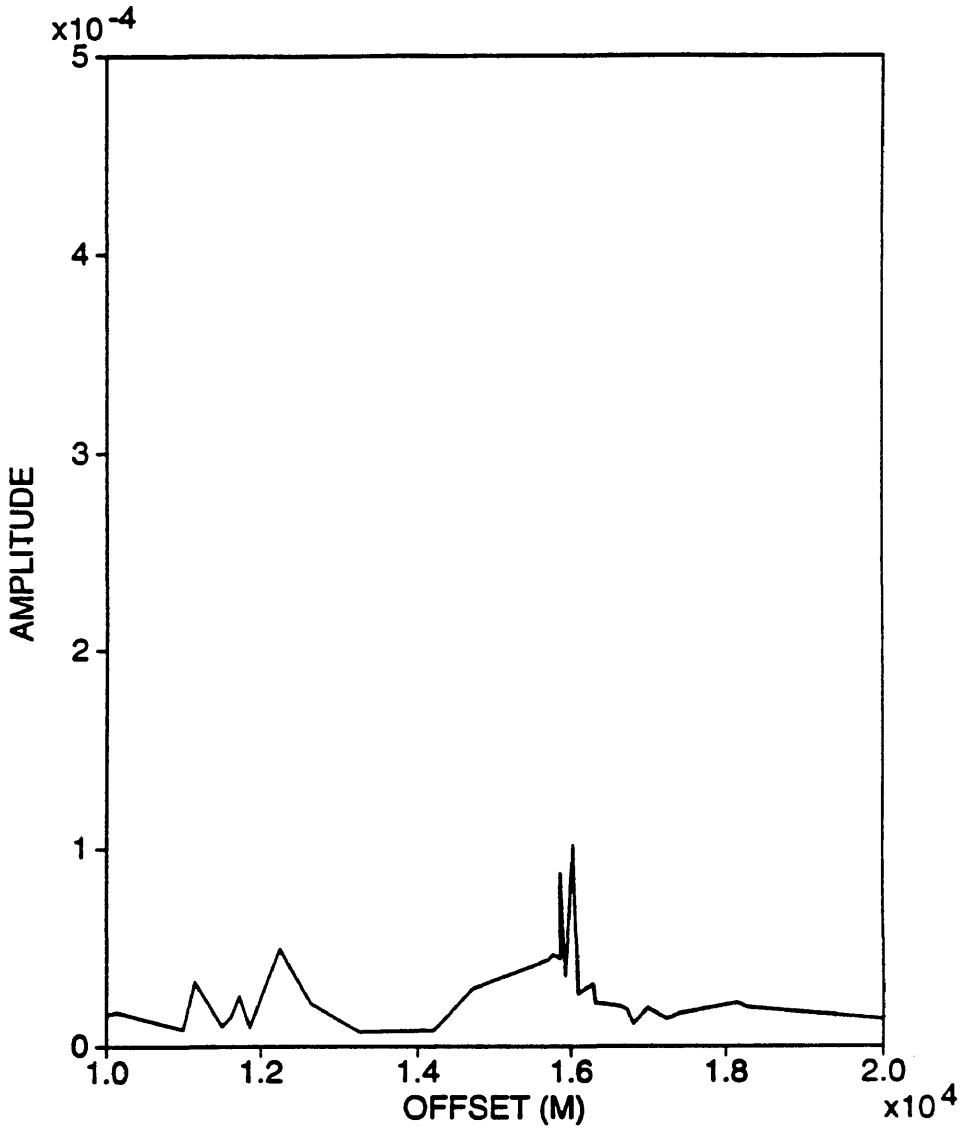

Figure 3.42: Ray-theoretic amplitude at depth 9000.0 ft using the Marathon Oil Company model with speeds 11750.0, 15750.0, 22410.0 ft, 15750.0, 19900.0, and 20000.0 ft/sec. Many fluctuations due to artifacts along the first, second, third, and part of the fourth interface.

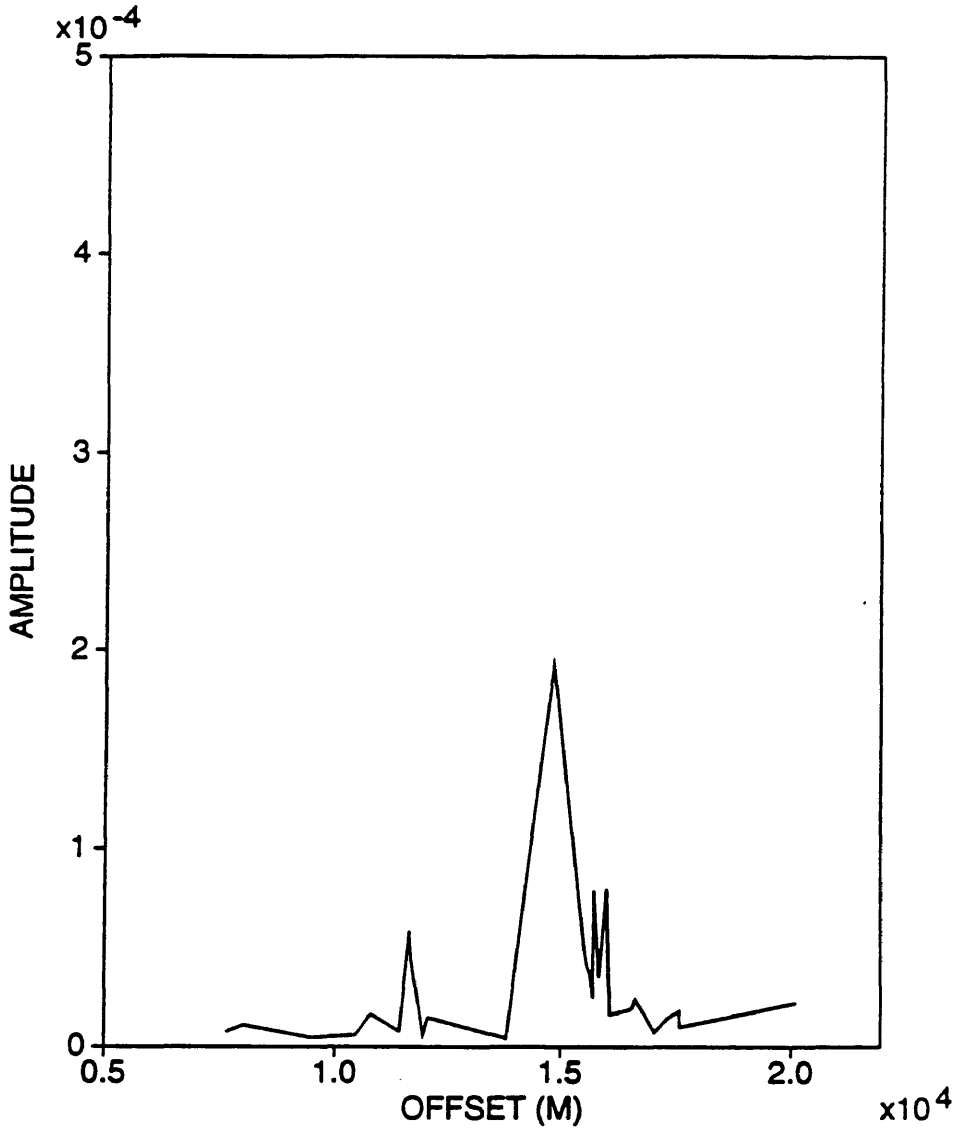

Figure 3.43: Ray-theoretic amplitude at depth 11000.0 ft using the Marathon Oil Company model with speeds 11750.0, 15750.0, 22410.0 ft, 15750.0, 19900.0, and 20000.0 ft/sec. Many fluctuations due to artifacts along the first, second, third, and fourth interface.

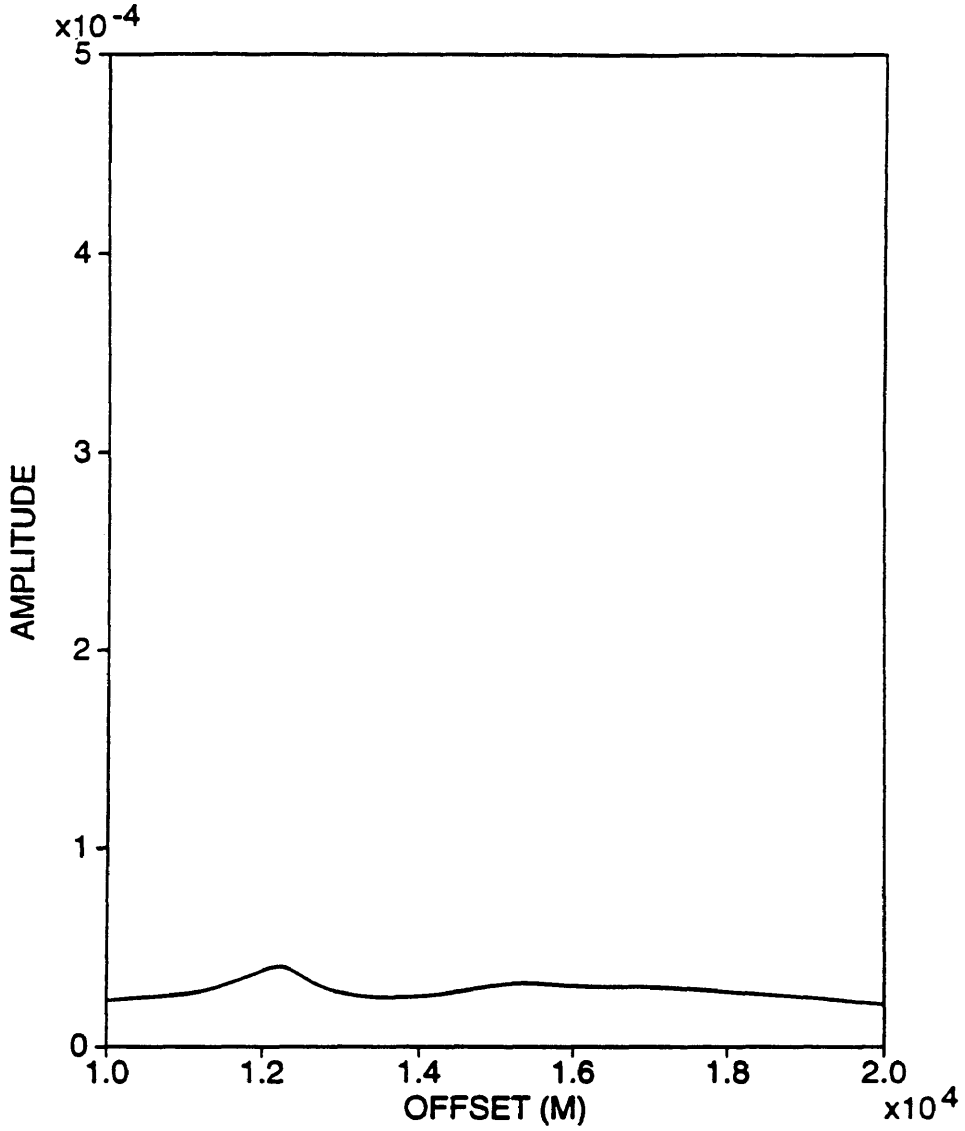

Figure 3.44: Smoothed version of Figure 3.41.

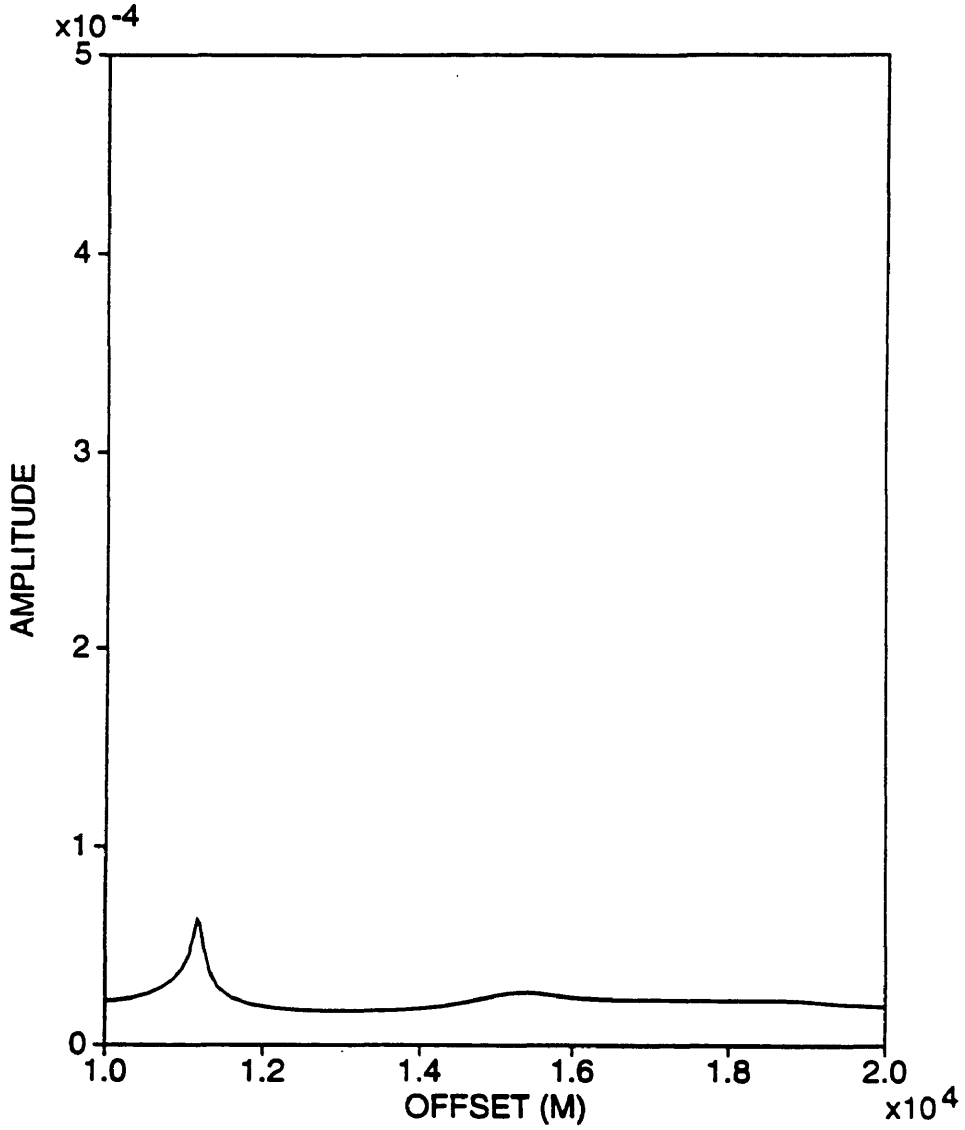

Figure 3.45: Smoothed version of Figure 3.42.

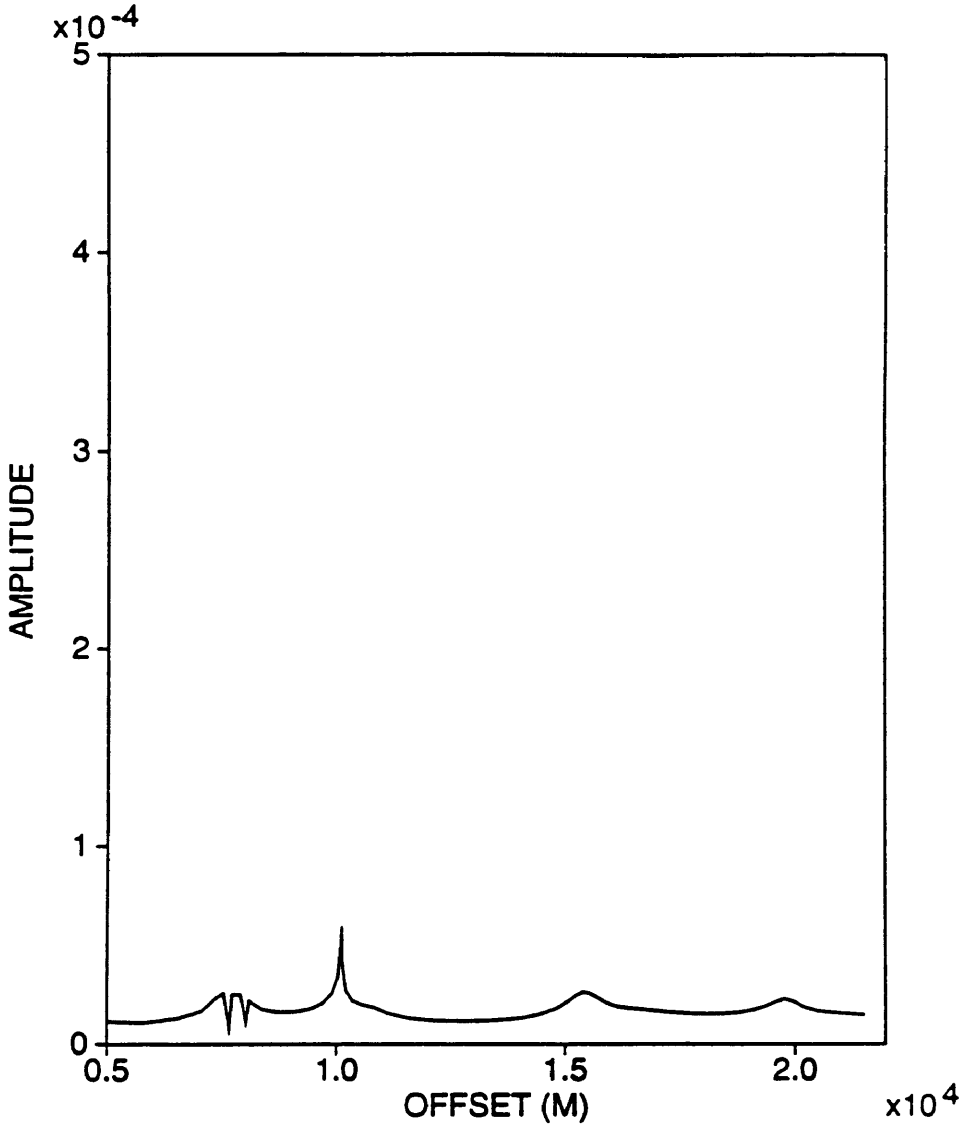

Figure 3.46: Smoothed version of Figure 3.43.

CONCLUSIONS

A ray tracing system in depth z is implemented using a speed distribution that depends only on x and z. The Jacobian of transformation in this case reduces from a 3×3 matrix to a 2×2 matrix for the 3-D case. The 2.5D assumption further reduces the in-plane Jacobian to an expression that depends only on two ray quantities for which one of these quantities is an auxiliary ray quantity. The expression also involves a dependence on the ray take-off angle that prevents tracing the horizontal rays because the amplitude is singular for these rays. Interface modeling was found to have unfavorable effects on ray tracing especially when a regular grid geometry is used to sample the speed distribution (i.e. rectangular grid). This was true even when spline interpolation is used to construct the interfaces. The reason is that speed cells (for regular grid) tend to fluctuate from one type to another as a result of artifact perturbations introduced by spline interpolation along interfaces. I believe that this fluctuation is responsible for the ray tracing sensitivity in our case. The sensitivity causes rays to deviate considerably and generates artificial gaps which could prevent modeling of important geological structures. Smoothing the speed using a 2-D Gaussian type filter was found to reduce the size of these gaps and to provide a more uniform ray distribution in the model. As an extension to this work, we recommend an implementation of this type of modeling to an inversion algorithm to test the output against the inversion operator performance.

REFERENCES

Bleistein, N., 1984, Mathematical Methods for Wave Phenomena: Academic Press Inc.

Bleistein, N., 1986, Two-and-one-half dimensional in-plane wave propagation: Geophysical prospecting, 34, 686-703.

Bleistein, N., Cohen, J. K., and Hagin, F. G., 1987, Two and one-half dimensional Born inversion with an arbitrary reference: Geophysics, 52, 26-36.

Byerly, P. E., 1965, Convolution filtering of gravity and magnetic maps: Geophysics, v. 30, 281-284.

Docherty, P. C., 1987, Ray theoretical modeling, migration and inversion in two and-one-half-dimensional layered acoustic media: Ph.D. thesis, Colorado School of Mines, Golden, Colo. also Center for Wave Phenomena Research Report number CWP-051.

Docherty, P. C., 1987, Two-and-One-Half-Dimensional Poststack Inversion, Expanded Abstracts of 57th Annual International SEG Meeting, Society of Exploration Geophysicists, New Orleans, pp. 564-565.

Dong, W., 1989, Finite difference ray tracing and common shot inversion: Master's thesis, Colorado School of Mines, Golden, Colo. Center for Wave Phenomena

Research Report number CWP-084.

Dong, W., 1989, Common-Shot Inversion in $c(x,z)$ Media, in Expanded Abstracts of 59th Annual International SEG Meeting, Society of Exploration Geophysicists, Dallas, pp. 928-931.

Gray, S. H., 1986, Efficient traveltime calculations for Kirchhoff migration: Geophysics, 51, 1685-1688.

Huang, T. S., 1972, Two-dimensional windows, IEEE Trans. Audio Electroacoustic (Correspondence), vol. AU-20, pp. 88-89, Mar. 1972.

LaFara, R. L., 1973, Computer methods fo Science and Engineering: Hayden Book Company, INC.

Langan, R. T., Lerche, I., and Cutler, R. T., 1985, Tracing of rays through heterogeneous media: An accurate and efficient procedure: Geophysics, 50, 1456-1465.

Madrid, J. A., 1985, Travel times and ray paths for continuous media: Geofisica Internacional, 24-3, 439-458.

Press W. H., Flannery B. P., Teukolsky S. A., and Vetterling W. T., 1988, Numerical Recipes in FORTARN: Cambridge University Press.

Smith, A. M., Goldberg, M., and E.S.K. Liu, Numerical ray tracing in Media involving continuous and discrete refractive boundaries: Ultrasonic Imaging, 2, 291-301 (1980).

APPENDIX A

In this appendix I derive the traveltime ray system given by equation (2.1.5) using the method of characteristics. This derivation can be found also in Bleistein (1984). We start with the equation

$$F(x, z, \tau, p, q) = 0. \tag{A.1}$$

where $p = \tau_x$ and $q = \tau_z$. To ensure that this is a partial differential equation of τ, we require that

$$F_p{}^2 + F_q{}^2 = 0. \tag{A.2}$$

From equation (A.1), the total differentials with respect to x and z satisfy

$$\frac{dF}{dx} = 0, \qquad \frac{dF}{dz} = 0, \tag{A.3}$$

this implies

$$F_x + F_\tau p + F_p p_x + F_q q_x = 0, \text{ and } F_z + F_\tau q + F_p p_z + F_q q_z = 0. \tag{A.4}$$

Here we assume that the solution is sufficiently smooth to allow the interchange of orders of differentiation

$$p_z = q_x, \tag{A.5}$$

which gives

$$F_p p_x + F_q p_z = -F_x - pF_\tau, \quad F_p q_x + F_q q_z = -F_z - qF_\tau. \tag{A.6}$$

These are quasi-linear first order partial differential equations for p and q. Such equations can be solved by the method of characteristics. The projections of the characteristics in the x,y-plane are called base characteristics. These trajectories are controlled by the coefficients of p_x, q_z or q_x, q_z respectively. Since those coefficients are the same, the base characteristics for these two equations are the same, and we can view the equations as describing the simultaneous propagation of p and q along these base characteristics. Applying the linear theory (Bleistein 1984)

$$\frac{dx}{F_p} = \frac{dz}{F_q} = -\frac{dp}{F_x + pF_\tau} = -\frac{dq}{F_z + qF_\tau} \tag{A.7}$$

We calculate the change in τ along these base characteristics as

$$d\tau = p\,dx + q\,dz = \left[pF_p + qF_q\right]\left[\frac{dx}{F_p}\right]. \tag{A.8}$$

Dividing by $pF_p + qF_q$, we obtain an equation for $d\tau$ that completes the set of equations in (A.7). The result is

$$\frac{dx}{F_p} = \frac{dz}{F_q} = -\frac{d\tau}{pF_p + qF_q} = -\frac{dp}{F_x + pF_\tau} = -\frac{dq}{F_z + qF_\tau} \tag{A.9}$$

The determination of τ for each (x, z) is a characteristic curve in 3-D space. Here, however, we also solve for p and q. It is helpful here to introduce a parameter σ along the characteristic curves and write (A.9) as

$$dx/d\sigma = \lambda F_p, \quad dz/d\sigma = \lambda F_q, \quad d\tau/d\sigma = \lambda\left[pF_p + qF_q\right],$$
$$dp/d\sigma = -\lambda\left[F_x + pF_\tau\right], \quad dq/d\sigma = -\lambda\left[F_z + qF_\tau\right]. \tag{A.10}$$

Choosing $\lambda = 1/2$ makes σ an unphysical quantity with units of $[length]^2/time$. For our case, equation (A.1) becomes the eikonal equation

$$F = (\nabla\tau)^2 - \frac{1}{c^2} = p^2 + q^2 - \frac{1}{c^2} = 0, \tag{A.11}$$

and F_x, F_z, F_p, F_q, and F_τ become

$$F_x = ss_x, \quad F_z = ss_z, \tag{A.12}$$
$$F_p = p, \quad F_q = q, \quad F_\tau = 0,$$

where $s = 1/c$ and $p = \tau_x$, $q = \tau_z$. We now write the system (A.10) in the following vector notation

$$\frac{d\Sigma}{d\sigma} = \frac{d}{d\sigma}\begin{bmatrix} x \\ z \\ p \\ \tau \end{bmatrix} = \begin{bmatrix} p \\ q \\ ss_x \\ s^2 \end{bmatrix}, \tag{A.13}$$

and the quantity q can be found from the eikonal equation itself

$$q = \sqrt{s^2 - p^2}. \tag{A.14}$$

For a given ray, the above system can be integrated in order to obtain the ray quantities as a function of the ray parameter σ. An alternative choice is to treat the depth as the independent parameter in place of σ. This is a suitable choice for our work, because it provides us with an easier way to determine where we should terminate the ray since we have a better sense for depth z than the sense for the ray parameter σ. We recall from the second line of equation (A.13) that

$$\frac{dz}{d\sigma} = q. \tag{A.15}$$

Using the above equation, we can make the following change of variables for the component $dx/d\sigma$ (for example)

$$\frac{dx}{d\sigma} = \frac{dx}{dz}\frac{dz}{d\sigma} = q\frac{dx}{dz}. \tag{A.16}$$

Applying this change of variables to the rest of the components in the vector $d\Sigma/d\sigma$, will allow us to write the system as

$$\frac{d\mathbf{D}}{d\sigma} = \frac{d}{dz}\begin{bmatrix} x \\ \sigma \\ p \\ \tau \end{bmatrix} = \frac{1}{q}\begin{bmatrix} p \\ 1 \\ ss_x \\ s^2 \end{bmatrix}, \tag{A.17}$$

where $\mathbf{D}$ denotes the reduced version of $(x, \sigma, p, q, \tau)^T$. The quantity q is again given by equation (A.13). The initial values of the ray quantities at the source location, denoted by the vector $\mathbf{D}_0$, are independent of the initial depth while they can be functions of the take-off angle β

$$\mathbf{D}_0(\beta) = \begin{bmatrix} x_0 \\ \sigma_0 \\ p_0 \\ \tau_0 \end{bmatrix} = \begin{bmatrix} x_0 \\ 0 \\ s_0 \sin\beta \\ 0 \end{bmatrix}, \quad q_0 = s_0 \cos\beta. \tag{A.18}$$

APPENDIX B

In this appendix I will derive the in-plane 2.5D amplitude given by equation (2.1.7). The ray equations in terms of the depth z in 3-D isotropic inhomogeneous media can be written as

$$\frac{dx}{dz} = \frac{p}{q}, \qquad \frac{dy}{dz} = \frac{r}{q},$$

$$\frac{dp}{dz} = \frac{1}{2q}\frac{\partial}{\partial x}\left[\frac{1}{c^2}\right], \quad \frac{dr}{dz} = \frac{1}{2q}\frac{\partial}{\partial y}\left[\frac{1}{c^2}\right], \quad \frac{dq}{dz} = \frac{1}{2q}\frac{\partial}{\partial z}\left[\frac{1}{c^2}\right], \tag{B.1}$$

where $p=\tau_x$, $r=\tau_y$, $q=\tau_z$, $c=c(x,y,z)$ is the propagation speed, and the initial conditions are given by

$$x(0)=0, \qquad y(0)=0,$$

$$p_0=\frac{1}{c_0}\sin\beta\cos\alpha, \quad q_0=\frac{1}{c_0}\cos\beta, \quad r_0=\frac{1}{c_0}\sin\beta\sin\alpha. \tag{B.2}$$

where c_0 is the propagation speed at the source location. The 3-D Jacobian of transformation from the Cartesian coordinates (x,y,z) to the coordinates (α,β,z) becomes

$$J_3 = \frac{\partial(x,y,z)}{\partial(\alpha,\beta,z)} = \det\begin{bmatrix} \frac{\partial x}{\partial\alpha} & \frac{\partial x}{\partial\beta} & \frac{\partial x}{\partial z} \\ \frac{\partial y}{\partial\alpha} & \frac{\partial y}{\partial\beta} & \frac{\partial y}{\partial z} \\ \frac{\partial z}{\partial\alpha} & \frac{\partial z}{\partial\beta} & \frac{\partial z}{\partial z} \end{bmatrix} = \det\begin{bmatrix} \frac{\partial x}{\partial\alpha} & \frac{\partial x}{\partial\beta} & \frac{\partial x}{\partial z} \\ \frac{\partial y}{\partial\alpha} & \frac{\partial y}{\partial\beta} & \frac{\partial y}{\partial z} \\ 0 & 0 & 1 \end{bmatrix}, \qquad \text{(B.3)}$$

where α is the azimuth angle of the initial slowness vector and β is the take-off angle from the vertical at the source location. The above determinant reduces to the determinant,

$$J_3 = \det\begin{bmatrix} \frac{\partial x}{\partial\alpha} & \frac{\partial x}{\partial\beta} \\ \frac{\partial y}{\partial\alpha} & \frac{\partial y}{\partial\beta} \end{bmatrix} = \frac{\partial(x,y)}{\partial(\alpha,\beta)} = \frac{\partial x}{\partial\alpha}\frac{\partial y}{\partial\beta} - \frac{\partial x}{\partial\beta}\frac{\partial y}{\partial\alpha}. \qquad \text{(B.4)}$$

1. Far Field Amplitude :

Equation (B.4) gives the full Jacobian of transformation in 3-D media. Since we are interested in 2.5D media, we assume that the propagation speed is only a function of x and z , $c = c(x,z)$. Given that the speed c is independent of y and using equation (B.1), we write

$$\frac{dr}{dz} = 0,$$

which implies that r is independent of the depth z and

$$r = r_0(\alpha, \beta) = \frac{1}{c(\mathbf{x}_s)} \cos\alpha \sin\beta, \tag{B.5}$$

where $\mathbf{x}_s$ is the source location which corresponds to z_0 in equation (2.1.5). From equation (B.1) again, y can be computed by integrating dy/dz

$$y(z, \alpha, \beta) = r_0(\alpha, \beta) \int_0^z \frac{d\zeta}{q(\zeta, \alpha, \beta)}. \tag{B.6}$$

Since $d\sigma/dz = 1/q$, recall equation (2.1.5), we can express σ in terms of the integral

$$\sigma(z, \alpha, \beta) = \int_0^z \frac{d\zeta}{q(\zeta, \alpha, \beta)}, \tag{B.7}$$

and (B.6) becomes

$$y(z, \alpha, \beta) = \frac{1}{c(\mathbf{x}_s)} \cos\alpha \sin\beta \, \sigma(z, \alpha, \beta). \tag{B.8}$$

If we differentiate the above equation with respect to α and β and evaluate equation (B.4) at $\alpha=0$ we get the following equation for the in-plane 2.5D Jacobian of transformation

$$J_{2.5} = \frac{\sin\beta}{c(\mathbf{x}_s)} \sigma \left| \frac{\partial x}{\partial \beta} \right|. \tag{B.9}$$

The in-plane far field amplitude in 2.5D smooth media is related to $J_{2.5}$ (Bleistein 1984) as

$$A_{2.5}(z, \beta) = \frac{K}{\sqrt{J_{2.5}}}, \tag{B.10}$$

where K is some constant that depends on the parameter β. The aim now is to evaluate the constant K.

2. Near Field Amplitude :

Assuming a homogeneous speed distribution near the source we can write the ray equations in equation (B.1) as

$$\frac{dx}{dz} = \frac{p_0}{q_0}, \qquad \frac{dy}{dz} = \frac{r_0}{q_0}. \tag{B.11}$$

Since p_0, r_0, and q_0 are independent of z, the above equations integrate to

$$x = \cos\alpha \sin\beta\, z, \qquad y = \sin\alpha \tan\beta\, z. \tag{B.12}$$

Differentiating with respect to α and β and setting $\alpha=0$ yields

$$J_s = \left[\frac{\partial x}{\partial \alpha}\frac{\partial y}{\partial \beta} - \frac{\partial x}{\partial \beta}\frac{\partial y}{\partial \alpha}\right]_{\alpha=0} = \sec^2\beta \tan\beta\, z^2, \tag{B.13}$$

where J_s is the Jacobian near the source. From equation (B.2) and (B.12) we can write

$$R^2 = x^2 + y^2 + z^2 = z^2\left[1 + \frac{x^2 + y^2}{z^2}\right] \tag{B.14}$$

$$= z^2\left[1 + \frac{p_0^2 + r_0^2}{q_0^2}\right] = z^2\left[\frac{1}{q_0^2 c_0^2}\right] = z^2\sec^2\beta.$$

Therefore equation (B.13) becomes

$$J_s = R^2\tan\beta. \tag{B.14}$$

This is used now to evaluate the in-plane amplitude near the source

$$A_s = \frac{K}{\sqrt{J_s}} = \frac{K}{\sqrt{R^2\tan\beta}} = \frac{K}{R\sqrt{\tan\beta}}. \tag{B.15}$$

Equating the above amplitude with the known amplitude for homogeneous speed distribution

$$A_s = \frac{1}{4\pi R}, \tag{B.16}$$

yields an expression for K given by

$$K = \frac{\sqrt{\tan\beta}}{4\pi} \tag{B.17}$$

The in-plane amplitude (ray-theoretical) in 2.5-D isotropic continuous medium is now given by

$$A(z,\beta) = \frac{K}{\sqrt{J_{2.5}}} = \frac{1}{4\pi}\sqrt{\frac{c(\mathbf{x}_s)\tan\beta}{\sin\beta\ \sigma(z,\beta)\ |\partial x/\partial\beta|}},$$

$$= \frac{1}{4\pi}\sqrt{\frac{c(\mathbf{x}_s)\sec\beta}{\sigma(z,\beta)\ |\partial x/\partial\beta|}}. \qquad \text{(B.18)}$$