3–D Kirchhoff Migration Using Finite Difference Traveltimes and Amplitudes

Seongjai Kim[*], *William W. Symes*[†], *and* *Maissa A. El–Mageed*[‡]

ABSTRACT

Imaging techniques based on high frequency asymptotic representation of the 3–D acoustic Green's function require efficient solution methods for the computation of traveltimes and amplitudes. Second–order finite difference schemes solve the eikonal and transport equations on regular grids with usable accuracy. Since the transport equation includes the gradient and the Laplacian of the computed traveltime field, special superconvergent difference formulas must be employed to guarantee accurate amplitudes. Upwind finite differences are requisite to sharply resolve discontinuities in the derivatives, while centered differences improve accuracy and enjoy extra–order accuracy for the averaged gradient of the traveltime field. A two–level, second–order, upwind, *essentially non–oscillatory* (ENO) scheme satisfies these requirements. The computed amplitude is at least first–order accurate, while the traveltime is second–order accurate. We embed the eikonal/transport solver in a 3–D Kirchhoff migration algorithm, resulting in accurate and efficient depth migrations.

Key words. Kirchhoff migration, upwind finite differences, traveltime and amplitude, eikonal/transport solver

INTRODUCTION

Kirchhoff migrations incorporate high frequency asymptotic representation of the seismic wave field and its integration along diffraction surfaces. Such migration methods play an important role in the seismic data processing and are interesting subjects

[*]Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005–1892 skim@caam.rice.edu

[†]Department of Computational and Applied Mathematics, Rice University, Houston, TX 77005–1892 symes@caam.rice.edu

[‡]Department of Environmental Studies, Institute of Graduate Studies and Research, Alexandria University, Alexandria, Egypt

of research. Computation of traveltimes and the corresponding amplitudes is essential and is the most costly component of the methods. The use of ray tracing followed by interpolation of traveltimes is a popular and robust method for computing diffraction trajectories. An alternative to the method is to compute traveltimes by solving directly the eikonal equation on a regular grid by finite difference schemes; see (Cerveny, 1985; Reshef and Kosloff, 1986; Vidale, 1988; van Trier and Symes, 1991; Gray and May, 1994; Symes et al., 1994). For a comprehensive treatment of ray theory, see e.g. (Friedlander, 1958) and (Cerveny et al., 1977).

A drawback for migrations using the eikonal solvers is that it is not easy to control the propagation angle as much as ray tracing methods. Nontheless the eikonal solvers have their advantages: they are much cheaper (in computation cost) than ray tracers and rarely produce shadow zones. For the regions of high velocity contrasts, ray tracing methods can produce quite large shadow zones where the computed traveltime field should be interpolated. Such ray–tracing/interpolation processes are quite cumbersome and computationally very expensive, especially in three dimensions.

The computational efficiency/accuracy and data storage issues are very interesting and challenging tasks for 3–D seismic data processing. In this article, we propose a shock–capturing second–order finite difference scheme for solving the eikonal and transport equations in 3–D, focusing on computational efficiency and the improvement of solution accuracy. It is one member of the so called *Essentially Non–Oscillatory* (ENO) schemes proposed by (Osher and Sethian, 1988) and (Osher and Shu, 1991) for Hamilton–Jacobi initial–boundary value problems, and adopted by (Symes et al., 1994) and (El-Mageed, 1996) for the eikonal equation in 2–D and 3–D, respectively.

The transport equation involves the gradient and the Laplacian of the computed traveltime field and that the involvement of traveltime derivatives easily deteriorates the accuracy of the computed amplitude field. We can see from the theory of numerical analysis that the computed amplitude may have accuracy of order zero, i.e. $\mathcal{O}(1)$. We may have to use higher–order schemes for traveltime computation in order to achieve a positive order accuracy in the computed amplitude field. Is the use of higher–order schemes only the remedy? To answer the question, we recall the following theory well–known in the literature of finite element methods (Křížek and Neittaanmäki, 1984): the computed solution of finite element methods and its averaged gradient can enjoy the same order of accuracy at interior grid points of uniform meshes. It is easy to verify that the averaged gradient of the tri–linear finite element solution (integrated by the Trapezoidal rule) is exactly the same as the centered finite differences for the first-derivatives of the solution. So the centered difference scheme leads to a second–order approximation for the gradient of the computed traveltime field at interior grid points of uniform meshes. Although there is no such (theoretical) superconvergence similarity for the Laplacian of the solution, we may *intuitively* expect a first–order accuracy for the traveltime Laplacian, and therefore for the computed amplitude field, provided that second–order centered finite difference schemes are carefully applied to traveltime derivatives on uniform meshes and that the data near the computational boundary are treated efficiently.

Another difficulty inherited in computation of the traveltimes and amplitudes is possibility of shock–development which prevents the eikonal/transport solver from utilizing uniformly centered differences. We have observed from various numerical tests that the second–order ENO scheme of Osher and Sethian (1988) forces centered differences at *most* grid points where the solution is smooth. Therefore the ENO differences for the traveltime first-derivatives are mostly second–order accurate for uniform meshes.

In the next section, we present a two–level second–order ENO scheme for the traveltime field, after discussing geometrical concerns. We will replace $\partial^2\tau/\partial z^2$ in the transport equation by an identity involving only first z–derivatives, in order to enforce the following conditions: to make possible a two–level scheme for depth matching compatible with the traveltime scheme; to make good use of the accurate approximation of $\partial\tau/\partial z$ obtainable from the traveltime scheme; to release the uniformity requirement of the grid size in z–direction, the marching direction. Each of these conditions plays an important role for computational efficiency and numerical accuracy. The last condition, in particular, allows the schemes to choose dynamic step sizes in z–direction without losing the extra–order accuracy for the amplitudes. In the same section, the stability condition and an efficient boundary data treatment are addressed in detail. The following section gives numerical examples verifying our expectation for numerical accuracy: *the amplitude field converges with accuracy order greater than one.* The algorithm is applied to the Kirchhoff depth migration of (Symes et al., 1994); the 3–D deliberation is presented in Appendix A. The last section includes the conclusions and future work.

TRAVELTIME AND AMPLITUDE COMPUTATION

In this section, we present a shock–capturing second–order finite difference scheme for solving the eikonal and transport equations in three–dimensional space. The scheme is a member of *Essentially Non–Oscillatory* (ENO) schemes introduced by (Osher and Sethian, 1988) for the solution of Hamilton–Jacobi initial–boundary value problems. It was utilized by (Symes et al., 1994) for the solution of 2–D eikonal and transport equations and extended to three dimensions in (El-Mageed, 1996). El-Mageed observed that the amplitude errors for her algorithm were in $\mathcal{O}(1)$, order zero, as the grid size decreases. To increase computational accuracy and efficiency, the scheme has been modified as follows. For the case of traveltime shocks, we employ

Godunov–type flux (Osher and Shu, 1991); for the x– and y–derivatives of the amplitude, ENO differences are adopted rather than one–sided second–order differences; for the treatment of boundary data, we add an outer border to the computational domain and expand the boundary data linearly. As the consequences of such computational and physical contemplation, we have observed a halved computation time and the accuracy order for the amplitude field at least one.

Geometrical considerations

Recall the technical (not intrinsic) limitations on finite difference eikonal solvers: rays bringing significant reflected energy propagate near vertically, and reflections take place only at points on simple, single–arrival wavefronts. The first of these restrictions implies that we exclude the possibility of imaging via turning waves: incident rays are wholly going down and reflected rays are entirely upcoming. This means that for the traveltime field $\tau = \tau(\mathbf{x}, \mathbf{x}_s)$,

$$\frac{\partial \tau}{\partial z} > 0,$$

at least for $\{\mathbf{x} = (x, y, z) : z > z_s\}$, below the source where reflection occurs. A nondimensionalized version of the criterion is $|\sin\theta| < 1$, where θ is the angle between the wavefront normal and the vertical.

Evidently rays are traveling in all directions, in particular, near point sources. Rays incident at high angles tend to turn before reaching deep reflectors in the typical velocity structure increasing with depth; these rays are irrelevant to the simulation of reflections. In order to eliminate this class of rays from the calculation, we need to modify the point source initial condition for the traveltime and amplitude fields and the eikonal equation itself. Let us impose the *maximum angle condition* to compute accurately only traveltimes along rays whose velocity vectors make angle θ with the vertical not larger than $\theta_{\max}$, i.e.,

$$|\sin\theta| \leq \sin\theta_{\max} < 1.$$

To enforce this condition, we modify the eikonal equation, written as an evolution equation in depth:

$$\frac{\partial \tau}{\partial z} = \sqrt{s^2 - \left(\frac{\partial \tau}{\partial x}\right)^2 - \left(\frac{\partial \tau}{\partial y}\right)^2}, \tag{1}$$

where $s\,(= 1/v)$ is the slowness. Note that

$$\left(\frac{\partial \tau}{\partial z}\right)^2 = s^2 - \left(\frac{\partial \tau}{\partial x}\right)^2 - \left(\frac{\partial \tau}{\partial y}\right)^2 = s^2 - s^2\sin^2\theta = s^2\cos^2\theta;$$

the maximum angle condition turns out to be

$$\frac{\partial \tau}{\partial z} \geq s\cos\theta_{\max}, \quad 0 < \theta_{\max} < \frac{\pi}{2}.$$

Following (Gray and May, 1994), we modify the eikonal equation as follows:

$$\frac{\partial \tau}{\partial z} = \sqrt{\mathrm{bmax}\left(s^2 - \left(\frac{\partial \tau}{\partial x}\right)^2 - \left(\frac{\partial \tau}{\partial y}\right)^2,\ s^2 \cos^2 \theta_{\max} \right)}\ , \tag{2}$$

where

$$\mathrm{bmax}(x,a) = \begin{cases} a, & \text{if } x \le 0, \\ a + \dfrac{x^2}{4a}, & \text{if } 0 < x < 2a, \\ x, & \text{if } x \ge 2a. \end{cases}$$

In effect we have rectified the wavefront; the part with near vertical normals ($\theta < \theta_{\max}$) is unaltered, but the steeper part is replaced by a plane wave. For the sake of merciful efficiency, bmax may be written as

$$\mathrm{bmax}\,(x,a) = \min\left(\left(a + \frac{x \cdot \max(x,0)}{4a} \right), \max(x, 2a) \right).$$

Remark. In the maximum angle condition, we might replace the function "bmax" in (2) by "max" for the computation of the traveltime field. However, since the transport equation involves the gradient and the Laplacian of the traveltime, the discontinuities of the derivatives caused by the max function will produce spurious oscillations in the amplitude field near the edge of the aperture, and the oscillations will contaminate the solution afterwards.

Remark. We allow the rays to propagate naturally, without correction, only if their angle is somewhat smaller than the maximum angle $\theta_{\max}$. That is, $\mathrm{bmax}(x,a) = x$ only if $x \ge 2a$. Let us, for example, assume $s \equiv 1$ and set $\theta_{\max} = 60°$. Then, $a = \cos^2 \theta_{\max} = 0.25$ and $\cos^{-1}\left(\sqrt{2a}\right) = 45°$. So, only the rays with propagation angle $\le 45°$ can propagate without being bothered by the rectification.

The initial condition for τ satisfies

$$\lim \left(\frac{\tau(\mathbf{x}, \mathbf{x}_s)}{|\mathbf{x} - \mathbf{x}_s|} - \frac{1}{v(\mathbf{x})} \right), \quad \text{as } |\mathbf{x} - \mathbf{x}_s| \to 0;$$

assuming the velocity is constant near the point source $\mathbf{x}_s$, we can initialize the traveltime as $\tau(\mathbf{x}, \mathbf{x}_s) = |\mathbf{x} - \mathbf{x}_s| / v(\mathbf{x}_s)$. We shall also rectify this initial condition; see below.

The transport equation (A-4.ii) for the amplitudes may be rewritten as

$$\left(\frac{\partial}{\partial z} + \left(\frac{\partial \tau}{\partial z}\right)^{-1} \frac{\partial \tau}{\partial x}\frac{\partial}{\partial x} + \left(\frac{\partial \tau}{\partial z}\right)^{-1} \frac{\partial \tau}{\partial y}\frac{\partial}{\partial y} \right) \log a = -\frac{1}{2} \left(\frac{\partial \tau}{\partial z}\right)^{-1} \nabla^2 \tau. \tag{3}$$

This formulation is advantageous in maintaining uniform accuracy of the numerical solution in the presence of discretization and round–off error; the amplitude field changes several orders of magnitude over the typical computational domain, while its logarithm changes much less.

As a by–product of the solution of the modified eikonal equation (2), we also modify the transport equation (3) *via* a tapered cutoff function g:

$$\begin{aligned}\left(\frac{\partial}{\partial z}+\left(\frac{\partial\tau}{\partial z}\right)^{-1}\frac{\partial\tau}{\partial x}\frac{\partial}{\partial x}+\left(\frac{\partial\tau}{\partial z}\right)^{-1}\frac{\partial\tau}{\partial y}\frac{\partial}{\partial y}\right)\log a \\ = -\frac{1}{2}\left(g\left(\frac{\partial\tau}{\partial z}\right)^{-1}\nabla^2\tau+(1-g)\max_{\{x,y\}}\left[\left(\frac{\partial\tau}{\partial z}\right)^{-1}\nabla^2\tau\right]\right).\end{aligned} \tag{4}$$

Here

$$g=\mathrm{bcut}\left(s^2-\left(\frac{\partial\tau}{\partial x}\right)^2-\left(\frac{\partial\tau}{\partial y}\right)^2,\, s^2\cos^2\theta_{\max}\right),$$

where $\mathrm{bcut}(x,a)$ is a piecewise cubic continuously differentiable function that is $\equiv 0$ for $x\le a$ and $\equiv 1$ for $x\ge 2a$, defined by

$$\mathrm{bcut}(x,a)=-2\left(\frac{r}{a}\right)^3+3\left(\frac{r}{a}\right)^2,\quad r=\min\Big(a,\max(x-a,0)\Big).$$

The initial condition for a is

$$a(\mathbf{x},\mathbf{x}_s)=\frac{\rho}{4\pi\,|\mathbf{x}-\mathbf{x}_s|},$$

for a constant velocity near the source point. We will “rectify” this initial condition also by setting $a=0$ (in practice, an arbitrarily small positive number) on the steeply dipping parts of the initial front. Together with the damping implemented by the taper cutoff, this makes the amplitudes along “incorrect” rays ($\theta>\theta_{\max}$) decay rapidly to zero.

Discretization of traveltime fields

We begin with the initialization of the traveltime field. Pick a convenient datum depth $z_d>z_s$ which lies in a homogeneous layer containing the source. For example, it can be the water bottom in an idealized model. Let

$$\mathbf{x}_d=(x_s,y_s,z_d),\quad e=|\mathbf{x}_s-\mathbf{x}_d|=|z_s-z_d|,\quad \text{and}\ \ R=e\cdot\tan\theta_{\max};$$

see Figure 1. We initialize the traveltime on $\{z=z_d\}$ as follows. At the grid points on the disk $\Big\{\mathbf{x}:\mathbf{x}\in\{z=z_d\},\ |\mathbf{x}-\mathbf{x}_d|\le R\Big\}$, simply set

$$\tau(\mathbf{x},\mathbf{x}_s)=\frac{|\mathbf{x}-\mathbf{x}_s|}{v(\mathbf{x}_s)}.$$

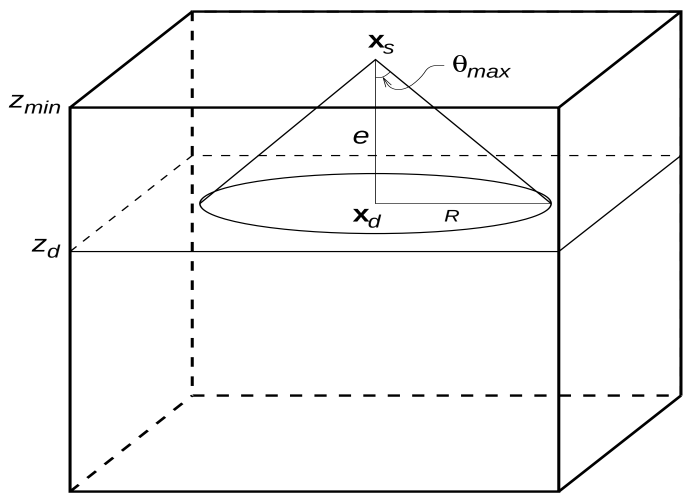

FIG. 1. Initialization of the traveltime field on $\{z = z_d\}$.

For the points outside of the disk, i.e. $r = |\mathbf{x} - \mathbf{x}_d| > R$, we use first–order Taylor expansion around r:

$$\tau(r) \approx \tau(R) + \frac{\partial \tau(R)}{\partial r}(r - R),$$

which incorporates the maximum angle condition. Since

$$\tau(R) = \frac{\sqrt{e^2 + R^2}}{v} = \frac{\sqrt{e^2(1 + \tan^2\theta_{\max})}}{v} = \frac{e}{v\cos\theta_{\max}},$$

$$\frac{\partial \tau(R)}{\partial r} = \frac{R}{v\sqrt{e^2 + R^2}} = \frac{e\tan\theta_{\max}}{ve\sec\theta_{\max}} = \frac{\sin\theta_{\max}}{v},$$

we can summarize the initialization of the traveltime field on $\{z = z_d\}$ as follows:

$$\tau(\mathbf{x}, \mathbf{x}_s)|_{\{z=z_d\}} = \begin{cases} \dfrac{|\mathbf{x} - \mathbf{x}_s|}{v(\mathbf{x}_s)}, & \text{if } |\mathbf{x} - \mathbf{x}_d| \le R, \\ \dfrac{1}{v(\mathbf{x}_s)}\left(\dfrac{|z_s - z_d|}{\cos\theta_{\max}} + (|\mathbf{x} - \mathbf{x}_d| - R)\cdot\sin\theta_{\max}\right), & \text{else,} \end{cases}$$

where $R = |z_s - z_d| \cdot \tan\theta_{\max}$.

The next objective is to introduce a second–order ENO scheme for τ on $\{z > z_d\}$, for the point source at $\mathbf{x}_s$. For a function f of the space variable $\mathbf{x}$ in the computational domain, write

$$f_{i,j}^k = f(\mathbf{x}_{i,j}^k),$$

$$\mathbf{x}_{i,j}^k = (x_i, y_j, z_k) = (x_{\min} + (i-1)\Delta x, y_{\min} + (j-1)\Delta y, z_{\min} + (k-1)\Delta z).$$

Let $\tau_{i,j}^k = \tau(\mathbf{x}_{i,j}^k, \mathbf{x}_s)$ and define the forward $(+)$ and backward $(-)$ finite difference operators:

$$D_x^{\pm}\tau_{i,j}^k = \pm\frac{\tau_{i\pm1,j}^k - \tau_{i,j}^k}{\Delta x}.$$

Then, from Taylor expansion, we have

$$\frac{\partial \tau}{\partial x} = D_x^{\pm}\tau \mp \frac{1}{2}\Delta x \cdot \frac{\partial^2 \tau}{\partial x^2} + \mathcal{O}(\Delta x^2).$$

For the shock–capturing second–order finite difference scheme, we choose the ENO schemes of (Osher and Sethian, 1988) as particularly appropriate. First we approximate the second derivative in the right hand side of the above equation by

$$\frac{\partial^2 \tau}{\partial x^2} \approx \mathrm{m}\Big(D_x^- D_x^+ \tau,\ D_x^{\pm} D_x^{\pm} \tau\Big),$$

where

$$\begin{aligned} \mathrm{m}(x,y) &= \min\Big(\max(x,0), \max(y,0)\Big) + \max\Big(\min(x,0), \min(y,0)\Big) \\ &= \begin{cases} 0, & \text{if } xy \le 0, \\ x, & \text{if } |x| \le |y| \text{ and } xy > 0, \\ y, & \text{if } |x| > |y| \text{ and } xy > 0. \end{cases} \end{aligned}$$

Then, the second–order ENO finite differences for $\partial\tau/\partial x$ reads

$$\frac{\partial \tau}{\partial x} \approx D_x^{\pm,2}\tau = D_x^{\pm}\tau \mp \frac{1}{2}\Delta x \cdot \mathrm{m}\Big(D_x^- D_x^+ \tau,\ D_x^{\pm} D_x^{\pm} \tau\Big).$$

Note that

$$D_x^{+,2}\tau_{i,j}^k = \begin{cases} D_x^+ \tau_{i,j}^k, \\ D_x^+ \tau_{i,j}^k - \dfrac{1}{2}\Delta x \cdot D_x^- D_x^+ \tau_{i,j}^k \;=\; \dfrac{\tau_{i+1,j}^k - \tau_{i-1,j}^k}{2\Delta x}, \quad \text{or} \\ D_x^+ \tau_{i,j}^k - \dfrac{1}{2}\Delta x \cdot D_x^+ D_x^+ \tau_{i,j}^k \;=\; \dfrac{1}{\Delta x}\Big(-\dfrac{3}{2}\tau_{i,j}^k + 2\tau_{i+1,j}^k - \dfrac{1}{2}\tau_{i+2,j}^k\Big). \end{cases}$$

The last two cases of the above are the centered and one–sided second–order finite differences, respectively. Also the first seems to be a second–order approximation for a smooth solution, since it appears at an inflection point, i.e. the derivative $\partial^2\tau/\partial x^2$ is zero at the grid point. The same can be addressed for $D_x^{-,2}\tau_{i,j}^k$.

Now, define the second–order *upwind* ENO approximation for $\partial\tau/\partial x$ incorporating the first–arrivals:

$$\left(\frac{\partial \tau}{\partial x}\right)_{i,j}^k \approx \widehat{D}_x \tau_{i,j}^k = \mathrm{mod\_max}\Big(\max\Big(D_x^{-,2}\tau_{i,j}^k, 0\Big), \min\Big(D_x^{+,2}\tau_{i,j}^k, 0\Big)\Big), \qquad (5)$$

where mod_max returns the larger value in modulus. Introduce the analogues for $\partial\tau/\partial y$: $D_y^{\pm}\tau_{i,j}^k$, $D_y^{\pm,2}\tau_{i,j}^k$, and $\widehat{D}_y\tau_{i,j}^k$. Then, the second–order ENO Runge–Kutta algorithm can be formulated as follows:

$$\begin{aligned} \tau_{i,j}^{k+\frac{1}{2}} &= \tau_{i,j}^k + \Delta z_{\mathrm{CFL}} \cdot H[\tau]_{i,j}^k, \\ \tau_{i,j}^{k+1} &= \frac{1}{2}\Big(\tau_{i,j}^k + \tau_{i,j}^{k+\frac{1}{2}} + \Delta z_{\mathrm{CFL}} \cdot H[\tau]_{i,j}^{k+\frac{1}{2}}\Big), \end{aligned} \qquad (6)$$

where

$$H[\tau]_{i,j}^{k} = \sqrt{\mathrm{bmax}\Big(\big(s_{i,j}^{k}\big)^2 - \big(\widehat{D}_x \tau_{i,j}^{k}\big)^2 - \big(\widehat{D}_y \tau_{i,j}^{k}\big)^2,\ \big(s_{i,j}^{k}\big)^2 \cos^2\theta_{\max}\Big)}$$

and Δz_{CFL} is to be determined to satisfy the stability condition.

Remark. The difference in (5) turns out to become a second–order ENO modification of Godunov–type flux; for a description of the Godunov scheme, see e.g. (Osher and Shu, 1991). Symes et al. (1994) and El–Mageed (1996) adopted the approximation

$$\left(\frac{\partial \tau}{\partial x}\right)_{i,j}^{k} \approx \sqrt{\Big(\max\big(D_x^{-,2}\tau_{i,j}^{k}, 0\big)\Big)^2 + \Big(\min\big(D_x^{+,2}\tau_{i,j}^{k}, 0\big)\Big)^2}$$

which was formerly suggested in (Osher and Sethian, 1988) for solving fluid flow problems.

Discretization of amplitude fields

It should be noticed that the amplitude a (and its logarithm) is accumulative and that its upwind direction should be the same as that of the traveltime field. In this section, we suggest a second–order ENO scheme for the amplitude computation, embodying the above physical observation. Letting

$$u = \log a,$$

we rewrite the transport equation (4) as

$$\begin{aligned} \frac{\partial u}{\partial z} &= -\left(\frac{\partial \tau}{\partial z}\right)^{-1}\left(\frac{\partial \tau}{\partial x}\frac{\partial u}{\partial x} + \frac{\partial \tau}{\partial y}\frac{\partial u}{\partial y}\right) \\ &\quad -\frac{1}{2}\left(g\left(\frac{\partial \tau}{\partial z}\right)^{-1}\nabla^2\tau + (1-g)\max_{\{x,y\}}\left[\left(\frac{\partial \tau}{\partial z}\right)^{-1}\nabla^2\tau\right]\right). \end{aligned} \tag{7}$$

To make possible a two–level scheme for depth matching compatible with the traveltime scheme, it is preferable to write the traveltime Laplacian (more precisely, $\partial^2\tau/\partial z^2$) in a form involving only first z–derivatives. Differentiation of the eikonal equation with respect to x, y, and z, followed by some algebra, yields

$$\begin{aligned} \frac{\partial^2 \tau}{\partial z^2} &= \left(\frac{\partial \tau}{\partial z}\right)^{-2}\left[\frac{1}{2}\left(\frac{\partial \tau}{\partial z}\frac{\partial s^2}{\partial z} - \frac{\partial \tau}{\partial x}\frac{\partial s^2}{\partial x} - \frac{\partial \tau}{\partial y}\frac{\partial s^2}{\partial y}\right)\right. \\ &\quad \left. + \left(\frac{\partial \tau}{\partial x}\right)^2\frac{\partial^2 \tau}{\partial x^2} + \left(\frac{\partial \tau}{\partial y}\right)^2\frac{\partial^2 \tau}{\partial y^2} + 2\frac{\partial \tau}{\partial x}\frac{\partial \tau}{\partial y}\frac{\partial^2 \tau}{\partial x \partial y}\right]. \end{aligned} \tag{8}$$

Now, introduce the approximations for $x-$ and $y-$derivatives of τ:

$$\begin{aligned}
\left(\frac{\partial\tau}{\partial x}\right)_{i,j}^{k} &\approx D_x[\tau]_{i,j}^k = \max\left(D_x^{-,2}\tau_{i,j}^k, 0\right) + \min\left(D_x^{+,2}\tau_{i,j}^k, 0\right),\\
\left(\frac{\partial\tau}{\partial y}\right)_{i,j}^{k} &\approx D_y[\tau]_{i,j}^k = \max\left(D_y^{-,2}\tau_{i,j}^k, 0\right) + \min\left(D_y^{+,2}\tau_{i,j}^k, 0\right),\\
\left(\frac{\partial^2\tau}{\partial x^2}\right)_{i,j}^{k} &\approx D_x^- D_x^+ \tau_{i,j}^k, \quad \left(\frac{\partial^2\tau}{\partial y^2}\right)_{i,j}^{k} \approx D_y^- D_y^+ \tau_{i,j}^k,\\
\left(\frac{\partial^2\tau}{\partial x\partial y}\right)_{i,j}^{k} &\approx \frac{\left(D_x^+ + D_x^-\right)\left(\tau_{i,j+1}^k - \tau_{i,j-1}^k\right)}{4\Delta y}.
\end{aligned}$$

The derivatives of the squared slowness are discretized by the centered finite differences as

$$\left(\frac{\partial(s^2)}{\partial e}\right)_{i,j}^{k} \approx \frac{\left(D_e^+ + D_e^-\right)(s^2)_{i,j}^k}{2}, \quad e = x,\ y,\ \text{and } z.$$

Substitute the above finite difference approximations and

$$\left(\frac{\partial\tau}{\partial z}\right)_{i,j}^{k} \approx H[\tau]_{i,j}^k \tag{9}$$

into (7) (and (8)) to approximate the Laplacian of the traveltime:

$$(\nabla^2\tau)_{i,j}^k \approx L[\tau]_{i,j}^k.$$

Define the upwind ENO difference for $\partial u/\partial x$ which corresponds to the upwind direction of $\partial\tau/\partial x$ as follows:

$$D_x^{\mathrm{up}} u_{i,j}^k = \begin{cases} D_x^{-,2} u_{i,j}^k, & \text{if } D_x[\tau]_{i,j}^k \geq 0,\\ D_x^{+,2} u_{i,j}^k, & \text{else.}\end{cases}$$

Then, we can formulate the second–order ENO Runge–Kutta scheme for the transport equation as

$$\begin{aligned}
u_{i,j}^{k+\frac{1}{2}} &= u_{i,j}^k + \Delta z_{\mathrm{CFL}} \cdot \Psi[\tau, u]_{i,j}^k\\
u_{i,j}^{k+1} &= \frac{1}{2}\left(u_{i,j}^k + u_{i,j}^{k+\frac{1}{2}} + \Delta z_{\mathrm{CFL}} \cdot \Psi[\tau, u]_{i,j}^{k+\frac{1}{2}}\right),
\end{aligned} \tag{10}$$

where

$$\begin{aligned}
\Psi[\tau, u]_{i,j}^k &= -\frac{1}{H[\tau]_{i,j}^k}\left(D_x[\tau]_{i,j}^k \cdot D_x^{\mathrm{up}} u_{i,j}^k + D_y[\tau]_{i,j}^k \cdot D_y^{\mathrm{up}} u_{i,j}^k\right)\\
&\quad -\frac{1}{2}\left(g\frac{L[\tau]_{i,j}^k}{H[\tau]_{i,j}^k} + (1-g)\max_{i,j}\left[\frac{L[\tau]_{i,j}^k}{H[\tau]_{i,j}^k}\right]\right),\\
g &= \mathrm{bcut}\left((H[\tau]_{i,j}^k)^2,\ (s_{i,j}^k)^2\cos^2\theta_{\max}\right).
\end{aligned}$$

Remark. The above algorithm is second–order; however, we cannot get the solution having the same accuracy, since the right–hand side includes the gradient and the Laplacian of the traveltime field computed by a second–order scheme. It is known that the computed solution of finite element methods and its averaged gradient can enjoy the same order of accuracy at interior grid points of uniform meshes (Křížek and Neittaanmäki, 1984). It is easy to see that the averaged gradient of the tri–linear finite element solution (integrated by the Trapezoidal rule) is exactly the same as the centered finite differences for the first-derivatives of the solution. So the finite differences for the first–derivatives of τ in the above algorithm are second–order accurate at *most* interior grid points where the traveltime field is smooth. (It has been observed that the ENO differences are centered differences at most grid points where the solution is smooth.) Although there is no similar superconvergence result for the Laplacian of the solution, we discretize $\nabla^2\tau$ by the centered finite differences on a uniform mesh, hoping that the algorithm relishes an extra–order accuracy. By virtue of (8) and (9), the uniformity of the grid size in the z–direction can be released.

Boundary treatment

Most finite difference eikonal solvers seem to be designed without serious attention to boundary conditions. Since the rays radiate from the source to the outer direction, it may not be so important for upwind schemes of the traveltime field itself. However, if the computed traveltime is being utilized in amplitude calculation and if we want to enjoy a higher accuracy in the amplitude field, then we need to treat the boundary data more carefully for both traveltimes and amplitudes.

We extend the computational domain so that the solutions are *linearly* extended over the modeling boundaries. For instance, for the boundary grid point $\mathbf{x}_{1,j}^k$ and its adjacent grid point $\mathbf{x}_{2,j}^k$,

$$\begin{aligned} D_x^{\pm}\tau_{1,j}^k &= \frac{\tau_{2,j}^k - \tau_{1,j}^k}{\Delta x}, \\ D_x^- D_x^- \tau_{1,j}^k &= D_x^- D_x^- \tau_{2,j}^k = D_x^- D_x^+ \tau_{1,j}^k = 0; \end{aligned}$$

the others can be properly treated by the original definition. With the boundary treatment, no noticeable near–boundary wiggle has been observed in numerical experiments; see Figures 5 and 6.

Stability condition

For given consistent finite difference schemes solving evolutionary partial differential equations, the Courant–Friedrichs–Levy (CFL) stability condition can be viewed as the requirement that the numerical domain of dependence contain the physical

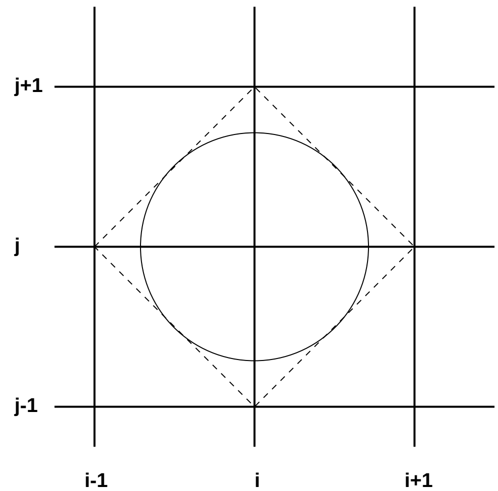

FIG. 2. The numerical domain of dependence depicted in the xy–plain $\{z = z_k\}$. The thick straight lines indicate the grid lines and the dashed box is the numerical domain of dependence for the grid point (x_i, y_j, z_k), that is, $\{(x, y, z_k) : |x - x_i|/\Delta x + |y - y_j|/\Delta y \leq 1\}$. The circle is inscribed in the box.

domain of dependence. This physical interpretation of the CFL condition is advantageous in simplicity, i.e. it is easy to check; the von Neumann condition can be too complicated to check for most numerical schemes solving general partial differential equations.

One should avoid choosing very small step size in the z–direction. Why? First of all, a small step size increases the computational cost. Secondly (with more importance), it may cause a severe numerical diffusion/dispersion (Strikwerda, 1989, Ch.5), since it diminishes the physical domain of dependence and therefore the numerical domain of dependence is relatively enlarged.

Obviously, the numerical domain of dependence for algorithms (6) and (10), evolving from the k–th level to the $(k+1)$–th level at the grid point (x_i, y_j, z_k), is

$$\left\{(x, y, z_k) : \frac{|x - x_i|}{\Delta x} + \frac{|y - y_j|}{\Delta y} \leq 1\right\}$$

and the radius of the inscribed circle is $\Delta x \Delta y / \sqrt{\Delta x^2 + \Delta y^2}$; see Figure 2. The physical domain of dependence for algorithm (10) is

$$\Delta z_{\mathrm{CFL}} \cdot \frac{\text{the speed of } u \text{ on the } xy\text{–plain}}{\text{the speed of } u \text{ in the } z\text{–direction}}.$$

So, the CFL condition for (10) reads

$$\Delta z_{\mathrm{CFL}} \cdot \max_{i,j} \left[\frac{\sqrt{(D_x[\tau]^k_{i,j})^2 + (D_y[\tau]^k_{i,j})^2}}{H[\tau]^k_{i,j}} \right] \leq \frac{\Delta x \Delta y}{\sqrt{\Delta x^2 + \Delta y^2}}. \tag{11}$$

That is, $\Delta z_{\rm CFL}$ can be chosen to satisfy the above inequality. From the maximum angle condition, we have

$$\frac{\sqrt{(D_x[\tau])^2+(D_y[\tau])^2}}{H[\tau]} \approx \frac{\sqrt{s^2-(H[\tau])^2}}{H[\tau]} \le \frac{\sqrt{s^2-s^2\cos^2\theta_{\rm max}}}{s\cos\theta_{\rm max}} = \tan\theta_{\rm max}.$$

Also note that the maximum angle condition forces the upper bound of the physical domain of dependence for algorithm (6) not larger than $\Delta z_{\rm CFL}\tan\theta_{\rm max}$. So the CFL condition (11) can be utilized for (6) as well as (10). Indeed, El–Mageed (1996) showed that (11) was the stability condition for the linearized eikonal equation.

The practical efficiency of the ENO scheme seems to strongly depend on the ability of the numerical code to control the step size $\Delta z_{\rm CFL}$ dynamically. One has to choose $\Delta z_{\rm CFL}$ large enough to reduce the computational cost and undesirable numerical diffusion and, on the other hand, small enough to satisfy the CFL condition which possibly varies on each step. The step size $\Delta z_{\rm CFL}$ must be determined based on the rays of a largest angle not exceeding $\theta_{\rm max}$. Therefore, at the areas where raypaths have small angles, the physical domain of dependence is smaller than the numerical domain of dependence. This may cause a numerical diffusion/dispersion, in particular, more easily for the amplitude field; see Figure 6 below.

Such numerical diffusion can be better–treated by introducing a local refinement technique at the cost of complication; smaller local steps can be performed at the places where large angle raypaths exist. Here, however, we retain only the dynamic step–sizing without local refinements.

NUMERICAL EXPERIMENTS

In order to give some idea of the accuracy and the convergence obtainable with the ENO difference schemes introduced in the previous section, we have carried out several tests using the constant velocity for which comparison with analytic quantities is straightforward. The computed traveltime and amplitude (TT/AMP) fields are made use of for the prestack Kirchhoff depth migration algorithm presented in Appendix A.

Finite difference traveltimes and amplitudes

In the experiments of TT/AMP computation, the constant velocity field (v = 1000 m/s) is sampled on a uniform mesh of grid size $h = \Delta x = \Delta y = \Delta z$, the uniform finite difference (UFD) grid. We take a sequence of step sizes to check the relationship between the step size and the relative maximum error. The partial step size $\Delta z_{\rm CFL} \le \Delta z$ is determined dynamically satisfying the CFL condition (11) and several partial steps complete the UFD step of size h.

The numerical results to be presented are obtained from the following computational parameters: The domain of computation $\Omega = (1680, 3320) \times (-660, 660) \times (0, 1000)$, the source point $\mathbf{x}_s = (2500, 0, 8)$, the datum depth $z_d = 100$, and the maximum aperture $\theta_{\max} = 75°$.

Figures 3 and 4 show the contours of the traveltime fields and the amplitudes, respectively, evaluated on the plain $\{(x, y, z) : z = 960\} \cap \Omega$ for the grid $h = 40$ (top) and $h = 10$ (bottom). Solid curves are the analytic quantities and dashed curves denote the computed ones. The traveltime fields are obtained accurately, as one can expect from the theory that the traveltime error converges in second–order with respect to the grid size h. The computed amplitudes seem to convergence in accuracy order not smaller than one, as the grid size decreases.

In Figures 5 and 6, we present the plots of the traveltime fields and the amplitudes, respectively, on $\{(x, y, z) : y = 0,\ z = 960\} \cap \Omega$. Solid line is the analytic traveltime, dotted line is the traveltime of $h = 40$, dash–dotted line is that of $h = 20$, and dashed line denotes that of $h = 10$. The computed traveltime fields seem to be accurate enough already for the coarse mesh of grid size $h = 40$, while computed amplitudes again show numerical convergence better than first–order as the grid size decreases. The amplitude error appears mainly at the center where the traveltime and amplitude fields have larger curvatures and where the rays propagate with a small angle. We believe that the error is mostly the numerical diffusion partially caused by the least–oscillatory choices of finite differences (ENO) and partially caused by the relatively large numerical domain of dependence.

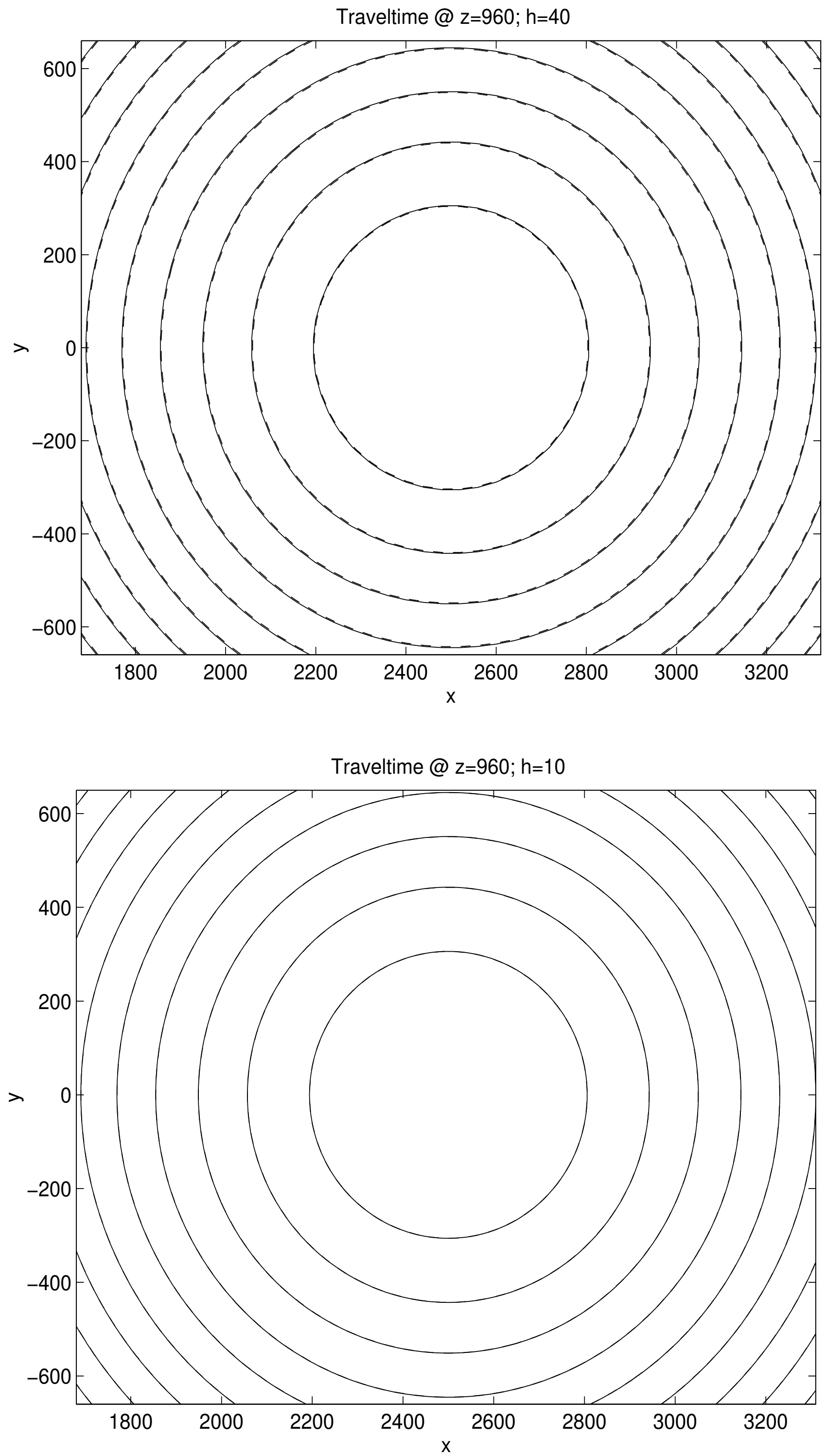

FIG. 3. Traveltimes on $\{z = 960\}$ with $h = 40$ (top) and $h = 10$ (bottom). Solid curves are the analytic traveltimes and dashed curves denote the computed traveltimes.

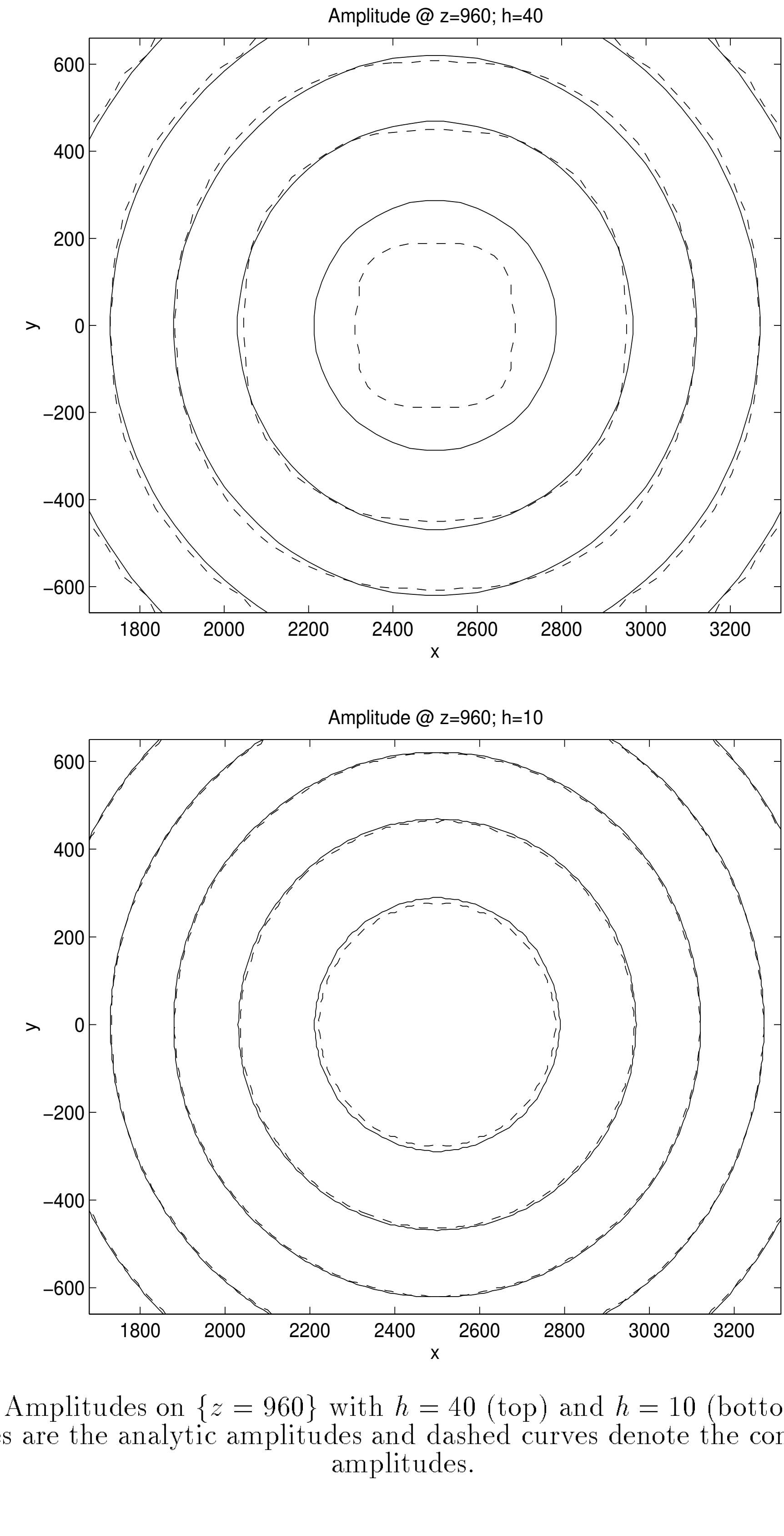

Fig. 4. Amplitudes on $\{z = 960\}$ with $h = 40$ (top) and $h = 10$ (bottom). Solid curves are the analytic amplitudes and dashed curves denote the computed amplitudes.

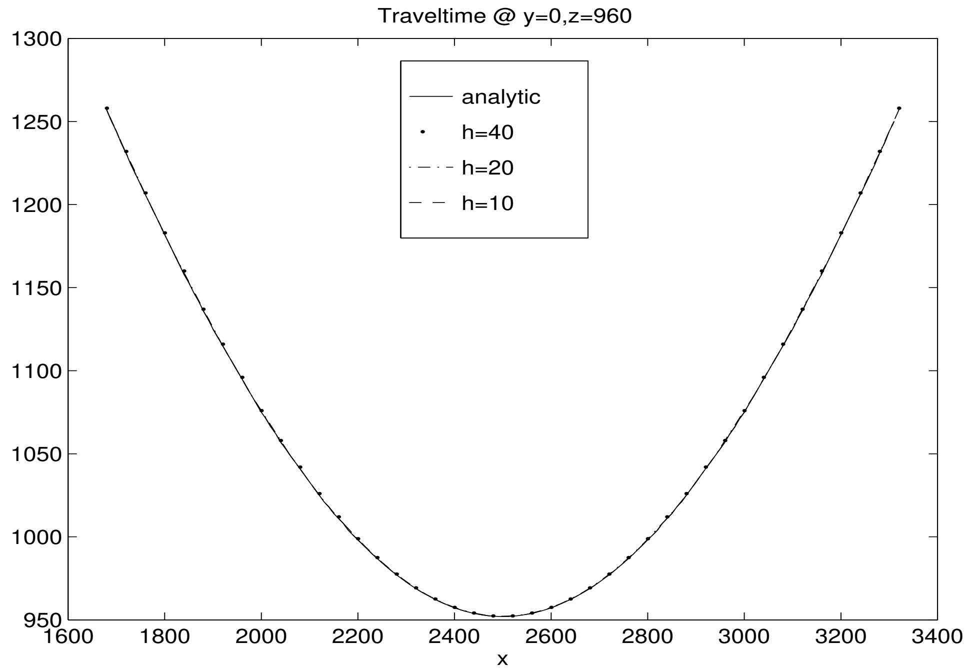

FIG. 5. Analytic and computed traveltimes on $\{(x, y, z) : y = 0,\ z = 960\}$. Solid line is the analytic traveltime, dotted line is the traveltime of $h = 40$, dash–dotted line is that of $h = 20$, and dashed line denotes that of $h = 10$.

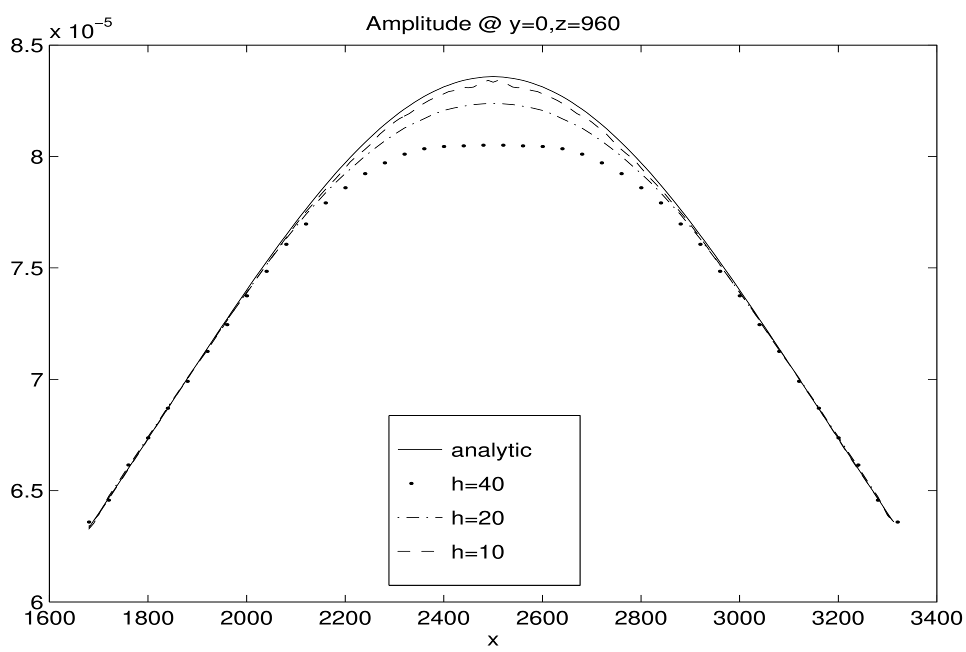

FIG. 6. Analytic and computed amplitudes on $\{(x, y, z) : y = 0,\ z = 960\}$. Solid line is the analytic amplitude, dotted line is the amplitude of $h = 40$, dash–dotted line is that of $h = 20$, and dashed line denotes that of $h = 10$.

Table 1. Convergence rates of relative errors in computed traveltimes and amplitudes. The errors are measured on $\{(x, y, z) : z = 960\}$.

	Traveltime		Amplitude	
h	$E(\tau, h)$	$\alpha(\tau, h)$	$E(a, h)$	$\alpha(a, h)$
80	7.12e–3		1.26e–1	
40	1.22e–3	2.55	3.64e–2	1.79
20	3.86e–4	1.66	1.42e–2	1.36
10	9.77e–5	1.98	6.02e–3	1.24

Table 2. The migration parameters.

velocity	$v \equiv 2000$ m/s
reflectivity domain	$(1000, 3000) \times (-1000, 1000) \times (0, 2000)$ m
reflectivity grid size	$\Delta x_{\text{refl}} = \Delta y_{\text{refl}} = \Delta z_{\text{refl}} = 20$ m
source	a 30Hz Gaussian source at $(2000, 0, 8)$
record	41×41 receivers at $\{(x_i, y_j, 8) : i, j = 0, \cdots, 40\}$, where $x_i = 1000 + 50i$ and $y_j = -1000 + 50j$
sampling interval	4 ms

In Table 1, we present the relative maximum errors and their convergence rates with respect to h. The relative errors in the computed traveltimes and amplitudes, by definition, are

$$E(\tau, h) = \frac{\|\tau^h_{\text{analytic}} - \tau^h_{\text{FD}}\|_\infty}{\|\tau^h_{\text{analytic}}\|_\infty}, \quad E(a, h) = \frac{\|a^h_{\text{analytic}} - a^h_{\text{FD}}\|_\infty}{\|a^h_{\text{analytic}}\|_\infty},$$

where τ^h_{analytic} and a^h_{analytic} are the analytic solutions and τ^h_{FD} and a^h_{FD} are the finite difference solutions for a given grid size h. The convergence rates α are given as

$$\alpha(\tau, h) = \frac{\log(E(\tau, 2h)/E(\tau, h))}{\log 2}, \quad \alpha(a, h) = \frac{\log(E(a, 2h)/E(a, h))}{\log 2}.$$

We can see from the table that accuracy order for the amplitude field is about 1.3 in average, asymptotically approaching one. The average number of partial steps in an UFD step is approximately 5.2.

Kirchhoff depth migration

To test the Kirchhoff depth migration algorithm presented in Appendix A, we set the migration parameters as in Table 2 and the reflectors as in Figure 7.

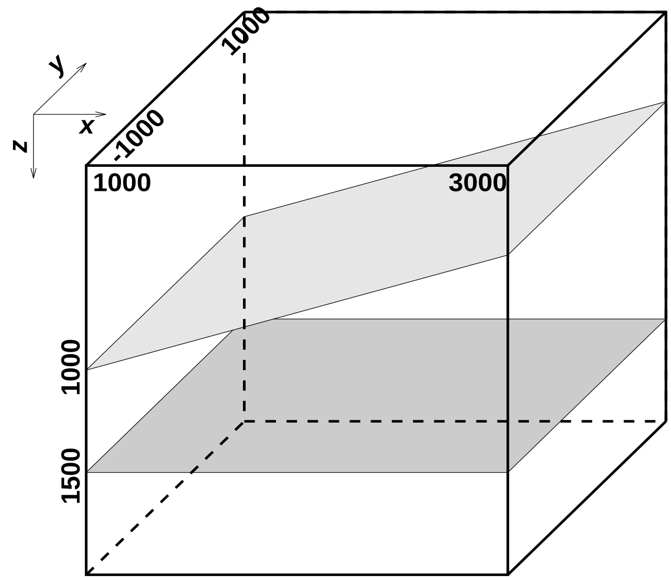

FIG. 7. The reflectors. The dip reflector is $\{z = -(x - 1000)\tan 15^\circ + 1000\}$, 15° dip in x–direction, and the flat one is $\{z = 1500\}$.

For each source/receiver position, we should establish a congruous computational domain to compute the corresponding TT/AMP; it does not have to include, or be contained in, the reflectivity domain. We choose the computational domain depending on the maximum angle, the vertically averaged slowness, and the maximum sampling time so that it can reach out not farther than maximum 320m from each vertical reflectivity boundary. Also, the migration algorithm is allowed to resample a coarse mesh from the expanded reflectivity mesh for the TT/AMP computation, which saves computational effort for smoother models. In the test, we choose 4 for the resampling rate, i.e. the TT/AMP computation is performed on the mesh of grid size 80m. The Kirchhoff summation is performed on the reflectivity mesh with the TT/AMP tri–linearly interpolated. The maximum angle $\theta_{\max} = 75^\circ$ and the datum depth $z_d = 100$ m for all cases.

Figure 8 shows the migrated depth sections. The migration is nearly perfect; the sides of the reflectors were not imaged because the seismic energy reflected by the locations reached at outside the modeling boundaries. Note that the wavelength set in the experiment is

$$\frac{2000 \text{ m/s}}{30 \text{ cy/s}} \approx 66.7 \text{ m/cy}$$

and that the reflectivity grid size is 20 m. So, we choose only approximately 3 migration grid points per wavelength which explains the noisy image for the dip reflector. The migration was carried out on a single processor of the SGI array (75MHz T8000), so there was no CPU time advantage obtainable by exploiting the parallel nature of the TT/AMP computations and Kirchhoff summations, although they could be fully vectorized. The total elapsed time was about 89 minutes.

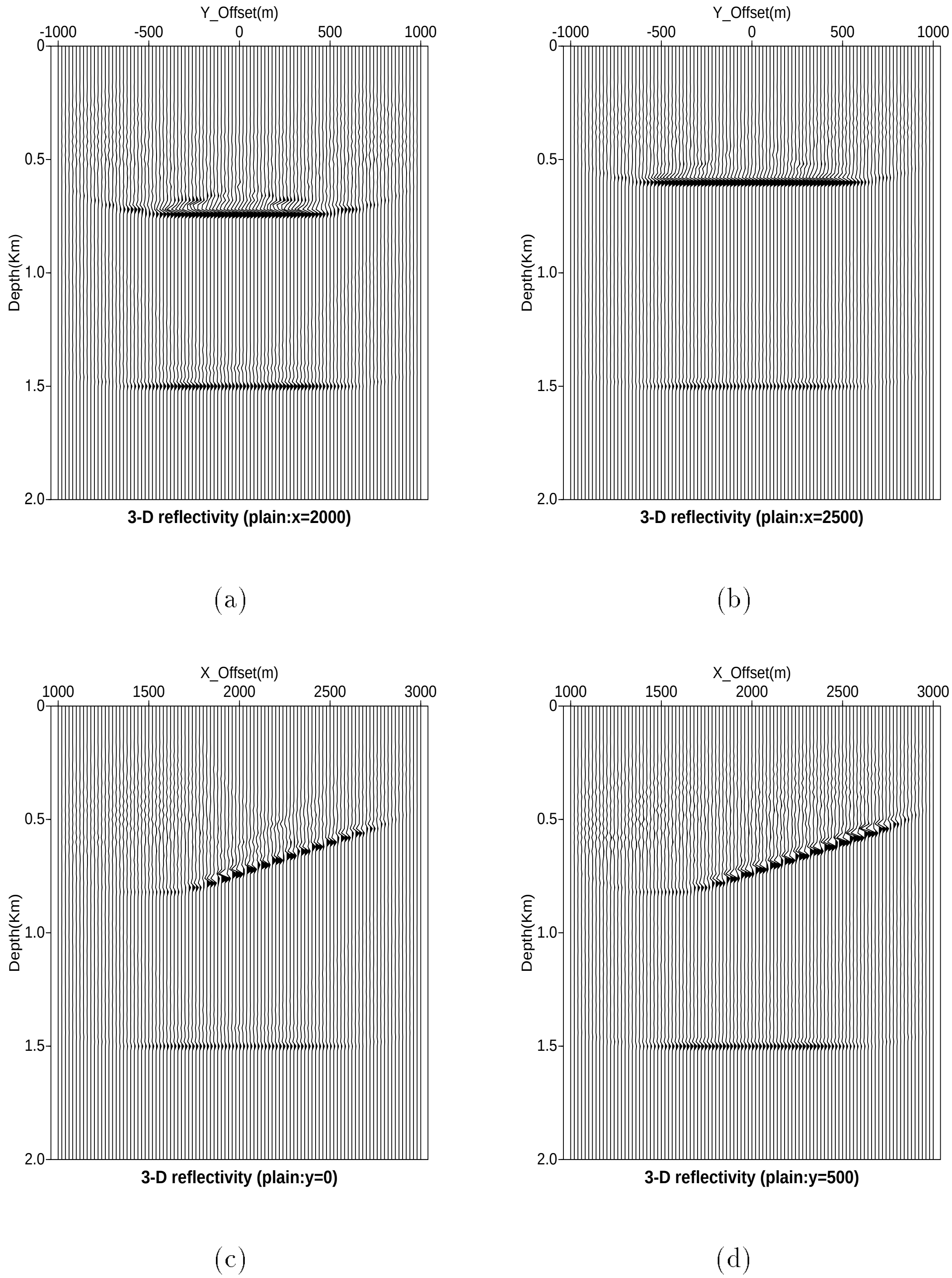

FIG. 8. The migrated depth sections on the plains: (a) $\{x = 2000\}$, (b) $\{x = 2500\}$, (c) $\{y = 0\}$, and (d) $\{y = 500\}$.

CONCLUSIONS AND FUTURE WORK

We have described a second–order ENO algorithm solving the eikonal and transport equations in 3–D. To improve numerical accuracy, the problem is formulated in a uniform mesh (in x– and y–directions) and special care has been taken in boundary data treatment and finite differences for the transport equation. To treat boundary data more effectively, an outer border is added to the computational domain so that the solution is linearly extrapolated outside the model boundaries. The z–derivatives of the traveltime field in the transport equation have been replaced by a combination of more convenient derivatives that are approximated by centered or mostly–centered finite differences, resulting in an extra–order accuracy for the computed amplitudes. Numerical examples show that *the computed amplitudes have order of accuracy at least one.* The eikonal/transport solver has been incorporated into a 3–D Kirchhoff migration algorithm, resulting in accurate depth migrations.

Since computation of the traveltime and amplitude field is the most costly component of Kirchhoff migration, both the exploitation of fast computational algorithms and the use of vector/parallel computers are extremely interesting; the finite difference eikonal/transport solver is being parallelized. We are also investigating strategies that allow the Kirchhoff migration to efficiently produce images in a portion of the subsurface (e.g. a horizontal slab), using a fraction of the available data.

Acknowledgement and Disclaimer: This work was partially supported by the National Science Foundation under grant numbers DMS 9404283 and DMS 9627355; the Office of Naval Research under grant number N00014-96-1-0156; the Air Force Office of Scientific Research, Air Force Materials Command, USAF, under grant number F49620-95-1-0339; and The Rice Inversion Project. TRIP Sponsors for 1997 are Advance Geophysical, Amoco Production Co., Conoco Inc., Cray Research Inc., Discovery Bay, Exxon Production Research Co., Interactive Network Technologies, Mobil Research and Development Corp., and Shell International Research. The US Government is authorized to reproduce and distribute reprints for governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the Air Force Office of Scientific Research, the Office of Naval Research, the National Science Foundation, or the US Government.

REFERENCES

Cerveny, V. (1985). Ray synthetic seismograms for complex two- and three-dimensional structures. *J. Geophys.*, 58:2–26.

Cerveny, V., Molotkov, I. A., and Psencik, I. (1977). *Ray Methods in Seismology.* University of Karlova Press, Prague.

El-Mageed, M. A. A. (1996). *3D first arrival traveltimes and amplitudes via Eikonal and Transport finite difference solvers.* PhD thesis, Department of Computational and Applied Mathematics, Rice University, 6100 Main St., Houston Texas 77005, USA.

Friedlander, F. (1958). *Sound Pulses.* Cambridge University Press.

Gray, S. and May, W. (1994). Kirchhoff migration using eikonal equation traveltimes. *Geophysics*, 59:810–817.

Křížek, M. and Neittaanmäki, P. (1984). Superconvergence phenomenon in the finite element method arising from averaging gradients. *Numer. Math.*, 45:105–116.

Osher, S. and Sethian, J. (1988). Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations. *J. Comp. Phys.*, 79:12–49.

Osher, S. and Shu, C.-W. (1991). High–order essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SIAM J. Numer. Anal.*, 28:907–922.

Rakesh (1988). A linearized inverse problem for the wave equation. *Comm. on P.D.E.*, 13(5):573–601.

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

Strikwerda, J. C. (1989). *Finite Difference Schemes and Partial Differential Equations.* Wadsworth & Brooks/Cole, Pacific Grove, California.

Symes, W., Versteeg, R., Sei, A., and Tran, Q.-H. (1994). Kirchhoff simulation migration and inversion using finite–difference traveltimes and amplitudes. TRIP Annual Report, The Rice Inversion Project, Rice University, Houston, TX 77005.

Symes, W. W. (1995). *Mathematics of Reflection Seismology.* The Rice Inversion Project, Rice University, Houston, TX 77005.

van Trier, J. and Symes, W. W. (1991). Upwind finite-difference calculation of traveltimes. *Geophysics*, 56(6):812–821.

Vidale, J. (1988). Finite-difference calculation of travel times. *Bull., Seis. Soc. Am.*, 78:2062–2076.

Appendix A. KIRCHHOFF MIGRATION

For the sake of completeness, here we present the Kirchhoff migration algorithm. For more detailed descriptions, see e.g. (Symes et al., 1994; Symes, 1995).

Derivation

The acoustic field $\widehat{p}(\mathbf{x},\mathbf{x}_s,t)$ $(\mathbf{x},\mathbf{x}_s \in \mathbb{R}^3,\ t \in \mathbb{R})$ of an isotropic point radiator with source $f(t)$ located at $\mathbf{x}_s$ satisfies

$$\begin{gathered}\left(\frac{1}{\rho\widehat{v}^2(\mathbf{x})}\frac{\partial^2}{\partial t^2} - \nabla\cdot\frac{1}{\rho}\nabla\right)\widehat{p}(\mathbf{x},\mathbf{x}_s,t) = f(t)\delta(\mathbf{x}-\mathbf{x}_s),\\ \widehat{p}(\mathbf{x},t) \equiv 0, \quad t \ll 0,\end{gathered} \tag{A-1}$$

where ρ is the density at equilibrium, assumed to be constant in this article, and $\widehat{v}(\mathbf{x})$ denotes the sound velocity. The first order perturbation δp of $\widehat{p}\,(:= p + \delta p)$ resulting from the perturbation δv of $\widehat{v}\,(:= v + \delta v)$ reads

$$\left(\frac{1}{\rho v^2(\mathbf{x})}\frac{\partial^2}{\partial t^2} - \nabla\cdot\frac{1}{\rho}\nabla\right)\delta p(\mathbf{x},\mathbf{x}_s,t) = \frac{2r(\mathbf{x})}{\rho v^2(\mathbf{x})}\,\frac{\partial^2}{\partial t^2}\widehat{p}(\mathbf{x},\mathbf{x}_s,t), \tag{A-2}$$

where $r(\mathbf{x}) = \delta v(\mathbf{x})/v(\mathbf{x})$, the reflectivity that is the oscillatory relative perturbation of the smooth component of the velocity v and is dimensionless.

Let $\widehat{G}(\mathbf{x}_s,\mathbf{x},t)$ be the causal Green's function, the solution of (A-1) for $f(t)=\delta(t)$, and $G(\mathbf{x}_r,\mathbf{x},t_r-t)$ be the incident field of the Green's function, the solution of (A-2) with the right hand side replaced by $\delta(t)\delta(\mathbf{x}-\mathbf{x}_s)$. Then, using Green's identity and equation (A-2), we find that the reflection field

$$\delta p(\mathbf{x}_s,\mathbf{x}_r,t_r) = f *_{t_r} \int d\mathbf{x}\int dt\left(\frac{2r(\mathbf{x})}{\rho v^2(\mathbf{x})}\,\frac{\partial^2}{\partial t^2}\widehat{G}(\mathbf{x}_s,\mathbf{x},t)\right)G(\mathbf{x}_r,\mathbf{x},t_r-t). \tag{A-3}$$

Assume that the incident field comes close to the total field:

$$\widehat{G}(\mathbf{x}_s,\mathbf{x},t) \approx G(\mathbf{x}_r,\mathbf{x},t);$$

which is called the Born approximation in literature. Since geophysical data reside in the high frequency regime, the Green's function can be replaced by the WKBJ approximation

$$G(\mathbf{x},\mathbf{x}',t) \;\approx\; a(\mathbf{x},\mathbf{x}')\,\delta(t-\tau(\mathbf{x},\mathbf{x}')),$$

where $\tau(\mathbf{x},\mathbf{x}')$ and $a(\mathbf{x},\mathbf{x}')$ indicate the traveltime and the amplitude, respectively, of the waves propagating from $\mathbf{x}'$ to $\mathbf{x}$, satisfying the following eikonal and transport equations

$$\begin{gathered}\nabla\tau\cdot\nabla\tau = \frac{1}{v^2(\mathbf{x})},\\ 2\nabla\tau\cdot\nabla a + a\nabla^2\tau = 0.\end{gathered} \tag{A-4}$$

Using the above approximations, we can rewrite (A-3) as

$$\begin{aligned}\delta p(\mathbf{x}_s,\mathbf{x}_r,t_r) &\approx f *_{t_r} \frac{\partial^2}{\partial t_r^2}\int d\mathbf{x}\,\frac{2r(\mathbf{x})a_s(\mathbf{x})a_r(\mathbf{x})}{\rho v^2(\mathbf{x})}\,\delta(t_r-\tau_s(\mathbf{x})-\tau_r(\mathbf{x}))\\ &= f *_{t_r} \frac{\partial^2}{\partial t_r^2}\int d\mathbf{x}\,A(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x})r(\mathbf{x})\,\delta(t_r-\tau_s(\mathbf{x})-\tau_r(\mathbf{x})),\end{aligned} \tag{A-5}$$

where $a_e(\mathbf{x}) = a(\mathbf{x},\mathbf{x}_e)$ and $\tau_e(\mathbf{x}) = \tau(\mathbf{x},\mathbf{x}_e)$, $e = s,\ r,$ and

$$A(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x}) = \frac{2a_s(\mathbf{x})\,a_r(\mathbf{x})}{\rho v^2(\mathbf{x})}.$$

Equation (A-5) is the Kirchhoff representation of the primaries only acoustic reflection field.

We are going to view the correspondence between the reflectivity r and the sampled field δp as a *linear operator* $L : r \rightarrow \delta p$. We will also idealize reality by assuming that the receiver positions form a continuum indexed by (x_r, y_r), with z_r fixed. We concentrate on the computation of the adjoint of this map for a single shot and drop $\mathbf{x}_s$ from the notation. The L^2 adjoint of L is a *prestack migration operator*, defined as

$$\langle Lr(\mathbf{x}_r,t_r), S(\mathbf{x}_r,t_r)\rangle_{\mathbf{x}_r,t_r} = \langle r(\mathbf{x}), L^*S(\mathbf{x})\rangle_{\mathbf{x}}$$

with $S(\mathbf{x}_r,t_r)$ being the seismic record; see e.g. (Rakesh, 1988). Integration by parts shows that

$$\begin{aligned}L^*S(\mathbf{x}) &= \int\int d\mathbf{x}_r dt_r\, A(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x})\Big(\check{f} *_{t_r} \frac{\partial^2}{\partial t_r^2}S(\mathbf{x}_r,t_r)\Big)\delta(t_r-\tau_s(\mathbf{x})-\tau_r(\mathbf{x}))\\ &= \int d\mathbf{x}_r\, A(\mathbf{x}_s,\mathbf{x}_r,\mathbf{x})\Big(\check{f} *_{t_r} \frac{\partial^2}{\partial t_r^2}S\Big)(\mathbf{x}_r,\tau_s(\mathbf{x})+\tau_r(\mathbf{x})).\end{aligned} \tag{A-6}$$

Here we have utilized the well–known identity

$$\int (f*g)h = \int (\check{f}*h)g, \quad \check{f}(t) = f(-t).$$

Discretization

Assume that all grids are uniform for simplicity. First we summarize some notation to be used in the remainder of the section.

Indices:

- i is the x_r index.
- j is the y_r index.
- n is the t_r index.

- k is the x index.
- ℓ is the y index.
- m is the z index.

Points:

- $\mathbf{x}_{k,\ell,m} = (x_k, y_\ell, z_m) = (x_{\min} + (k-1)\Delta x,\ y_{\min} + (\ell-1)\Delta y,\ z_{\min} + (m-1)\Delta z)$.
- $\mathbf{x}_{r,i,j} = (x_{r,i}, y_{r,j}) = (x_{r,\min} + (i-1)\Delta x_r,\ y_{r,\min} + (j-1)\Delta y_r)$.
- $t_{r,n} = t_{r,\min} + (n-1)\Delta t_r$.

Let $u_{k,\ell,m} = u(x_k, y_\ell, z_m)$ for functions $u(\mathbf{x})$ of the space coordinates and $w_{i,j,n} = w(x_{r,i}, y_{r,j}, t_{r,n})$ for functions $w(\mathbf{x}_r, t_r)$ of the sampling coordinates.

Grid–functions:

- $A_{i,j,k,\ell,m} = \dfrac{2a_{s,k,\ell,m}\, a_{r,i,j,k,\ell,m}}{\rho v_{k,\ell,m}^2}$.
- $\phi_{i,j,k,\ell,m} = \tau_{s,k,\ell,m} + \tau_{r,i,j,k,\ell,m}$, where $\phi(\mathbf{x}_r, \mathbf{x}) := \tau_s(\mathbf{x}) + \tau_r(\mathbf{x})$.

Assume the integrand in (A-6) vanishes at the boundaries of the grid; the trapezoidal rule is equivalent to the left hand rule:

$$(L^*S)_{k,\ell,m} = \Delta x_r \Delta y_r \sum_{i,j} A_{i,j,k,\ell,m} \cdot \frac{\partial^2}{\partial t_r^2}\Big(\check{f} *_{t_r} S\Big)(\mathbf{x}_{r,i,j}, \phi_{i,j,k,\ell,m}) + \mathcal{O}(|\Delta \mathbf{x}_r|^2). \quad \text{(A-7)}$$

We approximate the evaluation in t_r by piecewise linear interpolation, then the second–order t_r–derivative by the central difference formula: set

$$\begin{aligned} N_{i,j,k,\ell,m} &= \operatorname{int}\left(\frac{\phi_{i,j,k,\ell,m}}{\Delta t_r}\right), \\ T_{i,j,k,\ell,m} &= \phi_{i,j,k,\ell,m} - \Delta t_r \cdot N_{i,j,k,\ell,m}, \\ D_{t_r}^2 U_p &= \frac{U_{p+1} - 2U_p + U_{p-1}}{\Delta t_r^2}, \end{aligned}$$

for any sampled function U of t_r. Then

$$\begin{aligned} \frac{\partial^2}{\partial t_r^2}\Big(\check{f} *_{t_r} S\Big)(\mathbf{x}_{r,i,j}, \phi_{i,j,k,\ell,m}) &= (1 - T_{i,j,k,\ell,m}) D_{t_r}^2 \Big(\check{f} *_{t_r} S\Big)_{i,j,N_{i,j,k,\ell,m}} \\ &\quad + T_{i,j,k,\ell,m} D_{t_r}^2 \Big(\check{f} *_{t_r} S\Big)_{i,j,N_{i,j,k,\ell,m}+1} + \mathcal{O}(\Delta t_r^2). \end{aligned}$$

So finally we obtain

$$\begin{aligned} (L^*S)_{k,\ell,m} = \Delta x_r \Delta y_r \sum_{i,j} \Big[A_{i,j,k,\ell,m} \times \\ \Big\{(1 - T_{i,j,k,\ell,m}) D_{t_r}^2 \Big(\check{f} *_{t_r} S\Big)_{i,j,N_{i,j,k,\ell,m}} + T_{i,j,k,\ell,m} D_{t_r}^2 \Big(\check{f} *_{t_r} S\Big)_{i,j,N_{i,j,k,\ell,m}+1}\Big\}\Big] \quad \text{(A-8)} \\ + \mathcal{O}(|\Delta \mathbf{x}_r|^2) + \mathcal{O}(\Delta t_r^2). \end{aligned}$$

The time convolution in the above equation can be easily approximated by the trapezoidal rule as follows:

$$\begin{aligned}(\check{f} *_{t_r} S)(\mathbf{x}_r, n\Delta t_r) &= \int dt'\, f(-t')S(\mathbf{x}_r, n\Delta t_r - t') \\ &= \int dt'\, f(t')S(\mathbf{x}_r, n\Delta t_r + t') \;\approx\; \Delta t_r \sum_q f(q\Delta t_r) S(\mathbf{x}_r, (n+q)\Delta t_r).\end{aligned}$$

Throwing away the discretization error we take (A-8) as the definition of the (second–order) discretization of L^*. For a detailed description concerning the discretization error, see (Symes et al., 1994).

Algorithm

Now, we express explicitly the discrete algorithm for the Kirchhoff depth migration. One feasible arrangement of the algorithm for migration:

1. Precompute the scaling field
$$B_{k,\ell,m} = \frac{2\Delta x_r \Delta y_r}{\rho v^2_{k,\ell,m}}$$
2. Initialize output depth section to zero
3. Compute $\tau_{s,k,\ell,m}$ and $a_{s,k,\ell,m}$
4. Do for each receiver point (i,j):
   (a) compute $\tau_{r,i,j,k,\ell,m}$ and $a_{r,i,j,k,\ell,m}$
   (b) compute $H_{i,j,n} = D^2_{t_r}\left(\check{f} *_{t_r} S\right)_{i,j,n}$ for $n = n_{\min}, \ldots, n_{\max}$
   (c) do for each depth point (k,ℓ,m):
      i. compute
      $$P_{i,j,k,\ell,m} = B_{k,\ell,m}\, a_{s,k,\ell,m}\, a_{r,i,j,k,\ell,m}$$
      ii. calculate $N_{i,j,k,\ell,m}$ and $T_{i,j,k,\ell,m}$
      iii. add to (k,ℓ,m) sample of output depth section
      $$P_{i,j,k,\ell,m}\Big[(1 - T_{i,j,k,\ell,m})H_{i,j,N_{i,j,k,\ell,m}} + T_{i,j,k,\ell,m} H_{i,j,N_{i,j,k,\ell,m}+1}\Big]$$

   end do

   End do