Stanford Exploration Project, Report 65, January 13, 1998, pages 41–59

Stanford Exploration Project, Report 65, January 13, 1998, pages 41–59

Upwind finite-difference calculation of traveltimes

Jos van Trier and William W. Symes[1]

keywords: *traveltime, modeling, velocity, tomography*

ABSTRACT

Fluid flow is often described by conservation laws that define the conservation of mass, momentum, or energy in a fluid. In fluid mechanics, a standard technique for solving such laws is upwind finite differencing, a numerical method that uses different finite-difference operators depending on the direction of the fluid flow. Upwind finite-difference methods are more stable than centered finite-difference techniques because they mimic the behavior of fluid flow by only using information taken from upstream in the fluid.
Seismic traveltimes can be computed with upwind finite differences by solving a transformed eikonal equation. The transformed equation is a conservation law that describes the changes in the gradient components of the traveltime field among points on the computational grid.
A first-order upwind finite-difference scheme proves accurate enough for seismic applications. The method calculates single-valued traveltime functions (i.e. first arrival times) efficiently on a regular grid, and is useful both in Kirchhoff migration and modeling and in seismic tomography.

INTRODUCTION

Traveltime calculations play an important role in many methods of seismic data processing. For example, the Kirchhoff methods of migrating and modeling seismic data require the calculation of Green's functions that describe the traveltimes between survey points on the surface and depth points in the velocity model. These traveltimes are often calculated by ray tracing (Červený, 1987), and, because depth and surface points are normally distributed on a regular grid, the traveltimes along the rays are then interpolated onto the grid. For complicated velocity models, rays may cross each other or not penetrate shadow zones; the interpolation is thus cumbersome and computationally expensive.

Recently, several methods have been introduced to calculate traveltimes directly on a regular grid. In calculating the depth gradient of traveltimes, Reshef and Kosloff (1986) use a finite-difference approximation to the eikonal equation, which

[1]**email:** not available

they then integrate with a Runge-Kutta method. Vidale (1988) also approximates the eikonal equation by finite differences, but he solves directly for traveltime using a planar or circular wavefront extrapolation.

Although these methods are elegant alternatives to interpolating traveltimes between rays, and far more efficient computationally, they have some drawbacks. Reshef and Kosloff's method is unstable for models with large velocity contrasts; in regions with velocity discontinuities, traveltimes have to be calculated by alternative methods, such as ray tracing. The principal drawback in Vidale's method is that the points on the computational front must be ordered, with calculations starting at the so-called "relative minimum points". This ordering step complicates the algorithm, and, because it is not vectorizable, makes the method computationally expensive.

Here we suggest an alternative scheme that overcomes the problems described above. We observe that the components of the traveltime gradient satisfy so-called hyperbolic conservation laws. Conservation laws are extremely important in fluid mechanics — the Euler equations and the gas dynamics equations are examples — and a great deal of effort has been devoted to their numerical solution. Accordingly we have used an "off-the-shelf," upwind finite-difference scheme as the basis for our method. The resulting scheme is stable, involves none of the sorting implicit in Vidale's approach, and vectorizes fully.

Just like Vidale's method (which is in fact also an upwind finite-difference scheme), our scheme correctly estimates the traveltimes of first arrivals in caustics and of diffractions in shadow zones. A disadvantage is that neither method calculates multivalued traveltime curves. This restriction is intrinsic to the approaches pursued here and by Vidale. Although this limitation can be serious for some applications, many Kirchhoff algorithms use single-valued traveltime functions anyway, and the method is readily applicable to these algorithms. Also, many tomography codes assume that picked times are first arrivals. To the extent that this assumption is correct, finite-difference traveltime calculation might serve as the basis for very efficient tomography codes.

In this paper we first describe the notion of viscosity solutions, which underlies the finite-difference calculations, and explain the need for upwind differencing, which plays a key role in both our scheme and Vidale's. Next we state the eikonal equation and describe the related conservation laws and upwind finite-difference schemes. Then we demonstrate the calculations with some examples, where we compare the results both with analytical solutions and with those obtained by Vidale's method. Finally, we discuss some of the limitations and applications of the method.

VISCOSITY SOLUTIONS AND UPWIND SCHEMES

In this section we introduce the key ideas of *nonclassical solution* and *viscosity solution* of eikonal equations and related problems, which underlie our treatment of traveltime calculations. Our approach is based on this conjecture: *The first-arrival* *time field is a viscosity solution of the eikonal equation.* The conjecture is supported by some theoretical results and by a great deal of numerical ediffvidtimeence. The notion of viscosity solution was developed first for *hyperbolic conservation laws*, particular instances of which are satisfied by the gradient components of the time field. Numerical methods for these conservation laws use *upwind finite differencing*, and may be adapted to produce viscosity solutions of the eikonal equation. Here we introduce these ideas in a non-technical, heuristic way. Formal mathematical treatment can be found in subsequent sections, or in the references.

Nonclassical and viscosity solutions

The class of solutions computed by our scheme — the single-valued first-arrival-time fields — are nonclassical solutions to the eikonal equation. That is, these solutions typically have discontinuous gradients. Thus the equations are not satisfied in the normal ("classical") sense, as one does not know what values to assign to the gradient components at the points of discontinuity. The remedy is to introduce a notion of nonclassical or weak solution: essentially, a function with possibly discontinuous gradients that solves the equations "on average". Because the values at isolated points or curves do not influence averages, it does not matter how one assigns the gradient values at discontinuities.

Unfortunately, the class of weak solutions is too big: the usual (initial) data for these problems does not determine weak solutions uniquely. (The class of classical solutions is too small, since there usually are no classical solutions.) Therefore an additional criterion must be supplied to fix an unique solution. In the fluid-mechanics problems that motivated much of the work on these equations, it was recognized early on that the effect of viscosity had been neglected. It was found that when viscosity is included properly, no actual gradient discontinuities develop, and the solutions are generally classical, i.e. smooth. Because smooth solutions are uniquely determined by the appropriate initial data, the addition of viscosity fixes the uniqueness problem.

A very important property of viscosity solutions, first understood in the context of conservation laws (Lax, 1973), is that these are the unique, weak solutions, *stable* with respect to perturbations in the equation. Because round-off and discretization errors always perturb a numerical solution, viscosity solutions are a natural target for numerical methods.

Conservation laws

The notion of viscosity solutions was developed first for conservation laws: first-order, partial-differential equations linear, in the derivatives. (Note that the eikonal equation is nonlinear in the derivatives.) Conservation laws arise as simple models of inviscid fluid flow, and their weak solutions are made unique by a viscosity principle as well. These viscosity weak solutions are also called "entropy" or "admissible" solutions. An excellent survey of this topic is in Lax (1973). We note that the connections between viscosity, entropy, and the uniqueness of weak solutions are completely understood at present only in special situations, notably scalar conservation laws in two variables.

We base our numerical work on the observation that the gradient components of a solution to the eikonal equation satisfy conservation laws. Moreover, the gradient components of viscosity solutions of the eikonal equation are viscosity solutions of the conservation laws. As noted above, recent theoretical work (Crandall and Lions (1984)) and a lot of numerical ediffvidtimeence support the conjecture that the first-arrival condition actually picks out a viscosity solution of the eikonal equation, from amongst all the branches of traveltime (all of which solve the eikonal equation locally). Thus we can characterize first-arrival-time fields by noting that their gradient components are entropy or viscosity solutions to the corresponding conservation laws.

Upwind finite differences

Conservation laws are commonly solved by upwind finite differencing, a numerical technique that plays an important role in modern computational fluid dynamics; see for example Harten et al. (1987) for a review. Depending on the flow direction, upwind finite-difference schemes use a backward or forward finite-difference approximation to the derivative operator. The sense of the derivative is switched so that only field points from upstream in the flow are used to update any given point.

This direction switching is forced by the appearance of discontinuities (shocks) in the flow variables. For the eikonal equation, the gradient components of traveltime can be discontinuous: rays, whose direction is determined by the gradient components, "break" at interfaces in the model where velocity contrasts occur. Reshef and Kosloff (1986) observe stability problems in a centered finite-difference scheme at these interfaces. Upwind finite-difference methods, on the other hand, are stable because they mimic the underlying physics of the problem in two important aspects. First, they add numerical viscosity to the equations, thus finding a smooth viscosity solution. Second, they copy the behavior of continuum flow by taking their information from upstream.

Careful inspection of Vidale's ordering of gridpoints reveals that he applies the basic "box" (four-point) difference formula only in the upwind sense (see Vidale, 1988, Figure 5), having used an alternative extrapolation at the few points at which the flow (i.e. ray field) changes direction along the computational front. Those points are local traveltime minima. His scheme is thus an upwind finite-difference method, although not presented as such. It is an *implicit* method, as it connects more than one value on the grid level being updated. The ordering of points is required to achieve a closed-form solution of the difference formulas, as opposed to an iterative approximate solution (as is often chosen with implicit schemes). Our formulas are *explicit*, in contrast, and do not require any special ordering of points.

EIKONAL EQUATION

The eikonal equation in two dimensions is as follows (Aki and Richards, 1980, ch.4):

$$\left(\tau_x\right)^2 \; + \; \left(\tau_z\right)^2 \; = \; s^2(x,z), \tag{1}$$

where $s(x,z)$ is the 2-dimensional slowness model and $\tau = \tau(x,z)$ is the traveltime field. Subscripts x and z denote partial derivatives with respect to x and z, respectively.

Hyperbolic conservation law

We observe that the gradient components of the eikonal equation satisfy a hyperbolic conservation law. First, use $u = \tau_x$ to rewrite equation (1) as

$$\tau_z \; = \; \sqrt{s^2 - u^2}. \tag{2}$$

Second, take the derivative of this equation with respect to x:

$$u_z \; = \; F_x(u), \tag{3}$$

where the function $F(u)$ is defined as

$$F(u) \; = \; \sqrt{s^2 - u^2}. \tag{4}$$

$F(u)$ is called the conserved flux; if $F(u) = 0$, the rays do not "flow" downward anymore, but travel horizontally. Thus by choosing a positive sign in front of the square root in equation (2), and by using $u = \tau_x$ instead of $w = \tau_z$ as the substitution variable, we limit ourselves to time-fields with downward-traveling rays.

By following analogous reasoning for the z-derivative w of the time field, and by making the appropriate choices of square roots, we can build equations for rays traveling in other directions. These other equations will have to be solved, for instance, when the field of a point source is being computed, because the rays then move in all directions.

An alternative approach, which we will follow here, is to write the eikonal equation in polar coordinates (r,θ),

$$\left(\tau_r\right)^2 \; + \; \left(\frac{\tau_\theta}{r}\right)^2 \; = \; s^2(r,\theta), \tag{5}$$

and solve it along expanding circular fronts. The conserved-flux function in polar coordinates becomes

$$F(u) \; = \; \sqrt{s^2 - \frac{u^2}{r^2}}, \tag{6}$$

with $u = \tau_\theta$ satisfying

$$u_r \; = \; F_\theta(u). \tag{7}$$

The above equation (and equation (3)) now has the form of a hyperbolic conservation law, or flux-conserving equation. The meaning of this equation is illustrated with an example in the next section.

Flow balancing

Figure 1 shows the flow variable u and the flux function $F(u)$ for a constant-velocity medium. Both are two-dimensional functions, and are drawn as vectors on the 2-D grid, with u, the horizontal time gradient, pointing in the horizontal direction, and $F(u) = w$, the vertical time gradient, pointing in the vertical direction. The length of the vectors corresponds to the magnitude of the flow. Equation (3) describes the conservation of flow: for each grid cell, the equation balances the change in horizontal flow with the change in vertical flow. Thus, the total flow through each grid cell, the vector sum of u and $F(u)$, is constant (Figure 1c). The magnitude of the total flow is s, which denotes the preserved "substance" for this problem. (In many fluid-mechanics problems, the flow variable is velocity, and the preserved scalar quantity is energy.)

However, for inhomogeneous media s is not constant, and no classical solution exists. In these media, the flux function F is discontinuous at interfaces where velocity contrasts occur; flow is "destroyed" or "created" in grid cells along these interfaces. Possible nonclassical solutions to the problem would therefore also be discontinuous: gradient components of the traveltime field would locally obey Snell's law, describing rays that break when passing a velocity contrast.

As discussed before, nonclassical solutions are generally not accessible by finite-difference methods. Instead, viscosity solutions have to be computed, where a numerical "viscous" quantity is added to s, which is then preserved by the equation. (In fluid-mechanics problems with energy as the conserved quantity, viscosity allows dissipation of energy.) In computational fluid dynamics, many methods have been developed for computing these viscosity solutions (see Roache, 1976). The next section discusses one particular method.

FINITE-DIFFERENCE SCHEME

Although sophisticated, higher-order finite-difference schemes exist for the solution of hyperbolic conservation laws (Centrella and Wilson, 1984; Hawley et al., 1984; Harten et al., 1987), a first-order upwind finite-difference scheme appears generally accurate enough for geophysical applications. We implemented a first-order method described by Engquist and Osher (1980).

The first-order forward and backward finite differences in space are, respectively,

$$\begin{aligned} \Delta_+ u &= u_{j+1} - u_j; \\ \Delta_- u &= u_j - u_{j-1}, \end{aligned} \tag{8}$$

where u_j is the discrete representation of $u(\theta)$ on the grid $\theta_j = j\Delta\theta$.

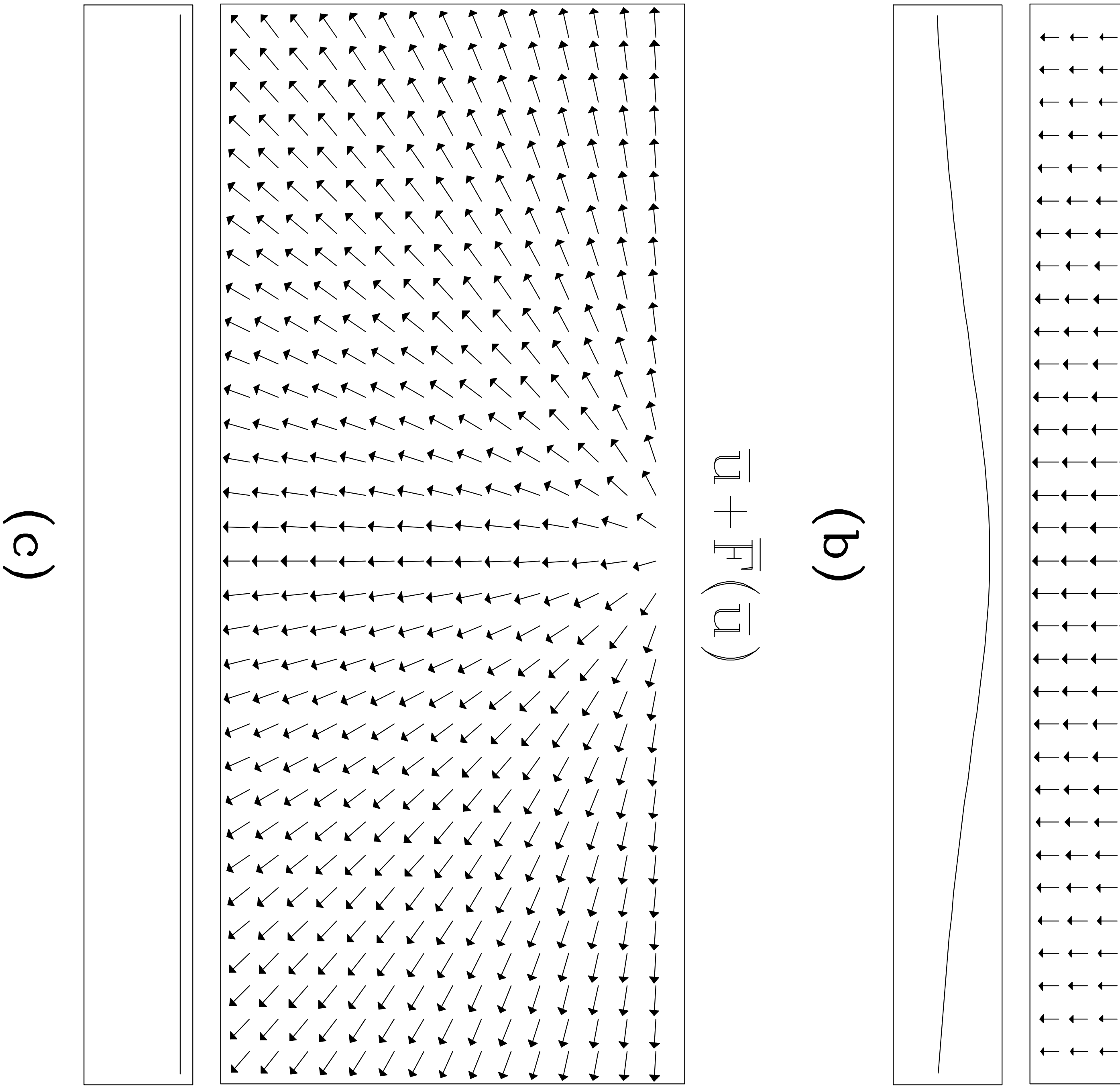

Figure 1: Flow in a constant velocity medium with a source at the surface in the middle of the model. Flow is shown as 2-D vector fields. The figure shows: (a) the flow variable u, (b) the flux function $F(u)$, and (c) the vector sum of u and $F(u)$. The graphs at the bottom of each plot display the length of the vectors at a constant-depth level in the middle of the model. jos-flow [NR]

The basic upwind scheme for equation (7) is

$$\frac{u_j^{n+1} - u_j^n}{\Delta r} = \frac{1}{\Delta\theta} \begin{cases} \Delta_+ \; F(u_j^n) & \text{if } F'(u_j^n) \text{ and } F'(u_{j+1}^n) \leq 0; \\ \Delta_- \; F(u_j^n) & \text{if } F'(u_j^n) \text{ and } F'(u_{j-1}^n) > 0, \end{cases} \tag{9}$$

with n the discrete-radius index of the grid, $r^n = n\Delta r$. The direction of the finite-difference operator depends on $F'(u)$: when the flux increases from left to right, flow is to the left; when it decreases, flow is to the right (see Figure 1). Note that an equivalent scheme can be set up in Cartesian coordinates, where r is replaced by z, and θ by x.

This specification is incomplete, of course. The various upwind schemes differ in the way in which the intermediate cases are handled; i.e. when the sign of $F'(u)$ changes amongst the three points of the stencil. The scheme of Engquist and Osher handles the intermediate cases via the formula:

$$u_j^{n+1} = u_j^n \; + \; \frac{\Delta r}{\Delta\theta} \left(\Delta_+ F_-(u_j^n) \; + \; \Delta_- F_+(u_j^n) \right). \tag{10}$$

Here

$$\begin{aligned} F_+(u) \quad &= \quad F(\max(u, \overline{u})); \\ F_-(u) \quad &= \quad F(\min(u, \overline{u})), \end{aligned} \tag{11}$$

with $\overline{u}$ the stagnation point: $F'(\overline{u}) = 0$. $\overline{u} = 0$ for the function F under consideration here. Engquist and Osher's scheme reduces to the standard first-order scheme (9) when u has a consistent sign, and is stable near gradient discontinuities.

Note that the specification given above is still not complete: the slowness is position-dependent, therefore so is the flux. We have suppressed this dependence in the notation, but it must be respected. Some obvious ways of evaluating the flux lead to inconsistent schemes. In general it seems sufficient to verify that the elementary upwind differences above are recovered when the sign of u is consistent. For example, taking the slowness locally constant at its value at u_j^n seems to work.

Because an explicit scheme is used, each step (fixed by the input slowness grid) may require several partial steps to satisfy the Courant-Friedrichs-Lewy stability condition. The appropriate partial steps are determined adaptively, and the full step is always taken if stably possible. Note the contrast with Vidale's code, which uses an implicit upwind scheme.

Initial and boundary conditions

We have presented the numerical scheme in polar coordinates for the sole purpose of solving the point-source problem. In polar coordinates the initial condition at the source is easily specified as $u(0, \theta) = 0$. However, the same problem may be solved in Cartesian coordinates if calculations are done on a sequence of expanding rectangular fronts. Although slightly more complicated, this approach saves the cost of mapping the slowness and traveltime fields to and from polar coordinates. The rectangular computational fronts are also used by Vidale (1988).

When one assumes outgoing rays at the boundaries, one-sided finite-differences can be used at the left and right sides of the model. These boundary conditions are of the same order as the scheme inside the model, and generally do not cause any problems. After u has been computed, traveltimes are calculated by integrating τ_r over r using a simple trapezoidal rule, where τ_r is found from equation (5).

EXAMPLES

In the first example we calculate traveltimes for a medium with a velocity gradient linear with respect to depth (the constant-velocity case is trivial in polar coordinates). The velocity at the top of the model is 2 km/s, the one at the bottom 2.5 km/s. The grid is evenly sampled in depth and laterally, with a sample interval of 10 m. Figure 2 shows the traveltime function for a source on the surface at the middle of the model, and Figure 3 displays the difference between that function and the analytical solution. Finite-difference traveltimes are calculated in polar coordinates, and errors accumulate as the radius increases. A simple bilinear interpolation is used in the mapping from and to Cartesian coordinates. Although interpolation errors in that mapping are not large, they cause the distinct pattern visible in Figure 3. The maximum error is about .3 ms (see Figure 5), which is one order of magnitude smaller than the standard time-sampling interval of 4 ms.

Figure 4 shows the difference between Vidale's scheme and the analytical solution. We have implemented only the plane-wave extrapolation method, and errors can probably be reduced if a combination of plane and circular wave extrapolation is used. The errors are largest away from the vertical, diagonal, and horizontal directions, where the plane-wave approximation breaks down. The errors at the bottom of the model are of the same magnitude as the errors in the method described here (again see Figure 5).

The next example illustrates the calculations for a more complicated model. The model is shown in Figure 6; it consists of 3 layers and a wedge intrusion. The velocity in the top layer is 2 km/s, the middle layer has a velocity of 1.75 km/s, and the bottom layer's velocity is 2.5 km/s. The velocity in the wedge that intrudes the middle layer from the right is 2.75 km/s. Figure 7 shows the result of tracing rays through a smoothed version of the model. The smoothing causes the rays to bend or turn in regions with a large velocity gradient.

As is obvious from Figure 7, interpolating traveltimes from the rays onto the grid is not easy for this model; some parts of the model are not illuminated by rays, and in some other parts rays cross. However, the finite-difference calculation correctly fills in the problem areas as can be seen in Figure 8: the contour lines in the plot reveal the correct curvature of the wave fronts in the high- and low-velocity regions.

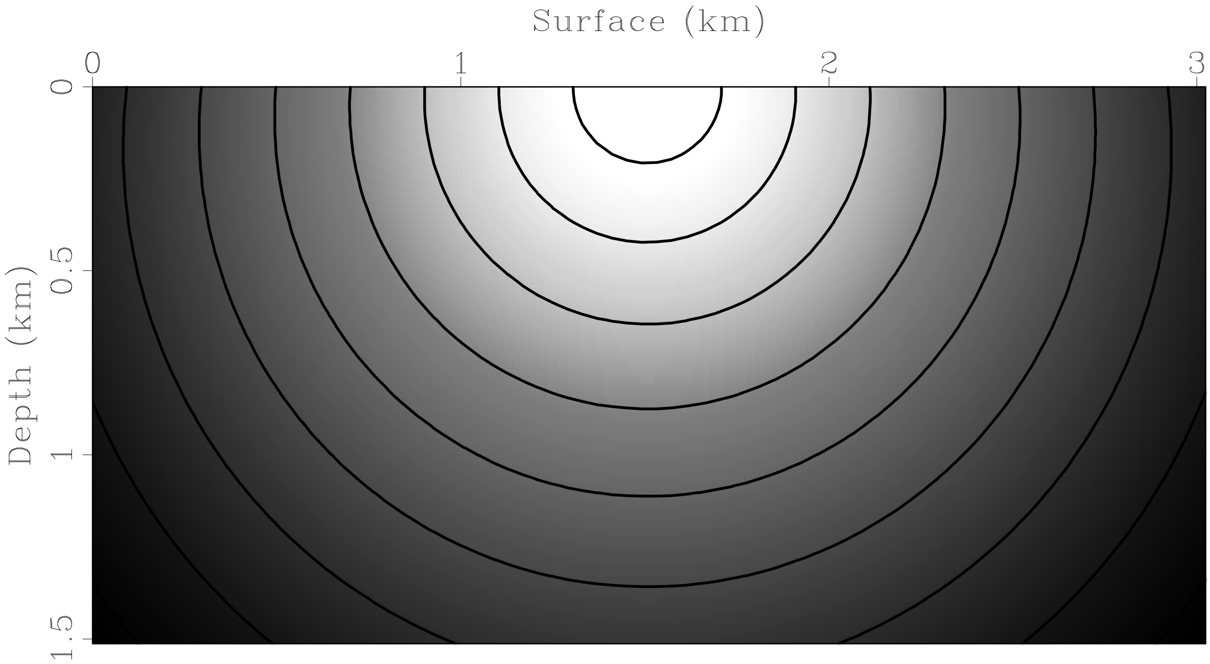

Figure 2: Finite-difference traveltime field for a model with a velocity function that is linearly increasing as a function of depth. The source is located at the surface in the middle of the model. The figure displays both an intensity and contour plot of the traveltime field. Low intensities denote small traveltimes; contour lines are drawn at .1 s intervals. jos-fdtimelin [NR]

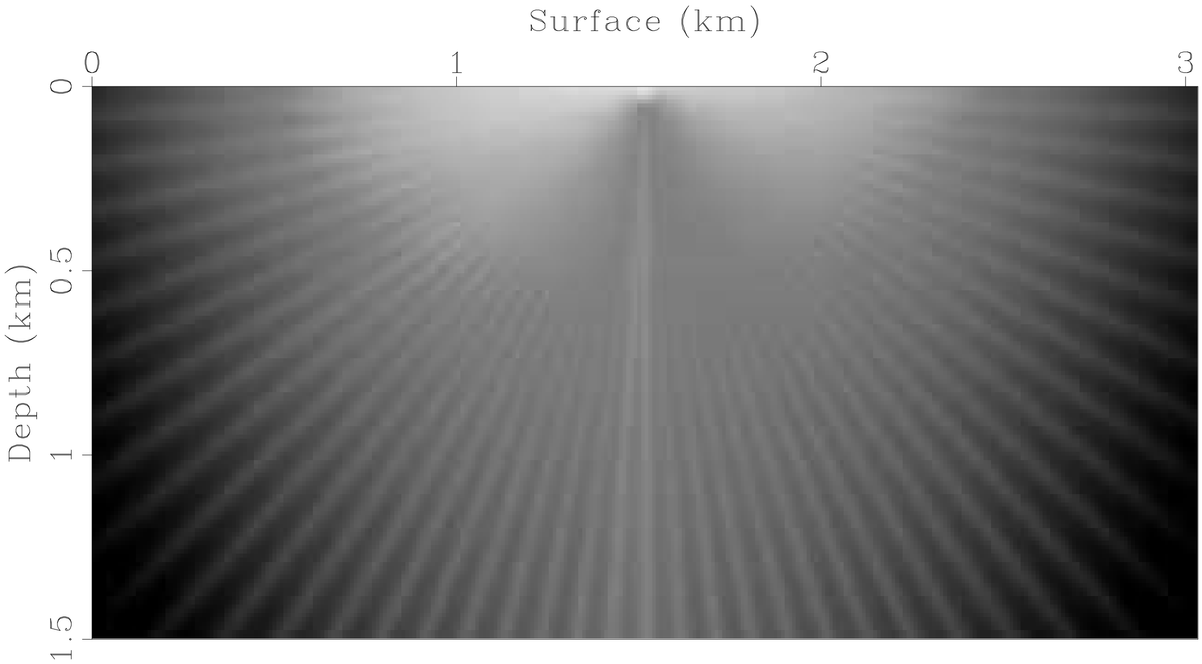

Figure 3: Difference between the traveltime map of Figure 2 and the analytical solution. Higher intensities in the plot represent larger errors. jos-difftime [NR]

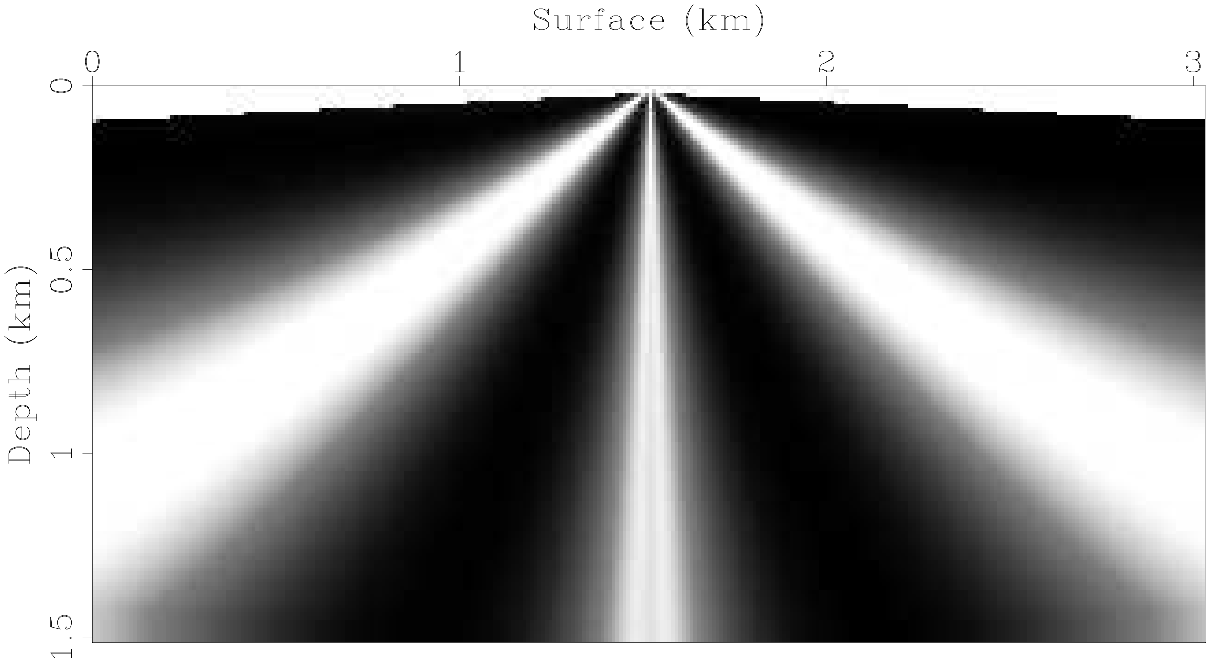

Figure 4: Difference between a traveltime function calculated with Vidale's plane-wave extrapolation method and the analytical solution. jos-diffvidtime [NR]

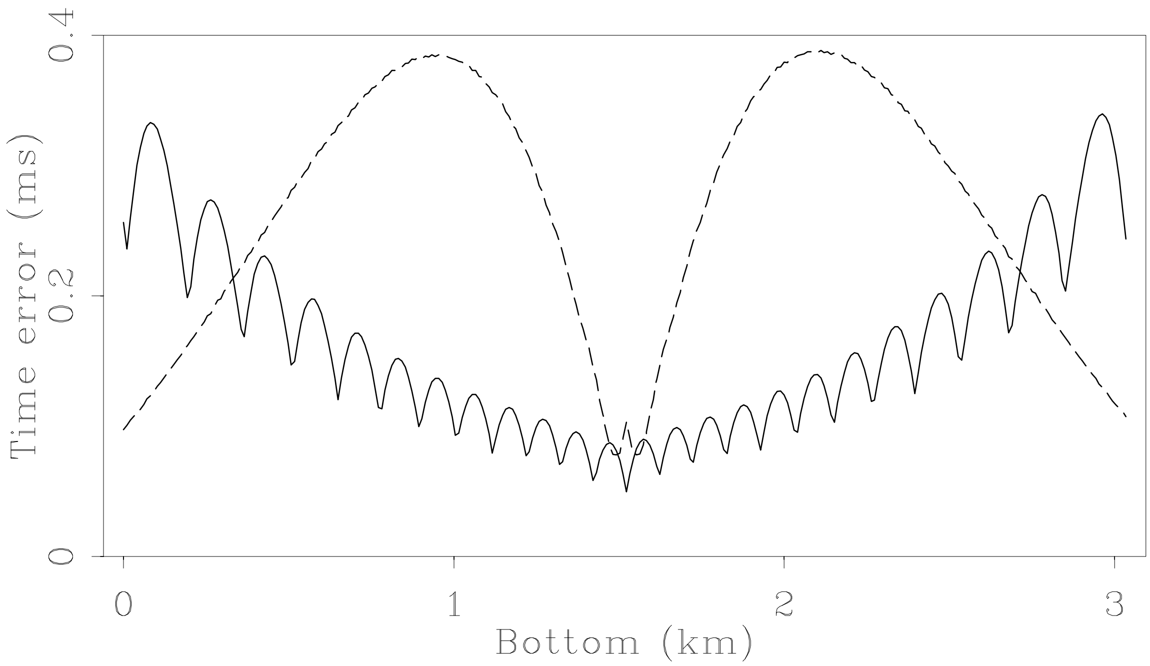

Figure 5: Errors in the finite-difference traveltime calculations at the bottom of model. The solid line represents the error curve for the method described here, the dashed line denotes the errors in Vidale's scheme. jos-bottom [NR]

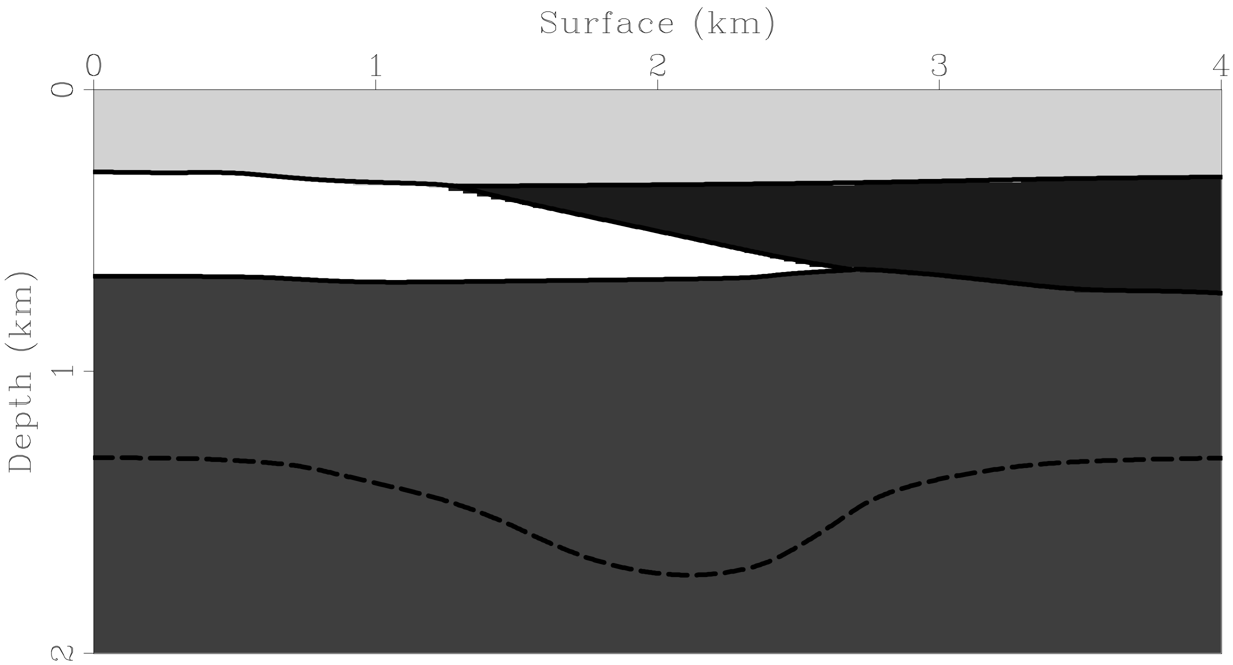

Figure 6: Wedge model. The grid spacing is 10×10 m. Low intensities denote low velocities. The interfaces between the different layers are represented by solid lines in the figure. The dashed line denotes an imaginary reflector in the bottom layer. Figure 10 shows reflection events that correspond to these reflectors. jos-strmodbnd [NR]

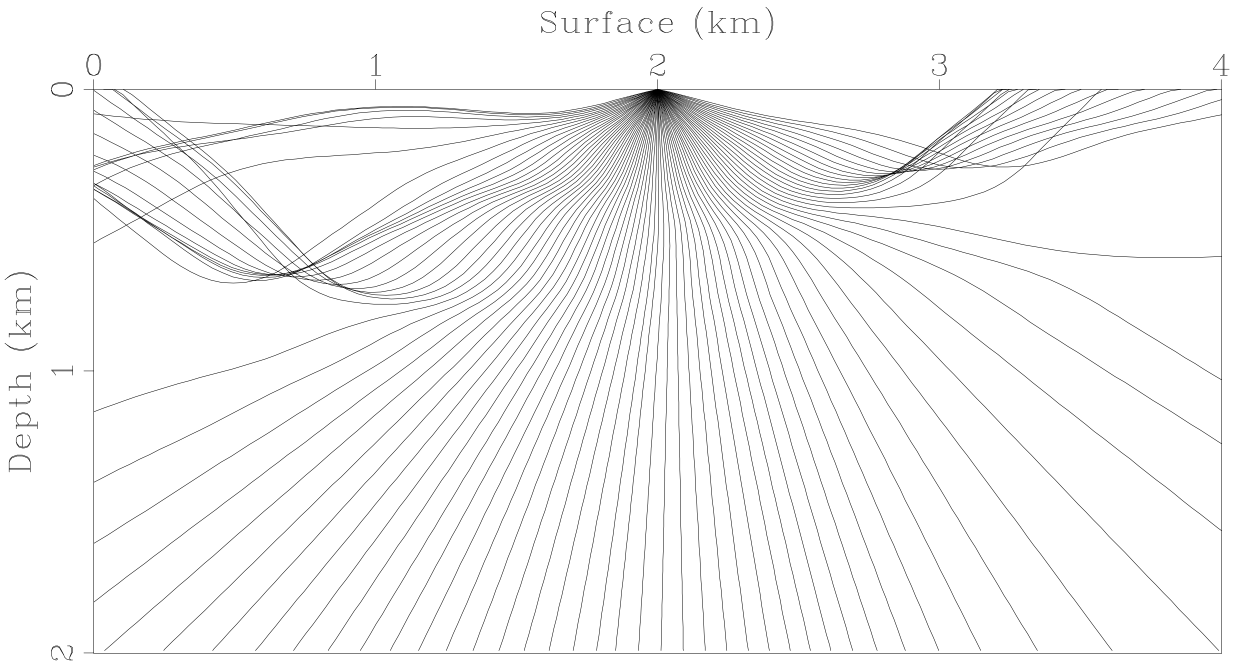

Figure 7: Rays traced through a smoothed version of the model in Figure 6. jos-raystr [NR]

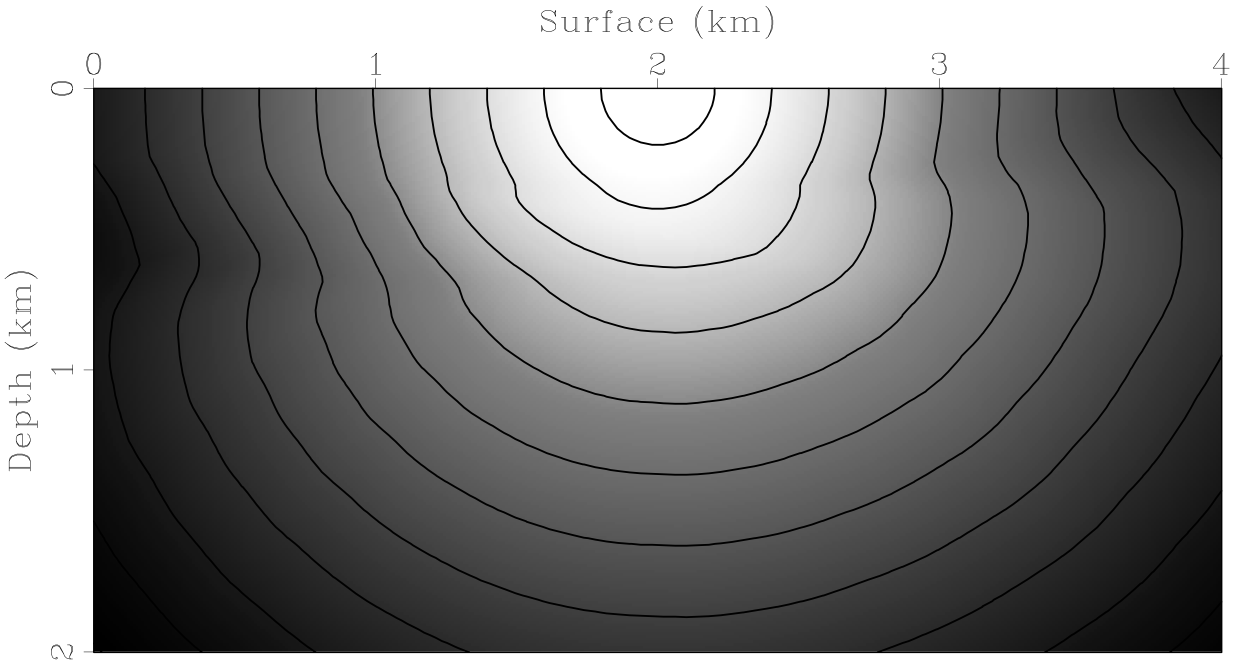

Figure 8: Finite-difference traveltimes calculated for the model of Figure 6. Overlain on the figure are contour lines of the traveltime field. The contour interval is .1 s. jos-fdtimestr [NR]

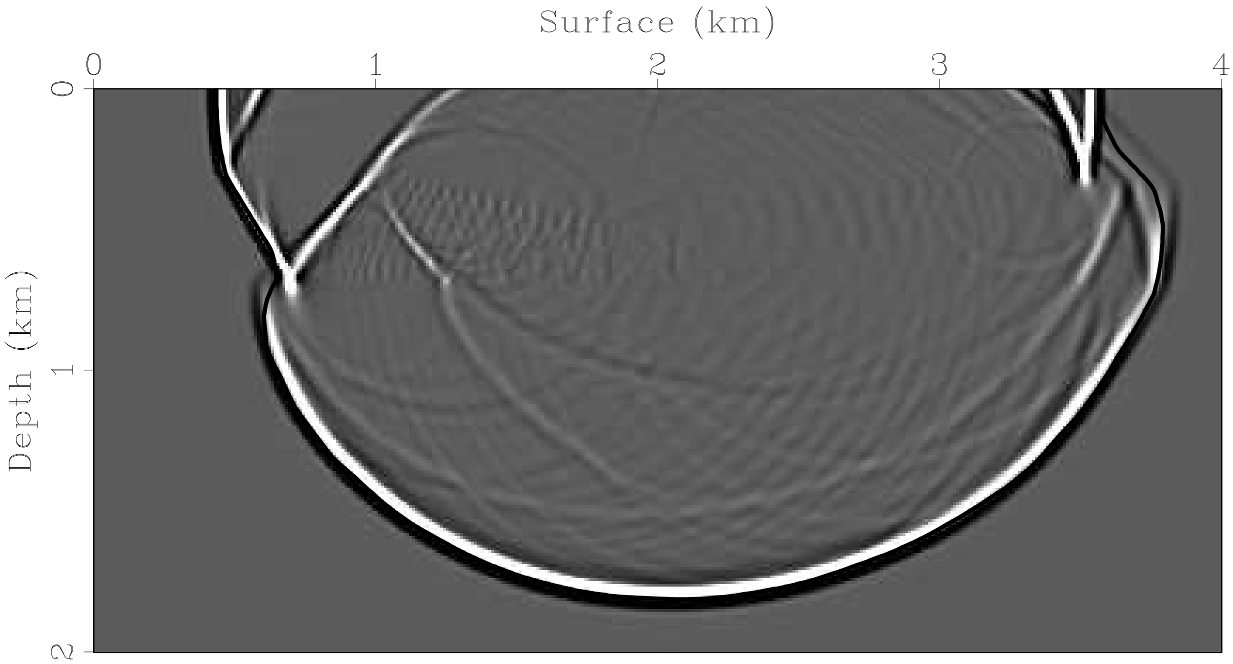

Figure 9: Comparison of the wave field computed by wave-equation modeling with the traveltime field calculated by upwind finite-differences. The intensity plot shows a snapshot of the wave field at .7s; the overlain dashed curve is the .7s-contour of the traveltime function (see Figure 8). jos-wave [NR]

This result is verified in Figure 9, which shows the result of finite-difference wave-equation modeling. The figure displays a snapshot of the wave field at .7 s. Also shown in the figure is the .7s-contour line of the traveltime function (Figure 8). Barring some discrepancies due to the limited bandwidth and dispersion of the source wavelet, the contour exactly follows the first-arrival wave front. In particular note the match between wave field and traveltime function in the upper-right corner of the model, where the refracted wave travels in front of the direct wave.

LIMITATIONS

The main limitation of the method was noted before: it computes only the first-arrival field, as that is the unique viscosity (thus stable) solution to the eikonal equation. Another limitation follows from the "thermodynamical" nature of viscosity: it is one-directional in time, as it tends to suppress high-frequency components. Even in the viscosity limit defining the solution class of interest to us, the computed solution depends on the direction of "time", which is actually a spatial direction varying from place to place.

It is *necessary* to *assume* that the least-time field is the admissible viscosity weak solution of the eikonal equation with respect to the "time" variable implicitly defined by the schedule of expanding circular or rectangular fronts — and this assumption might or might not be correct! Apparently the algorithm always computes some piece of a branch of the traveltime, but it may well stop before filling the computational domain. This feature is shared with Vidale's code. Both codes rely on an a-priori prescription of a version of "time", through the family of computational fronts along which the solution is marched. If the time field to be computed does not have an outward-pointing gradient at each front — i.e. if the time gradient becomes parallel to the computational front (a turned ray) — then the calculation stops. It is easy to construct examples that cause any a-priori prescribed family of computational fronts to fail to have this essential outflow property. Therefore neither this code nor Vidale's can be relied upon completely — both succeed only when the user has general knowledge of the shape of the time field. This limitation is *essential* — it comes from the necessary role of "thermodynamical" considerations in picking out the "correct" weak (nonclassical) solution.

APPLICATIONS

The main application of the method is in areas where many traveltime calculations need to be done, or where traveltimes are needed for large, complicated models. In particular the 3-D version of the algorithm promises interesting applications. We leave the discussion of the 3-D algorithm and its applications for a follow-up paper; here we just discuss some applications in 2-D.

Tomography

In seismic tomography, traveltimes recorded in a seismic experiment are compared with traveltimes computed for an assumed velocity model, and their differences are used to predict any anomalies in the model (Dines and Lytle, 1979; Bishop et al., 1985; Nolet, 1987). Tomographic methods generally use an iterative approach, in which many traveltime calculations are needed for the algorithm to converge to the correct solution. Because the calculations are fast and efficient, the finite-difference traveltime method is well-suited for tomography.

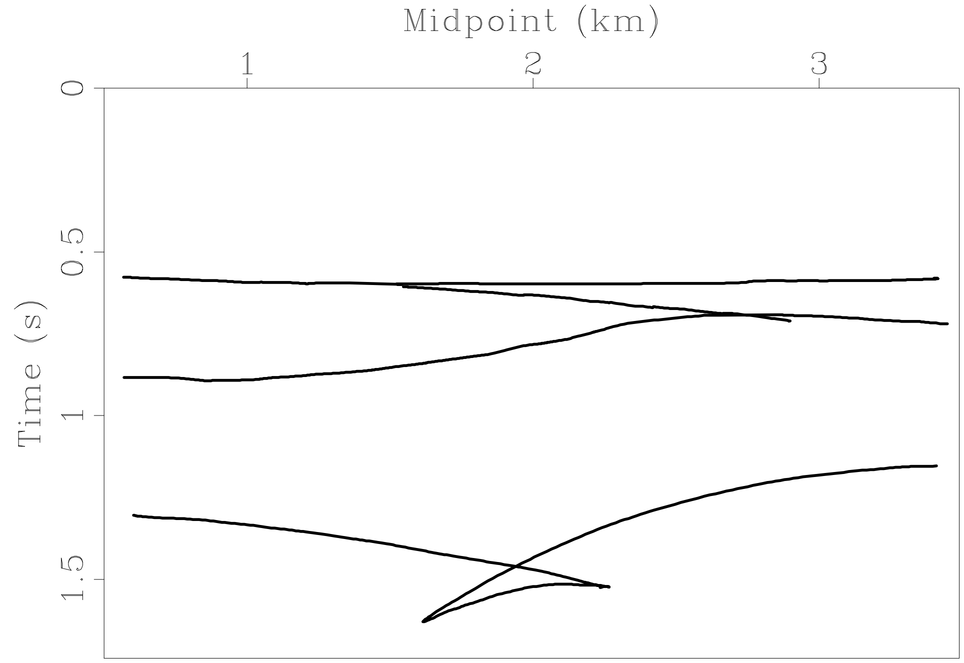

Figure 10: Reflection events in a constant-offset section computed for reflectors in the model of Figure 6. The offset is 1 km. jos-strmodevt [NR]

The application of this method to tomography is not limited to the transmission experiment; reflection traveltimes can be computed by combining information from "source" and "geophone" traveltime maps. For example, Figure 10 shows the reflection events in a constant-offset section, computed for interfaces in the model of Figure 6. After the calculation of the source and receiver traveltime maps, reflection times are found by the summation of source and receiver times at locations along the interface where Snell's law is satisfied. Incidence and reflection angles at the interface are computed from the traveltime gradients, which are readily available in the finite-difference calculations. When calculations must be done for a large number of regularly spaced sources and geophones, this approach is much more efficient than two-point ray tracing.

Note that no rays are needed in the computation of reflection times. Similarly, in seismic tomography, optimization schemes can be designed that do not require any rays in the calculation of the Jacobian of traveltimes with respect to slowness. We are currently investigating the application of these techniques to migration-velocity analysis (Van Trier, 1989) and reflection tomography (Santosa and Symes, 1989). Vidale and Ammon (1989) also apply the finite-difference traveltime calculations to tomography without using rays. They use a random-search optimization method in the inversion of traveltime perturbations.

Kirchhoff migration and modeling

The Green's functions used in Kirchhoff methods describe traveltimes between depth points in the model and survey points on the surface. One can use the described method to precompute these traveltime maps for source points that are densely spaced on the surface, and then use them when needed in the Kirchhoff summation; see for example Gray (1986), and Reshef and Kosloff (1986). If a map for a certain surface point is not available, a simple linear interpolation of traveltimes in two neighboring maps is accurate enough for most purposes.

The finite-difference method does not compute multi-valued traveltime functions, but geometry-related triplications in reflection events are still handled correctly (see Figure 10). The only problem arises when, for example, refracted waves travel faster than direct waves; this phenomenon normally occurs only in the shallow, wide-angle part of the data.

The amplitude terms in the Green's functions can be approximately estimated from the traveltime function: the geometrical spreading and obliquity terms are approximate functions of traveltime or its gradient (see also Vidale, 1989).

CONCLUSIONS

The finite-difference method greatly simplifies traveltime calculations on a regular grid. The algorithm has the advantage over Vidale's method that all points can be treated equally; i.e., minimum traveltime points do not have to be found at each extrapolation step, and the difference scheme vectorizes fully. The method allows efficient calculation of Greens's functions in Kirchhoff methods, and fast traveltime computations in iterative tomography schemes.

REFERENCES

Aki, K., and Richards, P., 1980, Quantitative seismology: Freeman.

Bishop, T.N., Bube, K.P., Cutler, R.T., Langan, R.T., Love, P.L., Resnick, J.R., Shuey, R.T., Spindler, D.A., and Wyld, H.W., 1985, Tomographic determination of velocity and depth in laterally varying media: Geophysics, **50**, 903–923.

Centrella, J., and Wilson, J., 1984, Planar numerical cosmology. II. The difference equations and numerical tests: Astrophysical Journal Supplement, **54**, 229-249.

Červený, V., 1987, Ray tracing algorithms in three-dimensional laterally varying layered structures: Seismic Tomography, edited by G. Nolet, Riedel Publishing Company, 99–134.

Crandall, G., and Lions, P., 1984, Two approximations of solutions of Hamilton-Jacobi equations: Math. Comp., **43**, 1-19

Dines, K.A., and Lytle, R.J., 1979, Computerized geophysical tomography: Proc. IEEE, **67**, 1065–1073.

Engquist, B., and Osher, S., 1980, Stable and entropy satisfying approximations for transonic flow calculations: Math. Comp., **34**, no. 149, 45-75.

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

Harten, A., Engquist, B., Osher, S., and Chakravarthy, J., 1987, Uniformly high order accurate essentially non-oscillatory schemes, III: J. Comput. Phys., **71**, 231-303.

Hawley, J., Smarr, L., and Wilson, J., 1984, A numerical study of nonspherical black hole accretion. II. finite differencing and code calibration: Astrophysical Journal Supplement, **55**, 211-246.

Lax, P., 1973, Hyperbolic systems of conservation laws and mathematical theory of shock waves: Reg. Conf. Series in Applied Math.

Nolet, G., 1987, ed., Seismic tomography: D. Reidel Publ. Co.

Reshef, M., and Kosloff, D., 1986, Migration of common shot gathers: Geophysics, **51**, 324-331.

Roache, P., 1976, Computational fluid dynamics: Hermosa, Albuquerque, New Mexico.

Santosa, F., and Symes, W.W., 1989, An analysis of least-squares velocity inversion: SEG Geophysical Monograph Series, **4**.

Van Trier, 1989, Migration-velocity analysis using geological constraints: presented at the 59th annual SEG meeting, New Orleans.

Vidale, J., 1988, Finite-difference calculation of traveltimes: Bull., Seis. Soc. Am., **78**, no. 6, 2062-2076.

Vidale, J., 1989, Rapid calculation of seismic amplitudes: submitted to Geophysics.

Vidale, J., and Ammon, C., 1989, Efficient seismic traveltime and amplitude calculations and application to velocity inversion and migration: presented at the 1989 Fall Meeting of the AGU, San Francisco.