COMPUTING METHODS IN APPLIED SCIENCES AND ENGINEERING
R. Glowinski, J.L. Lions (editors)
North-Holland Publishing Company

TWO-POINT RAY TRACING IN HETEROGENEOUS MEDIA AND THE INVERSION OF TRAVEL TIME DATA

V. Pereyra
Escuela de Computacion
Facultad de Ciencias
Universidad Central de Venezuela
Apartado 59002, Caracas 104

1. Introduction

A general algorithm and its corresponding computer implementation has been devised to solve both direct and inverse problems involving ray tracing of elastic waves in heterogeneous, isotropic, piecewise smooth, two or three dimensional media.

The practical applications require the tracing of rays between a source (earthquake, explosion) and a receiver (seismometer, geophone). The velocity distributions we admit can be arbitrarily varying both in depth and laterally. Thus the resulting nonlinear two-point boundary value problem must be solved numerically. For this purpose we use a general adaptive finite difference code which can compute two-point rays to any desired accuracy. This code has provisions which facilitate the solution of the inverse problem.

Given arrival time data, we consider three kinds of inverse problems: calculation of velocity distributions, relocation of hypocenters, and determination of reflectors.

Some numerical examples using artificial data are presented.

2. Two-Point three-dimensional ray-tracing in heterogeneous, piecewise smooth media.

We are interested in studying the propagation of elastic waves in an heterogeneous, isotropic, three-dimensional medium. We assume that this medium is described by giving the velocity of propagation of either pressure (P) or shear (S) waves.

The geometry of the medium must also be given, specifying any interfaces across which the velocity is discontinuous. We allow also for switching from P to S waves, or viceversa, when passing across interfaces.

We assume finally that the phenomena under study is such that ray theory approximation is applicable, and seek to compute rays between a source and a receiver.

Rays can be described as stationary paths of certain functionals. The corresponding Euler-Lagránge equations for this calculus of variations problem are

$$\frac{d}{ds}\left[u(\underset{\sim}{\eta})\frac{d}{ds}\underset{\sim}{\eta}\right] - \nabla u = 0 \ , \tag{2.1}$$

where $\underset{\sim}{\eta} = (x,y,z)$ represents the ray and $u(\underset{\sim}{\eta}) \equiv 1/v(\underset{\sim}{\eta})$, with $v(\underset{\sim}{\eta})$ the velocity of propagation in the medium. We discuss first the case in which $v(\underset{\sim}{\eta})$ is smooth, say in C^4.

We add the constraint

$$\dot{x}^2 + \dot{y}^2 + \dot{z}^2 = 1 \tag{2.2}$$

which ensures that the independent variable s is arc-length along the ray.

The two-point ray tracing problem consists of solving these equations subject to the boundary conditions

$$\underset{\sim}{\eta}(0) = P_S \ , \ \underset{\sim}{\eta}(S) = P_R \ ,$$

where P_S, P_R are the given Cartesian coordinates of the source and receiver respectively, and S is the total length of the arc of ray joining P_S and P_R. Obviously S is also unknown.

In the past, this problem has been solved mainly by shooting techniques (see [3,6,7]), taking advantage of available initial value problem software. Some researchers [8,16] have also looked into global methods based on finite differences.

However, it has not been until more recent times that software implementing versatile finite difference methods for nonlinear two-point boundary value problems have become available (see [12,14]). Is this software that has allowed us [15,17] to solve the two-point ray tracing problem routinarely and in a variety of situations. It will also allow us to attack the inverse problem in a more precise form than it has been done in the past, as we will show in Sections 3 and 4.

Our finite difference solver PASVA3 requires that the differential system be of first order. A few manipulations permit to transform (2.1) to the first order form

$$\dot{\omega}_i = \omega_{i+1}$$

$$(2.3) \qquad \dot{\omega}_{i+1} = v(-G(\underset{\sim}{\omega})\omega_{2i} + u_{\omega_{2i-1}})$$

$$i=1,2,3 \ ,$$

where $\underset{\sim}{\omega} = (x,\dot{x},y,\dot{y},z,\dot{z})$, and

$$G(\underset{\sim}{\omega}) = u_x\omega_2 + u_y\omega_4 + u_z\omega_6 \ .$$

Observe that these equations are very nonlinear as soon as $v(\omega_1,\omega_3,\omega_5)$ is non-constant.

Many other forms are possible, with various advantages and disadvantages. A very lucid and detailed account of this fascinating subject can be found in [3].

Unfortunately, system (2.3) is still not in a form appropriate for PASVA3, which requires a quite definite interval of integration to be given. We see that (2.3) is a free boundary problem, since the right end point S (in the independent variable s) is unknown. This is easily fixed by introducing the change of independent variable $s \to t \equiv s/S$, a new dependent variable $\omega_8 \equiv S$, and the artificial differential equation $S' = 0$. Now ' will denote differentiation with respect to t. The changes in (2.3) only involve multiplication of the right hand side of the equations by S, and the new interval of integration is simply [0,1].

Finally, since the quantity we are really interested in is the travel time

$$(2.4) \qquad T = S \int_0^1 u \, dt \ ,$$

we introduce still one more variable $\omega_7(t)$, for the partial travel time, and the corresponding differential form of (2.4)

$$(2.5) \qquad \omega_7' = Su \ ,$$

together with the two extra boundary conditions $\omega_7(0) = 0$ and $\omega_2^2(0) + \omega_4^2(0) + \omega_6^2(0) = 1$. This latest condition is all what is necessary in order to enforce the constraint (2.2), as is shown in [15].

In the case of ray tracing in two space dimensions, further simplifications allow us to write the corresponding boundary value problem as

$$\begin{aligned}
\omega_1' &= \omega_4 \cos \omega_3 \\
\omega_2' &= \omega_4 \sin \omega_3 \\
\omega_3' &= \omega_4\, v(\omega_1,\omega_2) \left[u_z \cos \omega_3 - u_x \sin \omega_3 \right] \qquad (2.6) \\
\omega_4' &= 0 \\
\omega_5' &= \omega_4\, u
\end{aligned}$$

with the boundary conditions

$$\omega_1(0) = x_S, \quad \omega_2(0) = z_S, \quad \omega_5(0) = 0,$$
$$\omega_1(1) = x_R, \quad \omega_2(1) = z_R \; .$$

In order to consider the piecewise continuous case, let $\phi(x,y,z) = 0$ be a given interface. We will use Snell's law to derive the condition that the ray must satisfy when meeting this interface [3,11].

In general we will have two possibilities: either the ray is transmitted from one side to the other of the interface, or it is reflected by it.

Rays coming to the interface at a critical angle can be difracted and travel along the interface with the velocity of the fastest medium. We shall see that our formulation takes this fact into account automatically.

Let $\underset{\sim}{\eta}^*$ be the incident point, let $N(\eta^*)$ be the unit normal vector to the interface $\phi(\eta^*) = 0$, pointing from $\underset{\sim}{\eta}^*$ towards the first medium S_I (that contains the incident ray). Let $I(\underset{\sim}{\eta}^*)$ be the unit tangent vector to the ray, pointing from $\underset{\sim}{\eta}^*$ towards S_I. Finally, let α_I be the angle between N and I, let α_R be the angle between the reflected ray and N, and let α_T be the angle between the transmitted ray and N.

Finally, let n be the index of refraction at $\underset{\sim}{\eta}^*$: $n = v_T(\underset{\sim}{\eta}^*)/v_I(\underset{\sim}{\eta}^*)$, where the subindeces I,T indicate velocities in the first and second medium respectively.

Since the reflected or transmitted ray must lie in the plane determined by I and N, then it follows from the law of reflection and Snell's law that:

(2.8) $$R = -I + 2\cos\alpha_R N ,$$

and

(2.9) $$T = -n\, I + (n\cos\alpha_I + \cos\alpha_T)N .$$

These are actually three conditions each for the direction of the reflected and transmitted ray. But because of our equations, we already know that the tangent vector to the ray $(\dot{x},\dot{y},\dot{z})$ will be normalized, so it is enough to give two of these conditions in order to determine the proper exiting direction for the ray.

After some manipulation, these two pairs of conditions become:

(2.10) $$\begin{aligned} \cos\alpha_I &= \cos\alpha_R \\ \langle I,R\rangle &= \cos 2\alpha_I , \end{aligned}$$

and

(2.11) $$\begin{aligned} -\{1-n^2\sin^2\alpha_I\}^{1/2} &= \cos\alpha_T \\ \langle I,T\rangle &= -n\sin^2\alpha_I + \cos\alpha_R\cos\alpha_I . \end{aligned}$$

Translating into our original variables we find that

$$I = -(\dot{x}_I,\dot{y}_I,\dot{z}_I) ,$$

$$N = (\phi_x,\phi_y,\phi_z)/\sqrt{\phi_x^2+\phi_y^2+\phi_z^2} ,$$

$$R = T = (\dot{x}_{II},\dot{y}_{II},\dot{z}_{II}) ,$$

$$n = v_{II}/v_I ,$$

where I, II indicate the two mediums and all functions are evaluated at $\underset{\sim}{\eta}^*$.

Since we want to use the solver PASVA3 without changes, we have to perform some transformations in order to accomodate this discontinuous problem. We will not spell out the details here, since that has already been done in [15], but we will briefly indicate the idea.

We consider the two smooth segments of the ray between the so and the point $\underset{\sim}{\eta}^*$ at the interface, and between $\underset{\sim}{\eta}^*$ and the recei For each segment we consider the ray equations and thus we end up w double the number of differential equations. We have in an obvious manner ten boundary conditions. The remaining six are provided by three continuity conditions that the ray must satisfy at $\underset{\sim}{\eta}^*$, plus two discontinuity conditions on the ray direction that we discussed above, and finally the condition that $\underset{\sim}{\eta}^*$ be actually on the interf $\phi(\underset{\sim}{\eta}^*) = 0$.

Obviously this process can be repeated for any number of inte face contacts, although it may become increasingly inefficient. The PASVA3 linear equation solver can be modified so that only one set o equations is solved for the whole ray, while the discontinuity condi tions are worked in explicitly. The only thing one has to make sure is that the jumps occur always at mesh points [10] .

3. The finite difference solver PASVA3

In this section we will briefly discuss some aspects of a gen program for solving nonlinear two point boundary value problems of tl form

$$\omega' = f(t,\omega) \;, \quad g(\omega(a),\, \omega(b)) = 0 \;, \tag{3.1}$$

where ω, f, g are smooth vector valued functions, and problem (3.1) i assumed to have an isolated solution $\omega^*(t)$.

PASVA3 [12,14,15] is a FORTRAN implementation of an adaptive finite difference method to solve problem (3.1). It has both the capability of adapting the mesh in a non uniform way and of changing the order of the approximating method by deferred corrections [13]. It produces asymptotic global error estimates and it has been extensively tested in a variety of applications.

Given a mesh $\pi = \{t_i:\ a=t_1 < \ldots < t_n = b\}$, the trapezoidal rule is used to discretize (3.1):

$$\omega_{i+1} - \omega_i = \frac{h_i}{2}(f_i + f_{i+1}) \;, \quad i=1,\ldots,n-1, \qquad g(\omega_1,\omega_n) = 0 \;, \tag{3.2}$$

where $h_i = t_{i+1} - t_i$, ω_i is sought to approximate $\omega^*(t_i)$, and $f_i = f(t_i,\omega_i)$.

This system of nonlinear equations is solved by a variant of Newton's method. Thus, the Jacobian matrix

$$f_\omega = \left(\frac{\partial f^i}{\partial \omega_j}\right)_{i,j=1,\ldots,m} ,$$

where f^i is the ith component of the vector function $f(t,\omega)$, is required. From it, a linearized form of (3.2) is constructed and solved by a stable LU factorization process that preserves sparseness by careful alternate pivoting. This linearized form amounts to a discretization of the variational equation associated with (3.1)

$$\text{(3.3)} \qquad \begin{aligned} \delta' &= f_\omega(t,\omega)\,\delta + r(t) \ , \\ g_{\omega(a)}\,\delta(a) + g_{\omega(b)}\,\delta(b) &= \beta \ . \end{aligned}$$

It is important to observe that once the Jacobian f_ω has been evaluated on the mesh π, and the matrix associated with the linearized discrete equations has been constructed and LU decomposed, the actual solution of any system with such a matrix amounts to the solution of one upper and one lower block bidiagonal system, which even have some further sparseness in their off-diagonal blocks.

This fact is taken full advantage of in PASVA3, since both the error estimation and the deferred corrections can be performed without recomputing the Jacobians, and therefore they cost very little extra computation. We will see in the following section that this feature is also very useful for computing partial derivatives of the travel time T with respect to parameters.

4. Inversion of travel time data.

Both in seismology and in the reflection method of seismic prospecting one models the interior of the Earth by using travel time data. In seismology the source will usually be a natural event, i.e. an earthquake. In such a case, it will be also of interest sometimes to calculate the position (hypocenter) and origin time of the event. In seismic prospecting the events are man-made explosions, and therefore source and origin time are known.

The modelling can involve both the determination of the veloci structure of a portion of the Earth interior, and that of any special material features that may produce strong discontinuities in the velocity field. These abrupt changes of material will be called "reflectors" in what follows.

We will consider three inverse problems:

(a) the modelling of the velocity of elastic waves, (b) locat of hypocenters, and (c) detection of reflectors.

In a given application they may appear individually, by pairs, or all three together, and our program allows for any of these combin tions to occur.

Each travel time observation (arrival time in some cases) shoul be accompanied by the coordinates of a source-receiver pair, and by a description of what kind of a ray may have produced the observation. For simplicity of the exposition, in what follows we would think of th model having none or at most one interface. If there are no interface the only possible rays are direct ones, and the only choice is between P and S waves. For long distances we may have reflections on the free surface, which is itself an interface. Thus we would have to indicate the number of reflections and if there is any switching between P and S waves or viceversa.

In the case of an interior interface, like a fault, a synclinal or a salt dome, there are more possibilities that must be indicated. For more complicated structures or multiple reflected rays this task may become very difficult, and then the ray tracing code may be used to try to match one of the many possible rays with the observation. We will assume in what follows that the matching between rays and observations has been achieved, and that our task is only to compute accurat travel times with the direct ray tracing code. Having these computed travel times we will alter the parameters of the model in an iterative fashion, so that the whole set of observed times is best fitted in the least squares sense.

Therefore, let $\{T_i^0\}$, $i=1,\dots,\ell$ be a set of observed travel times and let $\{T_i^c\}$ be the corresponding set of times computed with our ray tracing procedure. We recall (see (2.5)) that the travel time is actually one of the dependent variables in the ray tracing equations Although it could have been integrated independently once the ray was obtained, it is much more convenient to include it with the other equations.

Both the velocity $v(\underset{\sim}{\eta})$ and the interfaces $\phi(\underset{\sim}{\eta}) = 0$ will be modelled by a finite number of parameters.

This will be achieved by choosing an appropriate finite dimensional set of approximating functions. The choice of this representation is in itself a very important and delicate problem. Some examples can be found in the literature [1,2] , and we will not dwell here further on this topic, except to point out that splines [18] are probably fairly good candidates in many cases, unless some specific analytic form is indicated.

So, let $\underset{\sim}{\mu}$ be the vector of unknown parameters. Our problem can be readily stated as

$$(4.1) \qquad \min_{\underset{\sim}{\mu}} \sum_{i=1}^{\ell} (T_i^c(\underset{\sim}{\mu}) - T_i^0)^2$$

subject to

$$(4.2) \qquad \begin{aligned} \dot{\omega}^i &= f(t, \omega^i, \underset{\sim}{\mu}) \\ g(\omega^i(0), \omega^i(1); \underset{\sim}{\mu}) &= 0 \ , \end{aligned}$$

where (4.2) represents the ray tracing equations (2.3), (2.5), and where we have displayed explicitly the dependence upon the parameter vector $\underset{\sim}{\mu}$. This will come about either in the equations, because some of the parameters are used to represent an unknown velocity, or in the boundary conditions (2.10-2.11), when an unknown reflector has to be determined.

Problem (4.1)-(4.2) can be solved by more or less standard non-linear least squares techniques [4,5,9] . The most effective ones will require the partial derivatives of the computed times with respect to the parameters. These derivatives are easily obtained by observing that if we solve system (3.3), with the linearization being taken around the computed ray $\omega^i(t)$, and with

$$(4.3) \qquad r(t) = \frac{\partial f}{\partial \mu_j} \ , \quad \beta = \frac{\partial g}{\partial \mu_j} \ ,$$

then

$$(4.4) \qquad \frac{\partial T_i^c}{\partial \mu_j} = \delta_7(1) \ .$$

As we mentioned earlier, the discrete solution of (3.3) only involves the solution of two very simple triangular systems. Methods using shooting techniques for solving the ray equations require the independent setting and solution of these linearized equations, and thus usually avoid this calculation and resort to some gross approximations to the partial derivatives (4.4) [1,2] .

We point out in passing that this same observation is valid if one is interested in the computation of the geometrical spreading of the rays, which also requires the solution of the linearized system (3.3) with appropriate right hand sides. Thus PASVA3 is very economical in performing all these tasks as compared to shooting techniques.

We finally observe that hypocenters and origin times can be located by the same procedure, the unknown parameters appearing now in the boundary conditions and in the definition of the travel time itself.

The nonlinear least squares problem (4.1) can be attacked by any of a number of available methods. However, it is necessary to recognize that the evaluation of the functional (4.1) is an expensive task, especially if the number of observation is large. Therefore, it is appropriate to choose a method that at each iteration takes full advantage of the evaluation of the functional and its gradient. After reviewing several candidates we have settled for a combination of methods due to Gill and Murray [5] and Jupp and Vozoff [4] .

The main task is to perform a singular value decomposition of the Jacobian matrix

$$J = \left(\frac{\partial T_i^c(\underset{\sim}{\mu})}{\partial \mu_j} \right)_{\substack{i=1,\ldots,\ell \\ j=1,\ldots,m}} .$$

This is the matrix of the linear least squares problems that result at each step of Gauss-Newton or Marquardt type methods. Although singular value decompositions are considered expensive, they provide a maximum of information which is specially useful in nearby rank deficient cases, a not uncommon occurrence in these inverse problems. Besides, compared to the ray tracing part, the cost is insignificant.

Our algorithm uses mainly the Gauss-Newton iteration with step control. If this fails to reduce the residual, then a Marquardt type correction is attained by damping some singular values and cutting off those below a pre-specified threshold.

5. Numerical examples.

We present now some numerical examples computed with the procedures just described. They are artificial examples, with the data generated by the ray tracing code and specific values of the modelling parameters. We show then details of the recovery of those parameters via the inversion code.

All the computations were performed in single precision on a Burroughs 6700 computer (≈ 11 decimal digits).

i) Recovery of velocity in two-dimensions.

We consider a two-dimensional geometry with the velocity given by a linear function of the coordinates

$$v(x,z) \doteq \mu_1 + \mu_2 x + \mu_3 z.$$

Thus the medium is both inhomogeneous vertically and laterally. The data is provided by five events at the known locations (measured in kms.):

Event	1	2	3	4	5
x	-30	-10	5	10	25
z	100	110	130	150	120

There are nine stations on the surface (z=0), at which the first arrival times of P waves (say) are recorded. Their positions are $x_j = -50 + 10j$, j=1,...,9. Travel time data is generated by using the velocity $v^*(x,z) = 5 + .025(x+z)$. We show details of two runs, one starting with $\underset{\sim}{\mu}$ fairly close to the true solution (5, .025, .025), and a second one with a much poorer initial guess. In the column marked max. res. we show the maximum value of the difference $|T_i^c - T_i^o|$, while $(\Sigma\ res_i^2)^{1/2}$ gives the value of the L_2 residual (both measured in seconds):

It.	μ_1	μ_2	μ_3	max.res.	$(\Sigma\ res_i^2)^{1/2}$
0	5.100	.02600	.026	.508	1.68
1	4.998	.02497	.02496	.013	.043
2	5.000	.02500	.02500	3.01×10^{-5}	6.87×10^{-5}
0	4.0	.01	.01	8.92	28.43
1	4.857	.0189	.0178	2.17	6.38
2	5.009	.0243	.0239	.201	.533
3	5.000	.02499	.02499	.001	.003

ii) Simultaneous recovery of velocity and relocation of hypocenters.

With the same model of (i) we show now how we can recover the velocity parameters and the positions of the sources. To save computer time we use only 25 of the 45 available observations. The 10 additional parameters for the source positions are:

$$\mu_{3+2j-1} = x_j \quad , \quad \mu_{3+2j} = z_j \quad ,$$

and the artificial data is the same as for example (i).

It\μ	μ_1	μ_2	μ_3	μ_4	μ_5	μ_6	μ_7
0	5.200	.0200	.03000	-32.00	105.	-12.0	105.0
1	4.918	.0253	.02539	-29.58	98.8	- 9.69	109.6
2	4.997	.0250	.02502	-30.02	100.	-10.0	110.0
3	4.998	.0250	.02503	-29.97	100.	- 9.97	110.0
Exact	5.000	.0250	.02500	-30.00	100.	-10.0	110.0

	μ_8	μ_9	μ_{10}	μ_{11}	μ_{12}	μ_{13}	Max. Res.	Max.Rel Error
0	5.20	125.0	13.0	156.	22.0	125.0	6.77	.300
1	5.03	129.5	8.88	149.	24.4	118.5	.508	.117
2	4.96	130.0	9.90	150.	24.9	120.0	.026	.010
3	5.04	130.0	10.0	150	25.0	120.0	.001	.007
Exact	5.00	130.0	10.0	150	25.0	120.0		

In the last row we list the maximum relative error in the parameters for each iteration. The total CPU time for this run was 2 minutes.

iii) Simultaneous recovery of velocity and interfaces.

We consider the same velocity structure as before, but now we have stations and shooting points located in the surface (z=0) with $x_j = -5+j$, $j=1,\ldots,9$. We also have a reflector made up of three line segments:

$$z - \frac{4}{9}x - \frac{40}{9} = 0 \qquad -10 < x < -1 \,,$$

$$z - 4 = 0 \qquad -1 < x < 1 \,,$$

$$z + \frac{4}{9}x - \frac{40}{9} = 0 \qquad 1 < x < 10 \,.$$

Thus we have to determine two dipping reflectors and a horizontal one on a medium with a laterally inhomogeneous structure.

For economy, we take only 23 observations obtained by shooting from all the positions but receiving simple reflections only at the odd numbered ones. Again the first three parameters correspond to the velocity, while the last six serve to define the reflector segments: $z + \mu_{3+2j-1}x + \mu_{3+2j} = 0 \quad j=1,2,3.$

Param./Iter	0	1	2	3	Exact
μ_1	4.0	4.220	4.870	4.998	5.000
μ_2	0.0	0.0132	0.0208	0.02499	0.025
μ_3	0.0	0.2523	0.0678	0.02602	0.025
μ_4	-0.4	-0.4174	-0.446	-0.4453	-0.444
μ_5	-4.0	-4.072	-4.430	-4.451	-4.444
μ_6	0.1	0.072	-0.00076	-0.000066	0.0
μ_7	-3.8	-3.840	-3.989	-4.001	-4.000
μ_8	0.4	0.4203	0.446	0.4453	0.444
μ_9	-4.0	-4.060	-4.410	-4.451	-4.444
Max.Res.	1.4	0.284	0.0477	0.0013	
Rel. Error	0.2	0.227	0.043	0.001	
CPU time	317"				

Acknowledgement

This work was partly supported by Contract No. 14-08-0001-167 with the U.S. Geological Survey (Earthquake Hazards Reduction Progra

REFERENCES

1. Aki, K. and W.H.K. Lee, "Determination of three-dimensional velocity anomalies under a seismic array using first P arrival times from local earthquakes. 1: A homogeneous initial model", J. Geophys. Res 81 pp. 4381-4399 (1976).

2. Bois, P., M. La Porte, M. Lavergne, et G. Thomas, "Essai de determination automatique des vitesses sismiques p measures entre puits", Geophysical Prosp. 19 pp. 42-83 (1971).

3. Cerveny, V., I.A. Molotkov, and I. Psencik , Ray Method in Seismology. Univ. Karlova, Praha (1977).

4. Gay, D., "Modifying singular values: existence of solutions to systems of nonlinear equations having a possibly singular Jacobian matrix", Math. Comp. 31 pp. 962-973 (1977).

5. Gill, P.E., and W. Murray, "Algorithms for the solution of the nonlinear least squares problem", SIAM J. Numer. Anal. 15 pp. 977-992 (1978).

6. Jacob, K.H., "Three-dimensional seismic ray tracing in a laterally heterogeneous spherical earth", J. Geophys. Res. 75 pp. 6685-6689 (1970).

7. Julian, B.R., "Ray tracing in arbitrarily heterogeneous media", Tech. Note 1970-45, Lincoln Lab. (1970).

8. Julian, B.R. and D. Gubbins, "Three dimensional seismic ray tracin J. Geophys. 43 pp. 95-114 (1977).

9. Jupp, D.L.B., and K. Vozoff, "Stable iterative methods for the inversion of Geophysical data", Geophys. J. R. Astro. Soc. 42 pp. 957-976 (1975).

10. Keller, H.B., "Accurate difference methods for nonlinear two-point boundary value problems", SIAM J. Numer. Anal. 11 pp. 305-320 (1974).

11. Keller, J.B. and H. B. Keller, "Determination of reflected and transmitted fields by geometrical optics", J. Opt. Soc. Amer. 40 pp. 48-52 (1950).

12. Lentini, M. and V. Pereyra, "An adaptive finite difference solver for nonlinear two-point boundary problems with mild boundary layers", SIAM J. Numer. Anal. 14 pp. 91-111 (1977).

13. Pereyra, V., "Iterated deferred corrections for nonlinear operator equations", Numer. Math. 10 pp. 316-323 (1967).

14. Pereyra, V., "PASVA3: An adaptive finite difference FORTRAN program for first order nonlinear, ordinary boundary problems", Proc. Working Conf. Codes for BVP in ODE's. Lecture Notes in Comp.Sci., Springer-Verlag. To appear.

15. Pereyra, V., W.H.K. Lee, and H. B. Keller, "Solving two-point seismic ray-tracing problems in a heterogeneous medium. Part 1. A general numerical method based on adaptive finite-differences". Submitted to Bull. Amer. Seismological Soc.

16. Wesson, R.L., "Travel time inversion for laterally inhomogeneous crustal velocity models", Bull. Seism. Soc. Am. 61 pp. 729-746 (1971).

17. Yang, J.P. and W.H.K. Lee, "Preliminary investigations on computational methods for solving two-point seismic ray-tracing problems in a heterogeneous and isotropic medium", USGS Open File Rep. 76-707 (1976).

18. de Boor, C., A practical guide to splines. Applied Math. Sc. 27, Springer-Verlag, New York (1979).