GEOLOGICAL AND SEISMIC MODELING IN 3D

V. Pereyra
Weidlinger Associates
4410 El Camino Real #110
Los Altos, CA 94022 USA

RESUMEN

Discutiremos brevemente puntos básicos de la construcción y parametrización de modelos geológicos en medios complejos tri-dimensionales. Luego indicaremos como trazar rayos sísmicos en ésos modelos, incluyendo el cálculo de amplitudes. Estas herramientas se usan en varios aspectos de prospección petrolera por métodos sísmicos, tales como: planeamiento de levantamiento de datos, prueba de hipótesis, construcción de mapas en profundidad a partir de mapas de tiempos de viaje, y cálculo de tablas de tiempos de viaje para migración Kirchoff.

ABSTRACT

We will briefly discuss basic issues in the construction and parametrization of geological models in complex three-dimensional media. Then we will indicate how to trace seismic rays in such models, including the accurate calculation of amplitudes. These are tools used in various aspects of seismic oil prospecting, such as survey planning, hypothesis testing, time to depth mapping, and the creation of travel time tables for Kirchoff migration.

1 INTRODUCTION

We will discuss some issues related to the construction of geological models of the interior of the earth and on the simulation of elastic wave propagation processes on such models. The target application is oil exploration by seismic methods, but these techniques and tools apply equally well to reservoir characterization, reservoir administration, and the monitoring of enhanced recovery processes.

The concepts apply also to the modeling of many quantities related to physical properties of rock masses and by changes in the scale, they can apply to earthquake and engineering seismology, to the imaging and simulation of contamination plumes, and even to medical imaging and nondestructive inspection of machine pieces.

This project, that started more than 20 years ago, has focused in the early 80's on complex three-dimensional geology. That involves not only stratigraphy (i.e. geometry of material discontinuities), but also volumetric variations in velocity of propagation of elastic waves, in media that can be isotropic or anisotropic.

In this short paper we will indicate some of the basic points in our approach to model parametrization and construction, and to seismic ray tracing in such models. Ray tracing is an economical way to simulate some important aspects of wave propagation that are relevant to seismic exploration.

2 THREE-DIMENSIONAL MODELING

Our aim is to represent simple models easily, without limiting our ability to represent complexities such as salt domes with overhangs, normal and reverse faults, pinched out layers, lenses, folds and overthrusts.

To do this we introduce the concept of *blocky models* to represent an earth volume. We concentrate on elastic models, although clearly the methodology applies equally well to other types of material properties.

A blocky model describes an earth volume as an aggregate of irregularly shaped sub-volumes of "like materials" separated by material interfaces. The sub-volumes or regions will contain slowly and smoothly varying inhomogeneous materials, while the interfaces provide an explicit representation for those surfaces that bound sharply different rock types.

Both interfaces and inhomogeneous volumetric material properties within regions are modeled parametrically, using a variety of analytical representations. Namely, Coons, B-spline, and composite surface patches; for material properties we consider generalized Coons volumes, tensor product B-splines, and composite material functions. These classes of objects are by no means exhaustive, and other types can be easily added to the system as required. Representations on appropriate wavelet bases and digitized surfaces and volumes are good candidates for future development.

The main characteristics that distinguish this modeling paradigm is the use of smooth surface macro-patches to represent material interfaces, and the ability to represent material properties in irregularly shaped regions by means of smooth functions that conform to the boundaries, thus eliminating the typical staircase approximations that result when using Cartesian meshes. This approach also eliminates the need for using zero-thickness layers, counter-intuitive set theoretical operations to define complex regions, or irregular meshes that need to be refined around sharp variations in material properties.

Of course, the main advantages of this representation are that it has all the smoothness required to perform ray tracing and that is concise, requiring only a few hundred words to describe fairly complex models. This is not shared by mesh or triangulation based representations that require millions of words to describe material properties and interfaces and have no continuity at all. Conciseness is important in terms of data compression to store and manipulate these models, and even more so at inversion time, where the number of parameters will dictate the type of algorithms required and decide if the solution of the problem is feasible at all.

We require that surface patches have only one kind of material region on each side. Thus, a surface patch is formally defined by a type, a region *above* and a region *below* pointer, and a twice differentiable analytic function. Patches can be stitched together with or without (as in the case of a fault edge) derivative continuity.

For the material properties we consider four types of functional representations:

- Constant; for homogeneous blocks.
- Gradient in an arbitrary direction.
- Explicit tensor product of B-splines.
- Composite Coons-B-spline volumes.

In isotropic media, the properties to be represented are the velocity of propagation of P and S waves and the density. These quantities can be modeled independently or, alternatively, we can assume a constant Poisson's ratio and define the S velocity and density as functions of the P velocity.

Anisotropic properties can also be considered; the system currently includes the case of piecewise constant, transversely isotropic materials (with an arbitrarily oriented plane of isotropy), which requires five elastic parameters.

3 SEISMIC RAY TRACING IN ISOTROPIC BLOCKY THREE-DIMENSIONAL MEDIA

Seismic ray tracing in 2D complex media or 3D layered media is a well understood process. Here we will indicate briefly how to handle the additional difficulties that three-dimensional isotropic blocky media presents.

For wave propagation in isotropic media, rays are the orthogonal trajectories to wave fronts. If the media is homogeneous, rays are straight lines, while for inhomogeneous media, ordinary differential equations have to be integrated in order to accurately calculate the ray trajectories. These are the so called ray equations, which can be derived either from the Eikonal equation, or by invoking Fermat's principle of minimum time.

A convenient form of the ray equations in 3-D is:

$$\begin{aligned} \dot{\boldsymbol{\eta}} &= v\mathbf{w} \\ \dot{\mathbf{w}} &= \nabla u, \end{aligned} \tag{1}$$

where $\boldsymbol{\eta} = (x(s), y(s), z(s))$, $\dot{\mathbf{w}}(s) = u(\boldsymbol{\eta})\boldsymbol{\eta}, s$ is arc length along the ray, and $u = 1/v$ is the slowness, with v the velocity of propagation. Observe that since s is arc length, then $\| \dot{\boldsymbol{\eta}}(s) \|_2 = 1$, and therefore $\mathbf{w}$ is a vector in the ray direction with length equal to u. That is why this vector is sometimes referred to as a *slowness vector.*

The simplest form of ray tracing is shooting, in which the initial position, and the initial direction of the ray are prescribed:

$$\boldsymbol{\eta}(0) = \boldsymbol{\eta}_0, \quad \mathbf{w}(0) = \mathbf{w}_0. \tag{2}$$

We use shooting only as a vehicle to initialize a source-receiver, global or bending type, iterative calculation, or to check a posteriori if a two-point ray has changed signature (a ray signature is an ordered sequence of reflecting interfaces). This combination of shooting and bending was first employed (to the best of our knowledge) in two-dimensions in 1983, and published in (Pereyra, 1987).

Equations (1, 2) describe an initial value problem that can be solved numerically by a standard technique. In fact, for smooth velocity fields, these equations present no special problems. However, we are interested in models that involve velocity discontinuities and complex geological models. Let us assume that the ray intersects r interface patches $I_1, ..., I_r$, in its travel from the source $\boldsymbol{\eta}_0$ to its destination, through regions $R_1, ..., R_{r+1}$; i.e., it has (r+1) continuous segments.

At an interface between two different materials, the ray changes direction discontinuously according to Snell's law. When refraction of the ray occurs at $\mathcal{H}_\ell$, we have that

$$\mathbf{w}_\ell^+ = \mathbf{w}_\ell^- + \left\{ \sqrt{(u_\ell^+)^2 - (u_\ell^-)^2 + < \mathbf{w}_\ell^-, \boldsymbol{\mu}_\ell >^2} - < \mathbf{w}_\ell^-, \boldsymbol{\mu}_\ell > \right\} \boldsymbol{\mu}_\ell. \tag{3}$$

When reflection occurs, the following simplified formula applies instead:

$$\mathbf{w}_\ell^+ = \mathbf{w}_\ell^- - 2 < \mathbf{w}_\ell^-, \boldsymbol{\mu}_\ell > \boldsymbol{\mu}_\ell, \tag{4}$$

since $u_\ell^- = u_\ell^+$ in that case. Here $< .,. >$ represents the usual vector inner product, and $\boldsymbol{\mu}_\ell$ is the normal vector to the surface $\mathcal{H}_\ell$ at the point $\boldsymbol{\eta}(s_\ell)$. We assume that $\|\boldsymbol{\mu}_\ell\|_2 = 1$. Type conversion may also enter into consideration at this stage; this is simply handled by defining u_ℓ^-, u_ℓ^+, etc., in terms of the appropriate wave type velocity (v_s or v_p) before and after contact with the surface $\mathcal{H}_\ell$.

Since the ray has met r interfaces we need $r + 1$ conditions to determine the partial arc lengths of the legs: $s_\ell - s_{\ell-1}$, $\ell = 1, \ldots, r + 1$. One such condition is that s should actually be arc length, which is guaranteed by

$$\|\dot{\boldsymbol{\eta}}(0)\|_2^2 = 1 \Leftrightarrow v(\boldsymbol{\eta}(0))^2 \|\mathbf{w}(0)\|_2^2 = 1. \tag{5}$$

The remaining r conditions are given by explicitly requiring that $\boldsymbol{\eta}_\ell$ lies on the interface $\mathcal{H}_\ell$. If $\mathcal{H}_\ell$ is defined by an equation of the form:

$$\Psi_\ell(\boldsymbol{\eta}) = z - g_3(u(x, y), v(x, y)) = 0, \tag{6}$$

then these conditions are:

$$\Psi_\ell(\boldsymbol{\eta}_\ell) = 0, \qquad \ell = 1, \ldots, r. \tag{7}$$

Upon incidence, up to four different rays can be generated, accounting for reflections, transmissions and mode conversions. Although a tree of rays can be traced following all possible ray paths, we shall concentrate here on how to calculate only one ray for which a signature has been prescribed. Given a ray signature, the decision on which of the possible four ray paths to continue at each interface is made a priori. This does not require necessarily an explicit sequence of regions and patches, but only an unequivocal rule on which to base a decision. For instance, one can request a ray that starts at $\boldsymbol{\eta}_0 \in R_1$, and travels through the structure until it reaches interface I_d, reflects or transmits, and ends up at interface I_{r+1}; if any other interface is met we assume, for instance, that transmission should occur (if possible). The shooting scheme constructs the detailed, complete signature or ordered sequence of regions and patches traversed by the ray. Of course, this ray path must remain within the confines of the 3-D geological region under study. Mode conversions can easily be included at especified interfaces.

In summary, all that is required to perform this task in a blocky type model is the ability to know at any time in which region we are and to calculate the intersection of a curved or straight ray path with an interface patch. Since we know in which region we start (R_1), and have available sufficient connectivity information, once we identify that a patch has been crossed, we need only to turn on a zero-finder in order to get a sufficiently accurate intersection.

Since, by hypothesis, every patch has only one region on each side, there is no ambiguity with respect to the identity of the new region through which the ray is transmitting or reflecting. Once the intersection point has been obtained, the transmitted (or reflected) direction is calculated by Snell's law.

A particular difficulty in blocky media, not present in layered media, is that when a ray trajectory is being traced within a region there is no a priori knowledge of which patch is going to be intersected next: a region may be bounded by many patches and therefore some expensive checking must go on in order to guarantee that no model features are ignored and to determine which is the first patch traversed by the ray.

This problem can be alleviated by rasterizing the volume on a three-dimensional mesh, and assigning to each brick in that mesh a code indicating if the element is fully contained on a given region, or if it is in contact with a patch (or patches). Since it is very easy to locate the element to which a given (x, y, z) belongs, we can interrogate this data base to navigate the model rapidly. The rasterization process is implemented in a module called GEOMAP. We have a version of the ray tracer that takes good advantage of this information.

Since we are not interested in shot rays per se, but only as a means to initialize a two-point iteration for solving source-receiver problems, it is not necessary to calculate them to high precision. Observe that, even in the case of regions of constant material where no integration is necessary, one still needs to calculate the ray/interface intersection and needs to have in place all the information and logic related to the connectivity of the model.

Given $\boldsymbol{\eta}_0, \mathbf{w}_0$, a model description, and a ray signature, the shooting algorithm attempts to produce a ray that starts at $\boldsymbol{\eta}_0$ with direction $\mathbf{w}_0$, travels through the structure, honoring ray bending in inhomogeneous regions and Snell's law at interface crossings, respects the specified signature and arrives at the receiver area. If successful, we will have obtained a discrete ray:

$$(s_i, \boldsymbol{\eta}_i, \mathbf{w}_i), \quad i = 1, ..., N.$$

The shooting algorithm produces discrete rays with the same format as the ones required by the two-point solver described below. Thus, when a shot ray lands near a receiver it can be used directly to start a two-point iteration. A detailed ray signature is also produced; this is now an ordered sequence of regions R_ℓ and patches P_ℓ traversed by the rays that is needed by the two-point solver. By this procedure, the two-point solver is made essentially independent of the structural complexity of the model.

A control screen is used to automate the process of generating all the arrivals from a source to an array of receivers. This control screen, in its simplest version, consists of a number of equally spaced points on a rectangle parallel to the (x, y) plane, at some depth $z_{screen} > 0$.

Joining the source with the points in the control screen (sometimes referred to as pixels), produces initial directions for shooting rays. Thus, by defining the control screen appropriately, one can create and direct at will a discrete beam of rays, and also keep track of inspected directions.

A general two-point boundary value approach for source-receiver ray tracing in inhomogeneous layered media has been reported earlier in detail in (Pereyra, 1988, 1992). A multipoint boundary value finite difference solver for nonlinear systems of first order ordinary differential equations is used (Lentini and Pereyra, 1983). This solver has variable order, variable mesh, and global error estimation capabilities, combined to provide an accurate, efficient and robust algorithm, well suited for high resolution work. Versions adequate for solving two point boundary value problems in smooth inhomogeneous media are available in the public domain (IMSL, Harwell, NAG libraries, or through the electronic Numerical Analysis Network na.net).

The ray equations (1) are discretized on a mesh (not necessarily uniform) by the second order trapezoidal rule. We make sure that discontinuities, i.e., patch crossings, occur at mesh points where appropriate discontinuity conditions are enforced. Global error estimates, adaptive meshes and adaptive order through deferred corrections are used to obtain a solution with a prescribed accuracy in an efficient manner.

The finite difference process used is of global type and it does not suffer from the instabilities associated with shooting schemes. An efficient Newton type nonlinear equation solver, which includes a carefully crafted sparse linear solver, is used on the resulting nonlinear difference system. The sparse structure is such that a perfect elimination stable algorithm can be devised; i.e., no fill-in is produced in the Gaussian elimination process.

Discontinuities, additional algebraic conditions and unknown parameters are also handled by our current version. The linear equation solver produces a sparse triangular decomposition (LU), which is a discrete version of the linearized ray equations. This is quite useful for performing economically a number of additional tasks, like calculation of 3D geometrical spreading, sensibility studies, and nonlinear travel time inversion or geophysical tomography with bent rays; see also (Pereyra, 1980, Pereyra, Keller, and Lee, 1980, Pereyra, 1988).

In summary, this algorithm has the necessary generality to solve the ray equations (1) subject to the end conditions:

$$\boldsymbol{\eta}(0) = \boldsymbol{\eta}_{SOURCE}, \quad \boldsymbol{\eta}(S) = \boldsymbol{\eta}_{RECEIVER}, \tag{8}$$

(where S is the (unknown) total arc-length). It can also handle the additional interface conditions and of course many other similar problems.

Of course, the seismic ray tracing task in geophysics never consists of calculating an isolated ray, but rather, for a given shot location (in the case of non-zero offset ray tracing), one needs to calculate all possible arrivals with a prescribed signature for a given array of receivers. This array may consist of just one line of equally spaced receivers, as in the case of 2D surveys, or of a number of lines, either in a regular array or in more general positions, or they can be underground on wells. In any case, our algorithm takes into account this fact to accelerate the calculation by using a so-called receiver continuation strategy.

Receiver continuation is a technique that exploits the fact that if we have calculated a ray path joining a source with a receiver, then this ray can be used to initialize the two-point or bending calculation for a neighboring receiver position. In this way, the shooting exploratory phase is limited to finding the first ray that arrives near the receiver array, which is then used to initiate a sequence of two-point ray calculations by receiver continuation.

A naive implementation of this simple idea would stop here and would generally fail to calculate all possible arrivals, since the two-point continuation will not be feasible if it tries to move through caustics or into shadow zones, and may also fail for other physical or computational reasons. What makes our procedure robust is that as soon as the two-point continuation fails, the algorithm switches to shooting in order to find another starting ray, and this search can be made as fine and extensive as desired by choosing an appropriate control screen, which also aids us in keeping track of the work done.

Normal incidence or zero offset ray tracing is also easily implemented within this framework. This type of ray tracing is used to simulate stacked sections. Our implementation calculates only one half of the trajectory, say from the coincident source/receiver position to the reflector, since the return ray must retrace the same path. The normal incidence on the reflector is enforced as a new type of boundary condition. Both zero and non-zero offset diffracted ray paths from designated edges can be also calculated.

Many of these tasks are amenable to coarse grain parallelization on a network or multi-CPU setting, as we have demonstrated in (Koshy, Pereyra, and Meza, 1991).

References

[1] Koshy, M., Pereyra, V., and Meza, J.C. (1991), *Distributed Computing Applications in Forward and Inverse Geophysical Modeling.* SEG 61st Annual Meeting, Houston, TX.

Extended Abstracts, 349-352.

[2] Lentini, M., and Pereyra, V. (1983), *PASVA4: An Ordinary Boundary Solver for Problems with Discontinuous Interfaces and Algebraic Parameters,* Matematica Aplicada e Computacional, 2, 103-118.

[3] Pereyra, V., and Scherer, G. (1973), *Efficient Computer Manipulation of Tensor Products with Applications to Multidimensional Approximation,* Math. Comp., 27, 595-605.

[4] Pereyra, V. (1980), *Two-Point Ray Tracing in Heterogeneous Media and the Inversion of Travel Time Data,* In *Computational Methods in Applied Science and Engineering* (eds. Glowinski R. and J. L. Lions.) North-Holland, Amsterdam, 553-570.

[5] Pereyra V., Keller, H.B., and Lee, W.H.K. (1980), *Computational Methods for Inverse Problems in Geophysics. Inversion of Travel Time Observations,* Phys. Earth Plan. Int., 21, 120-125.

[6] Pereyra, V. (1987), *Modeling with Ray Tracing in Two-Dimensional Curved Homogeneous Layered Media,* In *Micro-Computers in Large Scale Scientific Computation* (ed. Wouk, A.) (SIAM Pub., Philadelphia) 39-67.

[7] Pereyra, V. (1988), *Numerical Methods for Inverse Problems in Three-Dimensional Geophysical Modeling,* Applied Numerical Mathematics, 4, 97-139.

[8] Pereyra, V. (1992), *Two Point Ray Tracing in General 3D Media*, Geophysical Prospecting, 40, 267-287.