An Adaptive Stencil Finite Difference Scheme For First Order Linear Hyperbolic Systems

Robert H. Hoar [1]

Department of Mathematical Sciences, Montana State University, Bozeman MT 59717, e-mail: `hoar@math.montana.edu`.

C. R. Vogel [2]

Department of Mathematical Sciences, Montana State University, Bozeman MT 59717, e-mail: `vogel@math.montana.edu`.

Abstract

A numerical method is presented to deal with the difficulties associated with discontinuous coefficients and data in hyperbolic partial differential equations. The method applies directly to first order linear systems. It combines aspects of the Courant-Rees-Isaacson method, the essentially non-oscillatory derivative approximations of Harten and Osher, and, to propagate solutions in more than one space dimension, Strang splitting. Several examples in one and two space dimensions are presented which indicate that the method maintains sharp waveforms without introducing spurious oscillations. One of the examples, taken from [Dablain, *The application of high-order differencing to the scalar wave equation*, Geophysics, vol. 51 (1985), pp. 54-66], allows a comparison with existing finite difference methods.

Keywords: hyperbolic partial differential equations, finite difference methods, essentially non-oscillatory (ENO) methods, Strang splitting.

1 Introduction

The simulation of wave propagation through highly heterogeneous media is important in many applications, e.g., seismic and ultrasound modeling and imaging. In these applications, material parameters like density and wave propagation speed may change drastically over short length scales. In practice,

[1] Research was supported in part by a DOE-EPSCoR Graduate Fellowship.

[2] Research was supported in part by the NSF under Grant DMS-9303222.

these rapid variations are often modeled with hyperbolic partial differential equations having nonsmooth coefficients.

To solve these equations, asymptotic techniques may sometimes be applied. They can effectively handle discontinuities, but they are difficult to implement if the geometry of the discontinuities becomes too complex. Numerical methods—primarily finite difference methods—are a widely used alternative. While relatively easy to implement, these methods may perform poorly when applied to equations with nonsmooth coefficients or nonsmooth initial or boundary data. Even for equations with smoothly varying coefficients and data, standard finite difference methods suffer from numerical dispersion, i.e., the propagation of differing frequency components at different speeds. This is manifested in the spread of waveforms and the introduction of spurious oscillations (see [2,9]).

In this paper, we present an adaptive stencil scheme to deal with these numerical difficulties. This scheme combines elements of the Courant-Isaacson-Rees (CIR) method [1], the essentially non-oscillatory (ENO) scheme of Osher and Harten [4], and Strang splitting [6]. While the method is adaptive in nature, it does not seek to increase accuracy or resolution by refining the computational grid. Instead, it uses a fixed regular grid and adapts the finite difference stencil from point to point. This strategy eliminates spurious oscillations, reduces the spreading of waveforms, and allows numerical propagation of waveforms across discontinuities in spatially dependent coefficients. The adaptive nature of the mesh selection strategy makes the method nonlinear—in spite of the fact that it is designed for linear hyperbolic systems in one or more spatial dimensions.

To illustrate this method, we describe in detail its implementation for the scalar wave equation. Section 2 deals with the case of one space dimension, while section 3 covers the 2-D case. Subsection 2.1 reviews the conversion of the 1-D scalar wave equation to first order system form and the implementation of the CIR method for this linear system. The CIR method exhibits only first order accuracy. In subsection 2.2, we utilize the ENO criterion to develop a stencil selection strategy which has higher order spatial accuracy. This strategy is illustrated with an example in subsection 2.3. Subsection 2.4 presents an approach for increasing temporal accuracy. Subsection 2.5 addresses the issue of computational complexity. In subsection 3.1, we review the conversion of the 2-D scalar wave equation to first order system form. Subsection 3.2 describes the implementation of Strang splitting for this system. This allows us to accurately propagate solutions in two (or more) space dimensions using a product of 1-D propagators.

Numerical results in one and two space dimensions are presented in sections 4 and 5, respectively. Results for a one dimensional initial value problem (IVP) for the scalar wave equation with non-smooth initial data are presented in subsection 4.1. A second IVP, this one with a discontinuous coefficient (wave speed), is given in subsection 4.2. This second example was taken from [2], allowing a comparison of the adaptive stencil scheme with existing finite difference methods. Subsection 4.3 briefly describes how boundary conditions are handled numerically. Subsections 5.1 and 5.2 present results for a pair of IVP's for the two dimensional scalar wave equation, each having discontinuous wave speeds. In the second 2-D example, the discontinuity is not aligned with the computational grid.

2 The One Dimensional Case

In this section, we outline the adaptive stencil finite difference method for the one dimensional acoustic wave equation. First, we review the CIR method; then we extend it to obtain higher order methods.

2.1 The CIR Method

Courant, Isaacson and Rees introduced what has become known as the CIR method in 1952 [1] (see also [3, p. 48]). The adaptive stencil method presented here may be viewed as an extension of CIR.

To illustrate, consider the one dimensional scalar, or acoustic, wave equation

$$\partial_t^2 u - c^2 \partial_x^2 u = 0. \tag{1}$$

This is equivalent to the first order system

$$\partial_t \vec{v} + A\partial_x \vec{v} = \vec{0}, \tag{2}$$

where

$$\vec{v} = \begin{bmatrix} \partial_x u \\ \partial_t u \end{bmatrix}, \quad A = \begin{bmatrix} 0 & -1 \\ -c^2 & 0 \end{bmatrix}. \tag{3}$$

Let E denote the matrix of eigenvectors and let $\Lambda = diag\{\lambda_1, \lambda_2\}$ be the diagonal matrix of corresponding eigenvalues of A, so that $A = E\Lambda E^{-1}$. Then (2) becomes

$$E^{-1}\partial_t \vec{v} + \Lambda E^{-1}\partial_x \vec{v} = 0.$$

Setting $\vec{w} = E^{-1}\vec{v}$, we have (assuming c is a function of space but not time)

$$\partial_t \vec{w} + \Lambda \partial_x \vec{w} = \Lambda(\partial_x E^{-1}) E \vec{w}. \tag{4}$$

The left hand side is decoupled, and with the spatial and temporal discretizations $\vec{w}_j^m \equiv [w_{1j}^m \ , w_{2j}^m]^T$, where $w_{ij}^m = w_i(m\Delta t, j\Delta x)$ and $m \geq 0, \quad j = \ldots, -1, 0, 1, \ldots$, the CIR method yields

$$\frac{\vec{w}_j^{m+1} - \vec{w}_j^m}{\Delta t} + \begin{bmatrix} \lambda_1 \ \Delta_x(\lambda_1) w_{1j}^m \\ \lambda_2 \ \Delta_x(\lambda_2) w_{2j}^m \end{bmatrix} = \Lambda(\partial_x E^{-1}) E \vec{w}_j^m, \tag{5}$$

where we define for a spatial grid function $f = \{f(j\Delta x)\} = \{f_j\}$,

$$\Delta_x(\lambda) f_j = \begin{cases} \dfrac{f_j - f_{j-1}}{\Delta x}, & \text{if } \lambda > 0 \quad \bullet \ \circ \\ \dfrac{f_{j+1} - f_j}{\Delta x}, & \text{if } \lambda < 0 \quad \circ \ \bullet \end{cases} \tag{6}$$

Hence, upwinding is used in the selection of the spatial difference stencils. At any given time step, one can recover $\vec{v}$ by computing $\vec{v} = E\vec{w}$.

The symbols to the right in (6) indicate which stencil is being used. The solid dot $\bullet$ represents the point at which the spatial derivative is being approximated, and the circles $\circ$ represent other points that are to be used. For example, $\circ \ \bullet$ is the first order finite difference approximation that requires f at x_j and the point to the left, x_{j-1}.

The local truncation error for the CIR method is $O(\Delta x) + O(\Delta t)$ (see [3]). Since $|\lambda_i| = c$, the CFL stability condition reduces to

$$\max_x c(x) \frac{\Delta t}{\Delta x} \leq 1.$$

Given stability, CIR is first order accurate in both space and time. Hence, one obtains first order accurate approximations to the components of $\vec{v}$ in (2)-(3). One can then integrate the approximation of u_t with respect to t and obtain a second order accurate approximation to the solution u of (1).

Note that the stencil *adapts* to the sign of the λ_i. For this reason, CIR can be viewed as an adaptive stencil finite difference method. Also for this reason, CIR deals with numerical dispersion better than standard methods based directly on difference approximations to the second derivatives in problem (1) (see [9, pp. 80-83]).

The low order accuracy of the CIR method results in undesirable smearing of wave forms, as can clearly be seen in Figure 5 below. Higher order accuracy is required to overcome this smearing. Unfortunately, higher order approximations tend to introduce spurious oscillations. This motivated the development of ENO derivative approximation schemes, which we now describe.

2.2 ENO Higher Order Stencil Selection

Clearly $O(\Delta x^n)$ approximations to the derivative $\partial_x w_i$ at the point x_j may be obtained using $n+1$ distinct, consecutive points (see for example [8, p. 37]). These need not be centered at x_j. In fact, there are $n+1$ possible stencils containing x_j that yield approximations with an $O(\Delta x^n)$ truncation error.

For $n = 1$, there are two stencil choices. With the CIR method, the sign of the λ_i's is used to make the appropriate choice. For $n > 1$, additional criteria are needed to select the stencil. In order to minimize spurious oscillations associated with higher order approximations, we adopt the ENO criterion. This was originally developed by Harten and Osher [4] for scalar conservation laws and later extended to first order scalar hyperbolic PDE's in Hamilton-Jacobi form [7].

To determine which $(n+1)$ spatial grid points to use in the finite difference stencil, we start with the point x_j and add points in a manner based on the local smoothness of the data. The necessary decisions and resulting stencils are displayed in the tree in Figure 1. Again, the solid dot ● refers to the point x_j, and the circles ○ refer to other points that are used in the stencil.

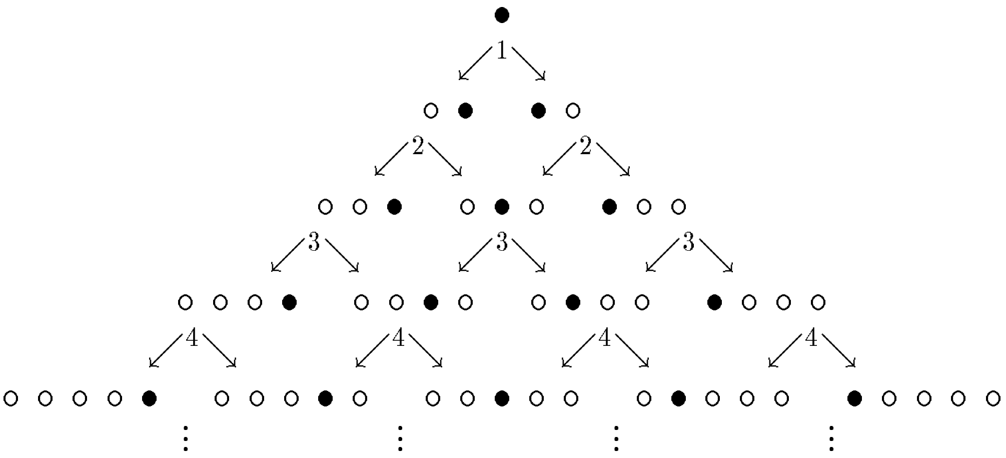

Fig. 1. Tree of possible finite difference stencils.

The $O(\Delta x^n)$ stencils are found on the $(n+1)^{st}$ level of the tree. Thus n decisions must be made in order to reach this level. At level 1, add the point that is in the upwind direction, which is determined by the sign of λ_i, as with the CIR method, c.f., (6). Take the arrow to the left at level 1 if $\lambda_i > 0$, otherwise take the arrow to the right. This ensures that the scheme is upwinding, a necessary condition for numerical stability. The remaining $n-1$ decisions are aimed at minimizing the spurious oscillations that often occur when using a fixed finite difference stencil (see [2,9] for examples).

Each finite difference stencil in Figure 1 corresponds to a polynomial interpolant. Harten and Osher [4] have shown that choosing polynomial interpolants with small (in magnitude) higher derivatives will reduce the spurious oscillations. This selection criterion yields what are known as essentially non-oscillatory (ENO) methods. Given a vector f of length N, let $D(f)$ be the vector of length $N-1$ whose j^{th} entry $D(f)_j$ is $f_{j+1} - f_j$ and define $D^n(f) = D(D^{n-1}(f))$ for $n > 1$. Consider the decision that must be made at level k. The arrow to the left is taken if

$$|D^k(f)_i| < |D^k(f)_{i+1}|, \tag{7}$$

otherwise the one on the right is taken. This choice ensures that the method will choose the stencil on the left at decision k if the leading coefficient of the degree k polynomial interpolating the data represented on the left is smaller in magnitude than the leading coefficient for the data on the right, which is the ENO criterion.

The value of i in (7) will depend on the results of the previous decisions. If l is the number of previous decisions that resulted in taking a path to the left (i.e., the number of $\circ$'s to the left of $\bullet$ in the current stencil), then $i = j - l$. Hence, the resulting method, even though applied to a linear system, is nonlinear.

2.3 An Example

To illustrate the advantages of the ENO stencil selection scheme, consider first derivative approximations for the piecewise smooth function in Figure 2. Below are the three possible approximations to $f'(x_j)$ based on 3-point difference stencils:

$$\begin{array}{ll} \dfrac{f_{j-2} - 4f_{j-1} + 3f_j}{2\Delta x} & \circ \; \circ \; \bullet \\ \dfrac{-f_{j-1} + f_{j+1}}{2\Delta x} & \circ \; \bullet \; \circ \\ \dfrac{-3f_j + 4f_{j+1} - f_{j+2}}{2\Delta x} & \bullet \; \circ \; \circ \end{array} \tag{8}$$

The circles and solid dots have the same meaning as in equation (6). For $f \in C^4$, each approximation is $\mathcal{O}(\Delta x^2)$ accurate.

Since f is not differentiable, we seek approximations to the left and right derivative at each point. In frame 2 of Figure 2, we present the values that the stencil $\bullet \circ \circ$ would assign to an approximation of the derivative by superimposing a line that passes through each point that has a slope that is equal to the approximation of the derivative at that point. Note that for $x = 0, 3, 4, 5$ or 6, the lines shown appear to be tangent, implying that the approximations are fairly accurate. Note also that at $x = 2$ and $x = 8$, the lines accurately approximate the tangents corresponding to the right derivative. Unfortunately, this stencil does not work well at $x = 1$, or $x = 7$, and the stencil may not be applied at $x = 9$ or $x = 10$ (there are not enough nodes to the right). The reason it does not do well at $x = 1$, for example, is that f is not smooth enough on the interval containing the nodes that are used; in this case the interval (1,3).

In frames 3 and 4 of Figure 2, we present similar results obtained by applying the centered stencil $\circ \bullet \circ$ and the stencil $\circ \circ \bullet$, respectively. Clearly, if any one fixed stencil is used, significant errors will be made. The last two frames show the approximations that would be assigned by the adaptive stencil selection scheme. The dash-dot lines were obtained from the centered stencil $\circ \bullet \circ$. The solid and dashed lines were obtained from the stencils $\bullet \circ \circ$ and $\circ \circ \bullet$, repectfully.

2.4 Temporal Accuracy

Expanding the solution to

$$\partial_t \vec{v} + A \partial_x \vec{v} = 0 \tag{9}$$

in a Taylor series about t yields

$$\vec{v}(x, t + \Delta t) = (1 + \Delta t \partial_t + \frac{1}{2} \Delta t^2 \partial_t^2 + \ldots) \vec{v}(x, t). \tag{10}$$

Using (9) to replace the temporal derivatives with spatial derivatives, we obtain

$$\vec{v}(x, t + \Delta t) = (1 - \Delta t A \partial_x + \frac{1}{2} \Delta t^2 (A \partial_x)^2 - \ldots) \vec{v}(x, t). \tag{11}$$

Taking the first 2 terms on the right hand side, diagonalizing A, and employing the spatial derivative approximation found in level 2 of the tree gives the CIR

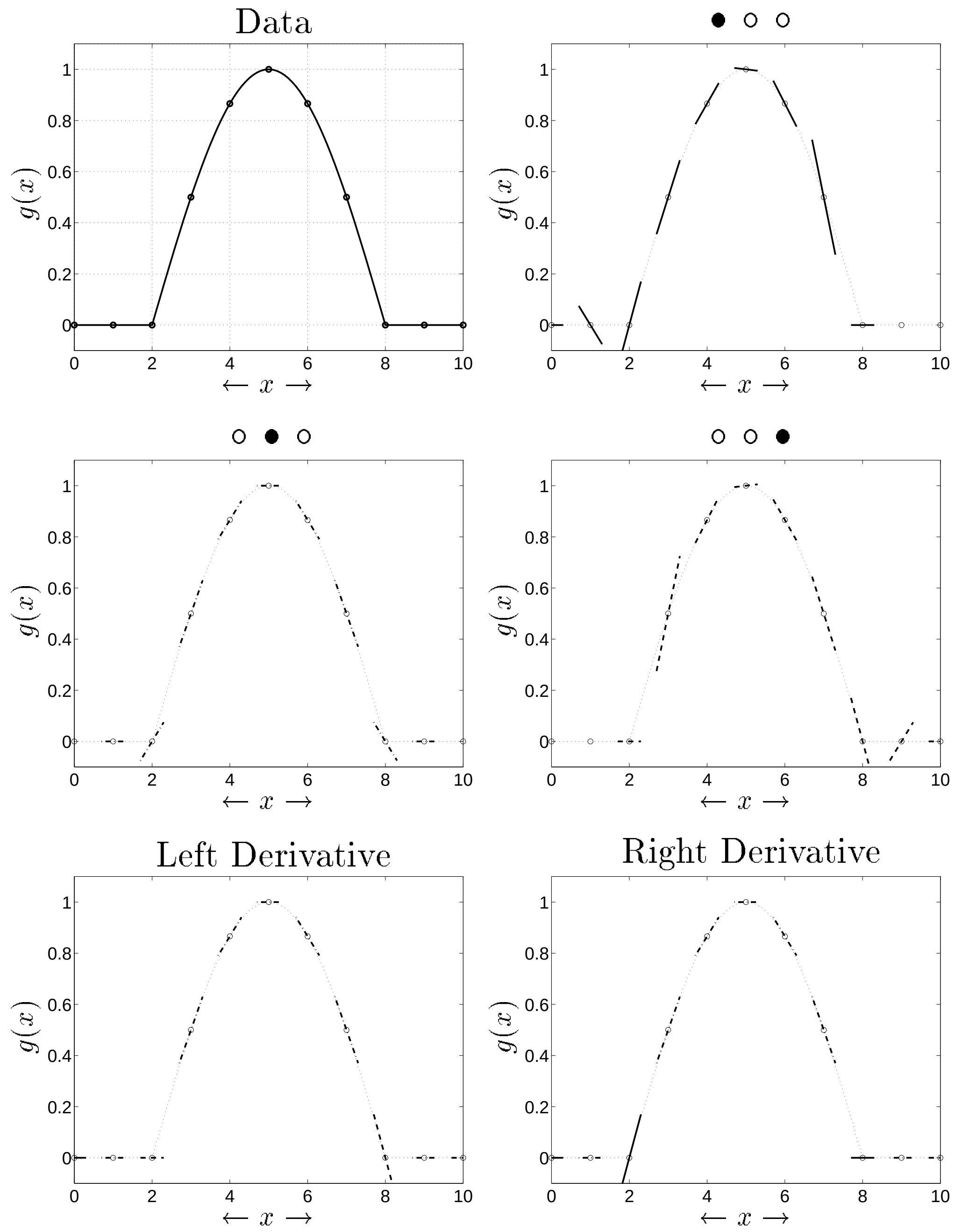

Fig. 2. Various Second Order Approximations To The Derivative

method. One can derive higher order methods simply by taking additional terms in the right hand side of (11) (increasing temporal accuracy), and by continuing to lower levels in the tree (increasing spatial accuracy).

2.5 Computational Complexity

Note that second order temporal accuracy requires computation of

$$(A\partial_x)^2\vec{v} = A\partial_x(A\partial_x\vec{v}),$$

and hence, an additional application of the approximation to $A\partial_x$. Similarly, n applications of $A\partial_x$ are required for n^{th} order temporal accuracy.

There are $n+1$, n^{th} order stencils that may be applied at any one point, each having an associated value $D^n(f)_i$. Although this appears to imply that $N(n+1)$ of the $D^n(f)_i$ must be computed in order to determine the appropriate stencil at N points, the trees (and therefore the elements $D^n(f)_i$) of neighboring points overlap. Thus, the one vector $D^n(f)$ contains all of the needed information for the decisions that need to be made on level n of the tree. Also, since the vectors $D^n(f)$ decrease in length as n increases, the *added* amount of work to go to the next level of the tree slightly decreases, even though the number of possible stencil choices increases.

Near a boundary, the tree in Figure 1 must be truncated. How these trees are truncated will affect the realized boundary conditions, and will be discussed in section 4.3.

3 Multi-Dimensional Problems

Clearly, the method described above can be applied to any first order linear hyperbolic system in one space dimension. Here we present a scheme that utilizes the splitting techniques of Strang [6] to extend the method to more than one space dimension.

3.1 System form of 2-D Scalar Wave Equation

To demonstrate this approach, consider the two dimensional acoustic wave equation

$$\partial_t^2 u - c^2(\partial_x^2 u + \partial_y^2 u) = 0.$$

This is equivalent to the first order system

$$\partial_t\vec{v} + A\partial_x\vec{v} + B\partial_y\vec{v} = \vec{0}, \tag{1}$$

where

$$\vec{v} = \begin{bmatrix} \partial_x u \\ \partial_y u \\ \partial_t u \end{bmatrix}, \quad A = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ -c^2 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -c^2 & 0 \end{bmatrix}.$$

Unfortunately, A and B do not commute and cannot be simultaneously diagonalized, so the numerical scheme described above must be modified.

3.2 Strang Splitting

Note that (11) can be expressed as

$$\vec{v}(x, t + \Delta t) = e^{-\Delta t A \partial_x} \vec{v}(x, t),$$

where

$$e^{-\Delta t A \partial_x} = I - \Delta t A \partial_x + \frac{\Delta t^2}{2} (A \partial_x)^2 - \ldots$$

defines the propagator for the operator $A\partial_x$. In the two dimensional case,

$$\vec{v}(x, y, t + \Delta t) = e^{-\Delta t (A \partial_x + B \partial_y)} \vec{v}(x, y, t).$$

Strang's idea [6] was to approximate the two dimensional propagators by a product of one dimensional propagators.

One can show

$$e^{-\Delta t (A \partial_x + B \partial_y)} = e^{-\Delta t A \partial_x} e^{-\Delta t B \partial_y} + \frac{\Delta t^2}{2} (AB - BA) + O(\Delta t^3), \tag{2}$$

so that a first order accurate approximation is based on

$$\vec{v}(x, y, t^{m+1}) = e^{-\Delta t A \partial_x} (e^{-\Delta t B \partial_y} \vec{v}(x, y, t^m)). \tag{3}$$

One can also derive a second order approximation, utilizing the fact that

$$e^{-\Delta t (A \partial_x + B \partial_y)} = e^{-\frac{1}{2} \Delta t A \partial_x} e^{-\Delta t B \partial_y} e^{-\frac{1}{2} \Delta t A \partial_x} + \mathcal{O}(\Delta t^3). \tag{4}$$

Applications of one dimensional propagators require numerical solutions to systems of the form

$$\partial_t \vec{v} + C \partial_s \vec{v} = 0, \tag{5}$$

where C is either A or B, corresponding to $s = x$ or $s = y$, respectively. $\vec{v}$ is a function of x, y, and t, but one of x or y is treated as a parameter. For example, to implement (3), one first propagates data $\vec{v}^m(x_i, y_j)$ from t^m to $t^m + \Delta t$ by numerically solving N_x linear systems

$$\partial_t \vec{v} + B \partial_y \vec{v} = 0,$$

one for each x_i. Call the resulting intermediate approximation $\vec{v}^{int} = \vec{v}^{int}(x_i, y_j)$. This is then taken as initial data, and one solves the N_y systems

$$\partial_t \vec{v} + A \partial_x \vec{v} = 0,$$

one for each y_j, to obtain $\vec{v}^{m+1}$.

4 Numerical Results in One Space Dimension

This section contains two examples that illustrate the attributes of the adaptive stencil method. We employed the approach described in subsection 2.4 to achieve second order temporal accuracy and, unless otherwise stated, an $O(\Delta x^2)$ adaptive spatial stencil (level 3 in the tree) is used, and Δt is taken to be $\Delta x / (2 \max_x c(x))$.

4.1 A one-dimensional problem with non-smooth initial data

The first example is the one dimensional initial value problem (IVP)

$$\begin{aligned} \partial_t^2 u - \partial_x^2 u = 0, -\infty < x < \infty,\ t > 0, \\ u(x,0) = f(x), \quad -\infty < x < \infty, \\ \partial_t u(x,0) = 0, \quad -\infty < x < \infty, \end{aligned} \tag{1}$$

where $f(x)$ is the piecewise linear function on the left in Figure 3. The piecewise constant derivative $f'(x)$ is shown on the right.

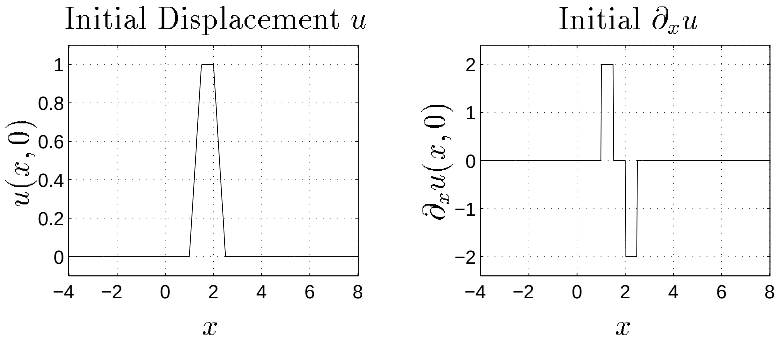

Fig. 3. Initial data a 1-D constant coefficient, homogeneous IVP.

Recall that our method approximates the vector of derivatives $\vec{v} = [\partial_x u, \partial_t u]^T$. The D'Lambert solution [5] of (1) is

$$u(x,t) = \frac{1}{2}f(x-t) + \frac{1}{2}f(x+t), \tag{2}$$

and so

$$\partial_x u(x,t) = \frac{1}{2}f'(x-t) + \frac{1}{2}f'(x+t), \tag{3}$$

i.e., $\partial_x u$ is the sum of translates of components of the piecewise constant function shown on the right in Figure 3.

In Figure 4, we present the numerical approximation of $\partial_x u$ at various times. In Figure 5 we present results obtained using approximations that are $O(\Delta x^n)$ for $n = 1, 2$ and 3. In this figure, we have zoomed in on the subinterval $3 \leq x \leq 6$ at time $t = 3$. Note that the approximations become increasingly sharp as n increases, and that none exhibit spurious oscillations.

We also applied a fixed stencil finite difference approximation to (1). The standard second order finite difference approximations to the derivatives,

$$\frac{u_j^{m+1} - 2u_j^m + u_j^{m-1}}{\Delta t^2} - \frac{u_{j+1}^m - 2u_j^m + u_{j-1}^m}{\Delta x^2} = 0, \tag{4}$$

was used. We took the same Δt and Δx as in the previous approximations to (1), and the CFL condition was met. On the left in Figure 6 is the resulting approximation of u at time $t = 3$, and on the right is the finite difference approximation to the derivative $\partial_x u(x,3)$. Note the spread in the waveform and introduction of spurious oscillations in the approximation of u. Taking the derivative $\partial_x u$ greatly magnifies these oscillations.

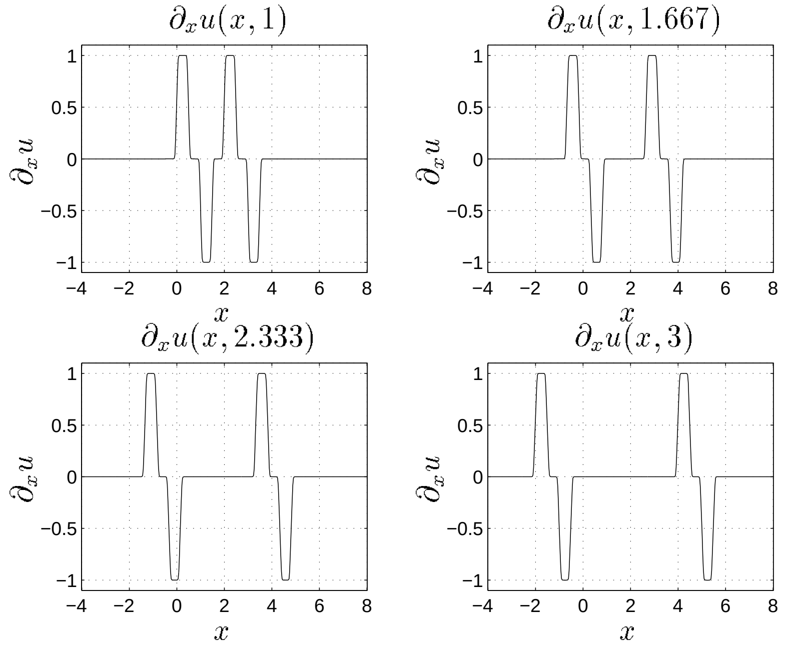

Fig. 4. Adaptive stencil solutions for a 1-D, constant coefficient, homogeneous, scalar wave IVP at times $t = 1, \frac{5}{3}, \frac{7}{3}$, and 3.

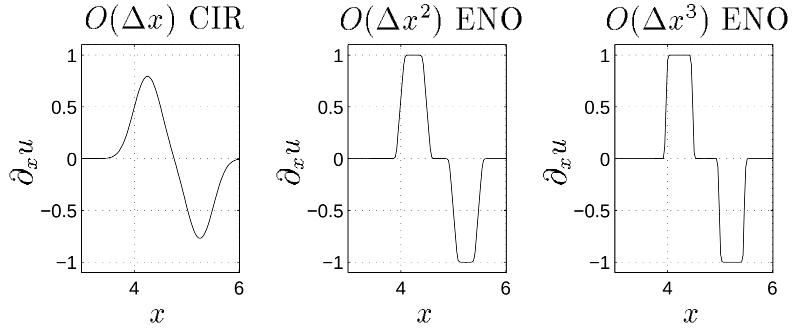

Fig. 5. Effects of degree on adaptive stencil solutions.

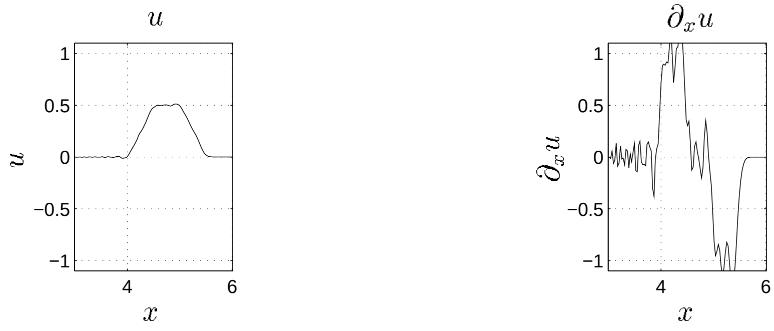

Fig. 6. Standard finite difference solution to a 1-D, constant coefficient, homogeneous, scalar wave IVP.

4.2 A one-dimensional discontinuous coefficient problem

The following test case was taken from [2]. Consider the IVP

$$\begin{aligned} \partial_t^2 u - c(x)^2 \partial_x^2 u = f(t)\delta(x - x_0), -\infty < x < \infty,\ t > 0, \\ u(x, 0) = \partial_t u(x, 0) = 0, \qquad -\infty < x < \infty, \end{aligned} \tag{5}$$

with the wave speed $c(x)$, the forcing function $f(t)$, and the source location x_0 given in Figure 7 below. Note that $c(x)$ is piecewise constant with a jump discontinuity at position $x = 5203$ meters, where the wave speed doubles. Figure 8 shows our adaptive stencil results. For x beyond the jump discontinuity, $\partial_t u$ is simply a rescaled translate of $f(t)$. The temporal rescaling factor is $1/2$, while the amplitude changes by a factor of $4/3$. In [2] Dablain advocated high order (up to 10^{th} order in space and 4^{th} in time) difference approximation schemes to accurately track u. This requires stencils with many points. Associated with these large stencils are difficulties in handling boundary conditions, large storage requirements, and the need for a separate temporal "start-up" procedure. Our second order adaptive stencil approach is relatively easy to implement, and gives results which compare very favorably with those in [2].

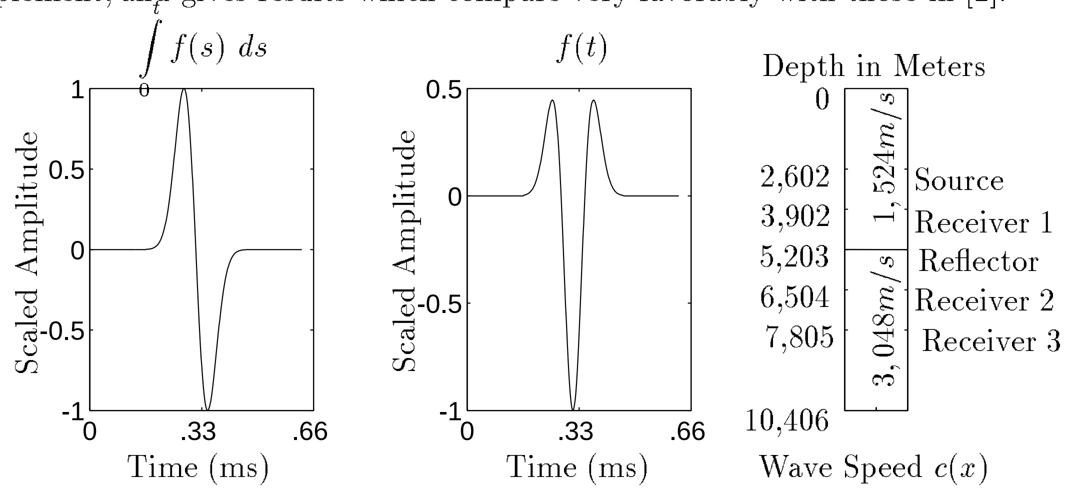

Fig. 7. The forcing function $f(t)$ and discontinuous wave speed $c(x)$.

4.3 Boundary Conditions

The adaptive stencil selection procedure outlined above cannot be applied near the boundary of the computational domain. Here we discuss modifications required to accommodate various boundary conditions.

The free surface boundary condition, $\frac{\partial u}{\partial n} = 0$, is easily handled since the components of ∇u (rather than u itself) are computed at each time step. For

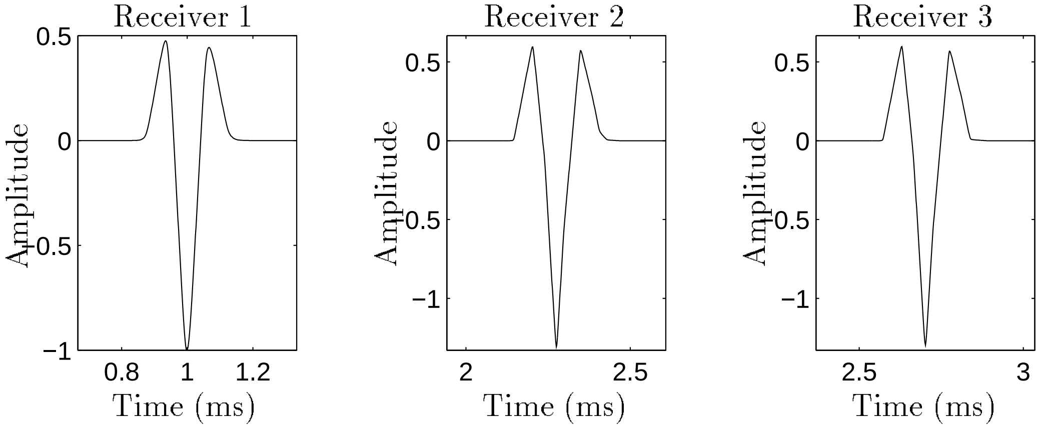

Fig. 8. Time traces at various spatial locations obtained using a second order adaptive stencil method.

instance, in the 1-D case, we can *set* $\partial_x u$ equal to zero at the end points at each time step. Standard methods (see (4)) require additional, often quite complicated, difference approximations near the endpoints.

Absorbing boundary conditions allow a wave to pass through a computational boundary with no reflections. Assuming a small homogeneous region near the boundary (of the one dimensional problem), the system (4) decouples into right- and left-moving waves that are easily absorbed by using stencils that *look inward* at the computational boundaries. Suppose we wish to approximate the derivative of $\vec{f} = (f_1, \ldots, f_N)$ with the procedure outlined above. We can force the selection of these inward looking stencils near the boundary by setting $f_0 = f_{N+1} = \infty$, or some relatively large value. The selection scheme described by the tree in Figure 1 will *not* select these new end nodes at any level, resulting in absorption.

5 Numerical Results in Two Space Dimensions

In the two examples below, we employed second order accurate Strang splitting (4) together with a second order adaptive spatial stencil. In both examples, absorbing boundary conditions are employed, $\Delta x = \Delta z$, and Δt is taken to be $\Delta x/(2 \max_{(x,z)} c(x,z))$.

5.1 A two-dimensional discontinuous coefficient problem

Consider

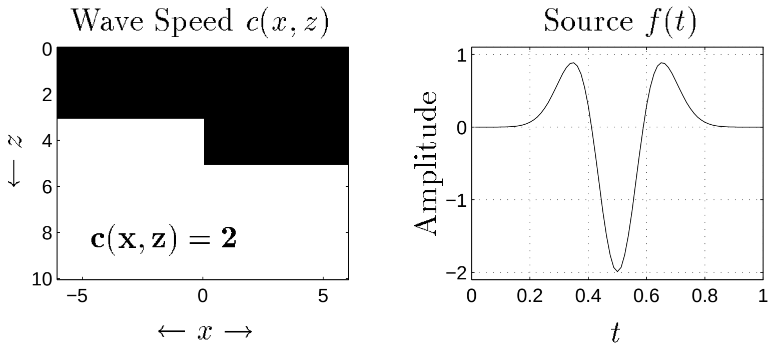

Fig. 9. Wave Speed and Forcing Function for a 2-D discontinuous coefficient problem.

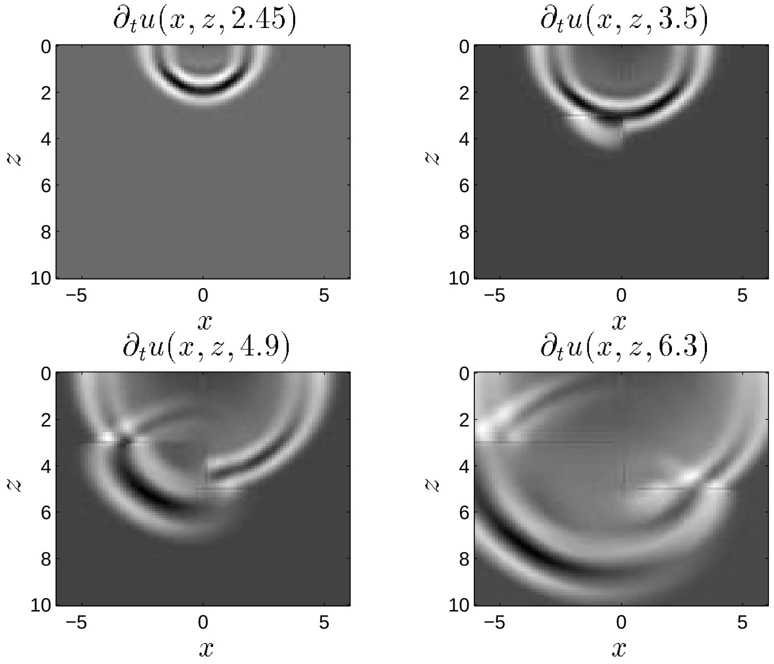

Fig. 10. Adaptive stencil approximations to $\partial_t u$ at Various Times t for a 2-D discontinuous coefficient problem.

$$\begin{aligned}\partial_t^2 u - c^2\left(\partial_x^2 u + \partial_z^2 u\right) &= f(t)\delta(x)\delta(z), \\ u = \partial_t u &= 0 \,\text{at}\, t = 0,\end{aligned} \tag{1}$$

where $c(x,z)$ is the piecewise constant function shown on the left in Figure 9 and $f(t)$ is the Ricker wavelet commonly used in geophysical problems [2], shown on the right in Figure 9.

Note the term $\partial_x c$ in equation (4). When c has jump discontinuities, $\partial_x c$ must be interpreted in a distributional sense. We use standard difference approximations to the required distributional derivatives $\partial_z c$ and $\partial_x c$. The support of the distributional derivative $\partial_z c$ consists of the horizontal interfaces and is the short vertical interface for $\partial_x c$. Numerical results are shown in Figure 10.

In the first frame the wave propagates from a point source at the top center of

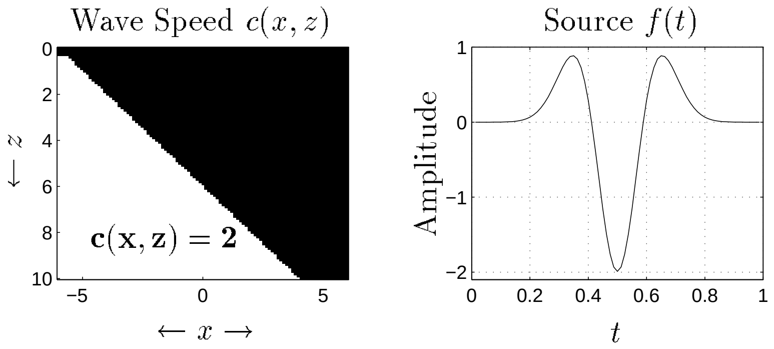

Fig. 11. Wave speed and forcing function for the Slanted Interface Problem.

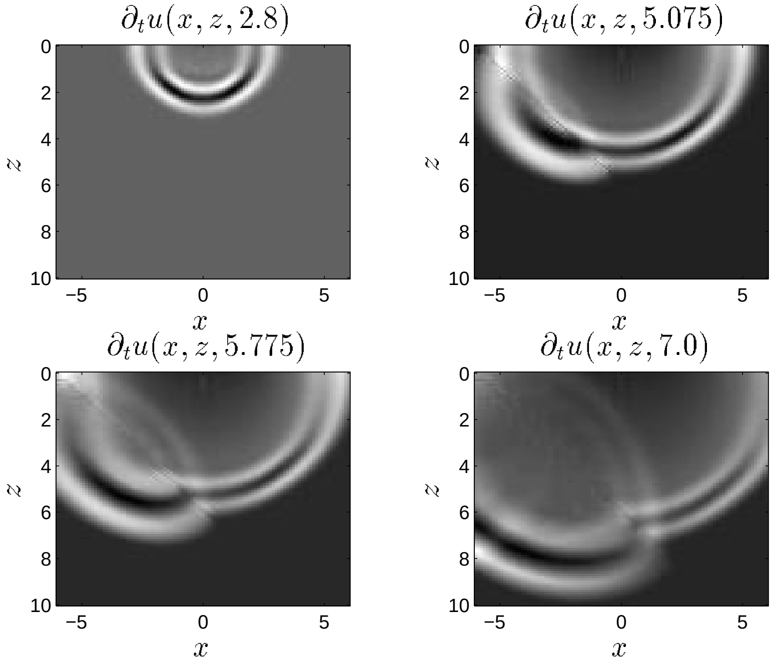

Fig. 12. Adaptive stencil approximations to $\partial_t u$ at various times t for the Slanted Interface Problem.

the media. In the second frame we see the wave soon after it reaches the left interface. Note both the returning reflection and the higher speed transmitted wave. In the third frame, the main wave meets the right horizontal interface, causing a reflection which can be seen in the fourth and final frame. The effect of the absorbing boundary conditions can also be seen in this final frame.

5.2 A two-dimensional problem with a slanted interface

Again consider (1) with the same forcing function and initial and boundary conditions as above, but with a different discontinuous wave speed, given in Figure 11.

In the first frame a wave propagates from a point source at the top center of the media. In the second frame we see the wave soon after it reaches the slanted interface. In the third frame, part of the transmitted wave reaches the computational boundary and is absorbed, and the reflected wave becomes apparent. In the final frame, both the transmitted and the reflected wave are being absorbed.

In this test case, there seem to be no difficulties associated with discontinuities that are not oriented along the computational grid.

References

[1] R. Courant, E. Isaacson, and M. Rees, On the Solution of Nonlinear Hyperbolic Differential Equations by Finite Differences, Comm. Pure Appl. Math., vol. 5 (1952), pp. 243-255.

[2] M. A. Dablain, *The application of high-order differencing to the scalar wave equation*, Geophysics, vol. 51 (1985), pp. 54-66.

[3] A. Hall and T. A. Porsching, *Numerical Analysis of Partial Differential Equations*, Prentice-Hall, 1990.

[4] A. Harten and S. Osher, Uniformly high-order accurate nonoscillatory schemes I, SIAM J. Numer. Anal., vol. 24 (1987), pp. 279-309.

[5] F. John, *Partial Differential Equations, 4th Edition*, Springer-Verlag, 1982.

[6] G. Strang, *On the construction and comparison of difference schemes*, SIAM J. Numer. Anal., vol. 4 (1968), pp. 506-517.

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

[8] P. M. Prenter, *Splines and Variational Methods*, Wiley, 1975.

[9] R. Vichnevetsky and J. B. Bowles, Fourier analysis of Numerical approximations of Hyperbolic Equations, SIAM, 1982.