



An adaptive stencil finite difference method for first order linear hyperbolic systems
by Robert Henry Hoar

A thesis submitted in partial fulfillment of the requirements for the degree of Doctor of Philosophy
Mathematics

Montana State University

© Copyright by Robert Henry Hoar (1995)

Abstract:

A numerical method is presented to deal with the difficulties associated with discontinuous coefficients and initial data for 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 splittings. Several examples in one and two space dimensions are presented which indicate that the method maintains sharp waveforms without introducing spurious oscillations.

AN ADAPTIVE STENCIL FINITE DIFFERENCE METHOD
FOR FIRST ORDER LINEAR HYPERBOLIC SYSTEMS

by

ROBERT HENRY HOAR

A thesis submitted in partial fulfillment
of the requirements for the degree

of

Doctor of Philosophy

in

Mathematics

MONTANA STATE UNIVERSITY
Bozeman, Montana

April 1995

D378
H651

APPROVAL

of a thesis submitted by

ROBERT HENRY HOAR

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the College of Graduate Studies.

April 19, 1995
Date

Curtis R. Vogel
Curtis R. Vogel
Chairperson, Graduate Committee

Approved for the Major Department

April 19, 1995
Date

John Lund
John Lund
Head, Mathematics

Approved for the College of Graduate Studies

5/1/95
Date

Robert Brown
Robert Brown
Graduate Dean

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment for a doctoral degree at Montana State University, I agree that the Library shall make it available to borrowers under rules of the Library. I further agree that copying of this thesis is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U. S. Copyright Law. Requests for extensive copying or reproduction of this thesis should be referred to University Microfilms International, 300 North Zeeb Road, Ann Arbor, Michigan 48106, to whom I have granted "the exclusive right to reproduce and distribute copies of the dissertation for sale in and from microform or electronic format, along with the right to reproduce and distribute my abstract in any format in whole or in part."

Signature Robert H. Hoar

Date April 24, 1995

ACKNOWLEDGEMENTS

I would like to thank the following people without whose help and support this thesis would never have been possible.

My committee members,

Dr. Curtis R. Vogel,

Dr. Jack Dockery,

Dr. Gary Bogar,

Dr. John Lund,

Dr. Mark Pernarowski,

and my wife,

Kerrie L. Hoar.

I would like to dedicate this to my daughter,

Emily Rose Hoar.

I am rediscovering the world through her eyes.

TABLE OF CONTENTS

	Page
LIST OF TABLES	vii
LIST OF FIGURES	viii
ABSTRACT	ix
1. Introduction	1
2. Wave Equations	7
A Derivation of the Elastic Wave Equation	7
Reduced Forms of the Elastic Wave Equation	10
Solution of the One-Dimensional Acoustic Wave Equation	12
Solutions for a Family of Two-Dimensional Problems	12
3. First Order Linear Hyperbolic Systems	15
First Order Linear Hyperbolic Systems	15
First Order System Form for Some Wave Equations	16
Diagonalization in the Acoustic Case	18
Diagonalization in the Elastic Case	21
4. Dimensional Splitting Methods	23
Propagators and Dimensional Splitting	23
Derivation and Extensions of Strang Splittings	25
5. Adaptive Stencil Finite Difference Approximations	29
Taylor's Theorem	30
Finite Difference Approximations	30
A Motivating Example	32
The CIR Method	37
Higher Order Adaptive Stencil Selection	38
Essentially Non-Oscillatory Schemes	40

Increasing Temporal Accuracy	43
The CFL Condition	44
Propagation of Information about a Point	47
6. Implementation Issues	49
An Algorithm for First Order Systems	49
Numerical Solutions of the Acoustic Wave Equation	53
Boundary Conditions	55
Efficient Stencil Selection	57
Computational Complexity	60
7. Numerical Results	62
First Order, One Space Dimension	62
Systems in One Space Dimension	70
Numerical Results in Two Space Dimensions	74
REFERENCES CITED	79

LIST OF TABLES

Table		Page
1	Matrices arising in the diagonalization of the first order system operator (3.14)	20
2	The matrices $D(\partial_s E^{-1})E$ in (3.15)	20
3	Matrices arising in the diagonalization of systems in (3.17)	22
4	Necessary constants in $\mathcal{O}(\Delta x^n)$ approximations to the first derivative.	33

LIST OF FIGURES

Figure		Page
1	Various second order approximations to the derivative.	35
2	Tree of possible finite difference stencils.	39
3	Illustration of the computational domain of dependence.	45
4	Initial data and (exact) numerical approximation at time $t = 3$	63
5	Non-zero entries of the matrix M^m for a certain m	65
6	Non-zero entries of \hat{P}	65
7	Initial and final displacement for an $O(\Delta x^2)$ method.	66
8	Random noise.	67
9	Initial data (including noise) and numerical solution at time $t = 3$	68
10	Initial data and 3-point fixed stencil solution at time $t = .5$	68
11	Initial data and 5-point fixed stencil solution at time $t = .5$	69
12	Initial and final displacement for an $O(\Delta x^3)$ method.	69
13	Initial data for a one-dimensional, homogeneous IVP.	70
14	Adaptive stencil solutions for a one-dimensional, homogeneous, scalar wave IVP at times $t = 1, \frac{5}{3}, \frac{7}{3}$, and 3.	71
15	Effects of stencil size on adaptive stencil solutions.	72
16	Standard finite difference solution to a one-dimensional, constant coefficient, homogeneous, scalar wave equation.	72
17	The forcing function, $f(t)$, and discontinuous wave speed, $\hat{c}(x)$	73
18	Time traces at various spatial locations obtained using a second order adaptive stencil method.	74
19	Initial data for a two-dimensional problem.	75
20	Energy norm of the error versus spatial step size.	75
21	Wave speed and forcing function for a two-dimensional discontinuous coefficient problem.	76
22	Adaptive stencil approximations to $\partial_t u$ at various times, t , for a two-dimensional discontinuous coefficient problem.	77
23	Wave speed and forcing function for a two-dimensional problem with a slanted interface.	77
24	Adaptive stencil approximations to $\partial_t u$ at various times, t , for the slanted interface problem.	78

ABSTRACT

A numerical method is presented to deal with the difficulties associated with discontinuous coefficients and initial data for 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 splittings. Several examples in one and two space dimensions are presented which indicate that the method maintains sharp waveforms without introducing spurious oscillations.

CHAPTER 1

Introduction

The original purpose of this thesis project was to develop a method to efficiently compute highly accurate approximate solutions to the elastic wave equation. The elastic wave equation is an equation that models the propagation of disturbances through certain materials. In the Geosciences, for example, the propagation of stress and shear waves through the earth may be modeled by the elastic wave equation. The method developed in this thesis has been successfully applied to a related equation known as the scalar, or acoustic, wave equation. This equation has important applications in sound and fluid propagation problems.

Except for very restrictive special cases, exact solutions to these problems are not available. Due to the lack of exact solutions, techniques have been developed to *approximate* the solution of certain wave propagation problems. The techniques fall into two general classes—asymptotic methods and numerical methods. Asymptotic methods (high frequency) are appropriate in situations in which sharp (δ) wave pulses are propagated and coefficients in material parameters are discontinuous. Numerical methods (both finite difference and finite element methods) are effective when the initial and boundary data are smooth and when the parameters are smooth.

Asymptotic techniques may be successfully applied when the solution contains discontinuities [22], [23], [2]. When the solution is not smooth, the leading-order term(s) in an asymptotic expansion of the solution will capture much of the information needed to resolve the solution. However, these methods are difficult to implement if the geometry of the discontinuities becomes too complex. Also, if the solution is

smooth, the leading terms in the asymptotic expansion may not capture enough information to accurately represent the solution.

The use of standard finite difference and finite element methods is justified in situations in which the material parameters and initial data are smooth [10], [17]. These methods do not perform well if smoothness assumptions are violated [7]. Even for equations with smoothly varying coefficients and initial data, standard finite difference methods can suffer from numerical dispersion, i.e., the propagation of differing frequency components at different speeds [21]. This is manifested in the spread of waveforms and the introduction of spurious oscillations [7]. Finite element methods also exhibit undesirable dispersive properties [1].

Attempts have been made to overcome the limitations of standard numerical methods. One such attempt resulted in a class of finite difference schemes known as local mesh refinement methods. In the areas of the computational domain where it is determined that large errors may be present, the computation mesh spacing is reduced. These methods work well when the solution is expected to be rapidly varying in isolated areas. If the initial data is non-smooth, or if the material parameters are too complex, then the 'local' refinements become global ones.

Other innovative methods based on finite differences have been developed. These include a method currently under development by LeVeque and Zhang [24] which is applicable to the two-dimensional acoustic wave equation. The method seeks to eliminate errors due to discontinuities in the wave speed. Near the interface of regions of different wave speeds, node weighting coefficients are determined so that the transmitted and reflected waves will have the correct amplitude and form. Away from these interfaces, standard finite difference techniques are applied. Since standard finite difference methods are utilized away from the interfaces, the solution must be smooth in these areas, implying that the initial data must be smooth. This method

makes explicit use of the location of the discontinuities, and although the development of this method has not been completed, it appears that it will be difficult to implement if the geometry of the discontinuities becomes too complex.

When a finite difference method fails to yield accurate approximations due to smoothness requirements being violated, the resulting errors are often manifested in the spread of waveforms and the introduction of spurious oscillations (see [7, 21]). A function containing many oscillations has a large *total variation*. Time stepping methods for scalar conservation laws have been developed [17] which require that the total variation of approximate solutions be non-increasing from time step to time step. These methods are called total variation diminishing (TVD) methods. Unfortunately, these methods can achieve only a limited order of accuracy for most problems [17].

The essentially non-oscillatory (ENO) methods of Harten and Osher [12] are also based on the total variation of the approximation. An ENO method allows the total variation to grow, but limits the rate at which it may grow [11]. The ENO based methods have been very successfully applied to first order scalar conservation laws [13]. The method uses a new interpolation technique that renders an essentially non-oscillatory scheme. The ENO scheme has been adapted to scalar Hamilton-Jacobi equations [19]. This formulation of the ENO scheme was used to study fronts propagating with curvature dependent speed [18].

To date, the ENO schemes have only been applied directly to first order scalar equations [8]. The purpose of this dissertation is to develop a numerical method that is based on the ENO scheme but is applicable to systems of first order equations. Each of the second order model problems considered can be put into this first order system form. The method uses standard finite difference approximations at each point in space, but the form of the stencil is allowed to vary from point to point, and the ENO polynomial selection scheme is used to select the finite difference stencil.

In the development of the adaptive stencil numerical method, a number of existing techniques were incorporated. The ENO polynomial interpolation scheme of Harten and Osher has been modified to select finite difference stencils to approximate first derivatives. A diagonalization process is used to essentially decouple one-dimensional systems. The Courant-Isaacson-Rees (CIR) method [6] is then applied to the resulting system. The CIR method is then extended to obtain higher orders of accuracy. For multi-dimensional problems, the Strang splitting [20] is used to reduce the problem into a coupled set of one-dimensional problems. 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.

This thesis is comprised of seven chapters, the first of which is this introduction. The remaining chapters examine important aspects of the research and contain the results of the research.

Chapter 2 presents the elastic wave equation and a number of related equations. A summary of solutions to wave propagation problems is included. These *model problems* and their solutions are used to test the numerical methods.

In Chapter 3, the various model problems are put into first order system form. For problems in one space dimension, the resulting first order system may be *diagonalized*—a process by which the system is essentially decoupled. This diagonalization is similar to the *factorization* (see [15]) of a second order problem into left and right moving waves. The high resolution numerical method is directly applicable

to systems in this decoupled form. For multi-dimensional problems, the diagonalization process does not apply. Utilizing a dimensional splitting method (outlined in Chapter 4), multi-dimensional problems may be approximated by a series of coupled one-dimensional problems. The diagonalization of the one-dimensional problems resulting from the dimensional splitting of the model problems is also outlined.

In Chapter 4, the Strang splitting technique, a technique used to *split* multi-dimensional problems into a coupled set of one-dimensional problems, is examined. This splitting technique is not exact. The order (accuracy) of the splittings, as well as extensions to higher orders, is presented.

In Chapter 5, aspects of finite difference approximations to derivatives are examined. This chapter contains a statement of Taylor's theorem, as well as a number of approximations that may be derived from it. The need for an adaptive stencil procedure is motivated by a simple example, and a procedure for adapting the stencil is presented. The first order accurate method of Courant, Issacson and Rees (CIR) is applicable to linear first order hyperbolic systems in one space dimension. The CIR method is examined, and an extension to higher orders of spatial accuracy utilizing the adaptive stencil technique is given. A method by which the temporal accuracy may be increased is discussed. Finally, stability concerns, including a CFL condition and the approximation of derivatives near discontinuities, are discussed.

Chapter 6 describes key aspects of the numerical implementation of the presented method. The method is a single step method, implying that no special considerations must be made in order to start the time stepping process, and implying that very little data must be saved. The adaptive stencil selection scheme, though non-linear, may be vectorized. The structure of the model problems may be utilized to increase the efficiency of the numerical approximation procedure. Also contained in this chapter are the necessary steps needed to numerically implement various bound-

ary conditions.

Chapter 7 contains numerical results. An in-depth study of the one-dimensional scalar advection equation is conducted. The numerical results for problems with discontinuous initial data are used to examine properties of the adaptive stencil procedure. Also included are numerical results for a one-dimensional initial value problem for the scalar wave equation with non-smooth initial data. An example with a discontinuous coefficient (wave speed) is presented. This example was taken from [7], allowing a comparison of the adaptive stencil scheme with existing finite difference methods. A problem from the class of two-dimensional initial value problems (with solutions) that is presented in Chapter 2 is considered. This example provides numerical evidence that the Strang splitting technique (presented in Chapter 4) converges. Also, results for a pair of two-dimensional scalar wave equations, each having discontinuous wave speeds, are presented, and the qualitative behavior of the results are examined. In one of the two-dimensional examples, the discontinuity is not aligned with the computational grid.

CHAPTER 2

Wave Equations

In the first section of this chapter, the elastic wave equation in \mathbb{R}^3 and the related terminology and notation are presented. The second section deals with simplifications of the elastic wave equation that occur in reductions to one and two space dimensions, as well as the related acoustic wave equation. This chapter also presents some of the known solutions to special cases of these equations. These solutions will be used later to develop and test numerical methods.

A Derivation of the Elastic Wave Equation

In this section, the equation known as the elastic wave equation is derived. This equation describes the motion of a material subject to what is known as an elastic deformation. A material that has the ability to return to its original shape after being deformed or stressed (within limits) is said to be an *elastic material*. If the material is deformed and is able to return to its original shape, then the deformation is known as an *elastic deformation*.

The *stress* acting on a material is the force per unit area normal to the direction of the applied force. The symbol σ_{ij} , ($i, j \in \{x, y, z\}$) refers to the stress where i denotes the direction of the force and j denotes the normal direction to the plane on which the force acts. The position of a point, $\mathbf{x} = (x, y, z)$, will vary with time as the deformation is applied and the material returns to its original shape. The displacement of each of the coordinates is recorded in the vector-valued function, $\mathbf{u} = [u_i, u_j, u_k]^T$, where u_i, u_j, u_k are components of \mathbf{u} in directions x, y , and z respectively.

The position of the point (x, y, z) at time t is given by $(x+u_i(x, y, z, t), y+u_j(x, y, z, t), z+u_k(x, y, z, t))$.

Although many materials have parameters that vary with time t , here it is assumed that the material parameters under consideration are functions of spatial location, \mathbf{x} , only. The net forces, in terms of stress, acting in the x , y and z directions on a cube of size Δx by Δy by Δz centered at (x, y, z) are

$$(\partial_x \sigma_{xx} + \partial_y \sigma_{xy} + \partial_z \sigma_{xz}) \Delta x \Delta y \Delta z$$

$$(\partial_x \sigma_{yx} + \partial_y \sigma_{yy} + \partial_z \sigma_{yz}) \Delta x \Delta y \Delta z$$

$$(\partial_x \sigma_{zx} + \partial_y \sigma_{zy} + \partial_z \sigma_{zz}) \Delta x \Delta y \Delta z$$

respectively.

Newton's second law implies that the force is equal to the mass, m , of the $\Delta x \Delta y \Delta z$ cube times the acceleration. The acceleration is the second (partial) time derivative of the corresponding displacement, so that

$$m \partial_t^2 u_i = (\partial_x \sigma_{ix} + \partial_y \sigma_{iy} + \partial_z \sigma_{iz}) \Delta x \Delta y \Delta z, \quad i \in \{x, y, z\}. \quad (2.1)$$

The relationship between stress and displacement is given by Hooke's law (see [16]),

$$\sigma_{ij} = \delta_{ij} \lambda (\nabla \cdot \mathbf{u}) + 2\mu e_{ij},$$

where δ_{ij} is the Kronecker delta and the strain tensor is given by

$$e_{ij} = \frac{1}{2} (\partial_j u_i + \partial_i u_j).$$

The parameters λ and μ , known as the Lamé parameters, are used to describe the elastic properties of materials. Another relevant quantity is the density, ρ , of the material. A material or medium that has constant density and constant Lamé parameters is said to be *homogeneous*. A material or medium that has non-constant

Lamé parameters is said to be *heterogeneous*. Letting

$$\rho = \frac{m}{\Delta x \Delta y \Delta z}$$

and taking the limit as Δx , Δy , and $\Delta z \rightarrow 0$, (2.1) becomes

$$\rho \partial_t^2 u_i = \partial_j [\delta_{ij} \lambda \partial_k u_k + \mu (\partial_i u_j + \partial_j u_i)], \quad (2.2)$$

which is referred to as the *elastic wave equation* [16]. The subscripts run from 1 to 3 and repeated indices are summed from 1 to 3 (summation notation). It is often convenient to express (2.2) in equivalent forms. For example, utilizing dot and cross product notation, the elastic wave equation may be expressed as

$$\rho \partial_t^2 \mathbf{u} = \nabla((\lambda + 2\mu) \nabla \cdot \mathbf{u}) - \nabla \times \mu (\nabla \times \mathbf{u}). \quad (2.3)$$

Alternative forms of the elastic wave equation will be presented where needed.

Finally, a number of specific initial boundary value problems for the elastic wave equation can be formulated. The general initial value problem for the elastic wave equation in \mathbb{R}^3 is defined as follows: Given $\lambda(\mathbf{x})$, $\mu(\mathbf{x})$, $\mathbf{g}(\mathbf{x})$, $\mathbf{h}(\mathbf{x})$, and $\mathbf{f}(\mathbf{x}, t)$, find the vector \mathbf{u} satisfying

$$\rho \partial_t^2 u_i = \partial_j [\delta_{ij} \lambda \partial_k u_k + \mu (\partial_i u_j + \partial_j u_i)] + f_i(\mathbf{x}, t), \quad \mathbf{x} \in \mathbb{R}^3, t > 0 \quad (2.4)$$

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{g}(\mathbf{x}) \quad \mathbf{x} \in \mathbb{R}^3$$

$$\partial_t \mathbf{u}(\mathbf{x}, 0) = \mathbf{h}(\mathbf{x}) \quad \mathbf{x} \in \mathbb{R}^3$$

Here the f_i are external body forces, \mathbf{g} contains the initial displacements, and \mathbf{h} contains the initial velocities.

An important problem in the Geosciences is *Lamb's half space problem*. This is the general elastic wave equation restricted to the lower half space $\{(x, y, z) \in \mathbb{R}^3 : z < 0\}$ together with boundary conditions that are imposed at $z = 0$. It is assumed

that the stresses at the surface ($z = 0$) must be equal to the force per unit area acting on the surface.

An important special case of Lamb's half space problem is the *free surface half space problem*. If there are no forces acting on the surface ($z = 0$), the surface is called a *free surface*. This is modeled by setting the stress components σ_{xz} , σ_{yz} , and σ_{zz} equal to zero along this boundary.

Reduced Forms of the Elastic Wave Equation

It is convenient to consider the reductions to two and one space dimensions. These reduced equations model various physical situations. These reduced equations are also useful in the testing of numerical methods.

For a two-dimensional problem, there is assumed to be no motion in the second, or y dimension¹. Then $\mathbf{u} = [u_i, u_k]^T$,

$$\nabla(\nabla \cdot \mathbf{u}) = \begin{bmatrix} \partial_x(\partial_x u_i + \partial_z u_k) \\ \partial_z(\partial_x u_i + \partial_z u_k) \end{bmatrix}, \quad \text{and}$$

$$\nabla \times \nabla \times \mathbf{u} = \begin{bmatrix} \partial_x \partial_z u_k - \partial_z^2 u_i \\ -\partial_x^2 u_k + \partial_x \partial_z u_k \end{bmatrix}.$$

Under the additional assumption that the Lamé parameters are constant, the elastic wave equation (2.2) reduces to the system

$$\partial_t^2 u_i - c_1^2 \partial_x^2 u_i - c_2^2 \partial_z^2 u_i = (c_1^2 - c_2^2) \partial_x \partial_z u_k \quad (2.5)$$

$$\partial_t^2 u_k - c_2^2 \partial_x^2 u_k - c_1^2 \partial_z^2 u_k = (c_1^2 - c_2^2) \partial_x \partial_z u_i.$$

In this system, c_1 and c_2 are defined to be the longitudinal and the shear wave speeds, respectively, given by

$$c_2 = \sqrt{\frac{\mu}{\rho}} \quad \text{and} \quad c_1 = \sqrt{\frac{\lambda + 2\mu}{\rho}}.$$

¹This follows conventions in the Geosciences, where the z axis is normal to the surface (plane) of the earth.

The free surface half plane problem in two dimensions requires boundary conditions at $z = 0$. The appropriate boundary condition is

$$\begin{pmatrix} \mu(\partial_z u_i + \partial_x u_k) \\ (\lambda + 2\mu)\partial_z u_k + \lambda\partial_x u_i \end{pmatrix} (x, 0, t) = \mathbf{0}.$$

In one space dimension, $\mathbf{u} = [u_k] = u$,

$$\nabla(\nabla \cdot \mathbf{u}) = \partial_z^2 u_k = \partial_z^2 u,$$

and

$$\nabla \times \nabla \times \mathbf{u} = \mathbf{0} = 0.$$

This implies that with λ and μ constant and $c = \sqrt{\frac{\lambda+2\mu}{\rho}}$, the elastic wave equation becomes

$$\partial_t^2 u - c^2 \partial_z^2 u = 0,$$

which is the one-dimensional version of the acoustic, or scalar, wave equation.

The general form of the acoustic, or scalar, wave equation is

$$\partial_t^2 u = c^2 \nabla \cdot (\nabla u). \quad (2.6)$$

The displacement, u , is scalar valued and c may, in general, depend on \mathbf{x} . This equation also models the propagation of a disturbance through a fluid at speed c . A simple three-dimensional example is sound traveling through the air.

In the next two sections, exact solutions are presented for certain one- and two-dimensional acoustic wave equations. The solution to the acoustic wave equation in one dimension, with constant wave speed and appropriate initial conditions, is easily obtained analytically, and is, therefore, an excellent test case for numerical methods. In two dimensions, it is impossible, in general, to derive exact solutions, even when the wave speed is constant. However, solutions may be derived in certain special cases involving radial symmetry.

Solution of the One-Dimensional Acoustic Wave Equation

The initial value problem for the constant coefficient acoustic wave equation in one dimension is

$$\begin{aligned} u_{tt} - c^2 u_{xx} &= h(x, t), & -\infty < x < \infty, & t > 0, \\ u(x, 0) &= f(x), & u_t(x, 0) &= g(x), & -\infty < x < \infty. \end{aligned}$$

The solution to this problem is known as D'Lambert's solution,

$$u(x, t) = \frac{1}{2}f(x - ct) + \frac{1}{2}f(x + ct) + \frac{1}{2} \int_{x-ct}^{x+ct} g(u) du + \int_0^t \int_{x-c(t-\tau)}^{x+c(t-\tau)} h(u, \tau) du d\tau.$$

Under certain assumptions on f , g , and h , the derivation of this solution is well known (see [14]).

Solutions for a Family of Two-Dimensional Problems

This section contains a formula for exact solutions to the two-dimensional constant coefficient acoustic wave equation with a family of initial conditions. If the domain is all of \mathbb{R}^2 , and it is assumed that the initial data is radially symmetric, then a change of variables will reduce the two-dimensional constant coefficient equation to a one-dimensional variable coefficient equation.

Assume radial symmetry, so that $u(x, z, t) = u(r, t)$, where $r = \sqrt{x^2 + z^2}$. Note that

$$\partial_x u = \partial_r u \partial_x r, \quad \partial_x^2 u = \partial_r^2 u (\partial_x r)^2 + \partial_r u \partial_x^2 r.$$

The scalar wave equation in 2 variables

$$\partial_t^2 u + c^2 (\partial_x^2 u + \partial_z^2 u) = 0$$

becomes

$$\partial_t^2 u + c^2 \left(\partial_r^2 u + \frac{1}{r} \partial_r u \right) = 0. \tag{2.7}$$

Since the initial conditions entail radial symmetry,

$$u(r, 0) = f(r); \quad \partial_t u(r, 0) = g(r), \quad r > 0. \quad (2.8)$$

To solve (2.7)-(2.8), apply the (zero-order) Hankel transform [23] to obtain

$$\partial_t^2 U + c^2 \lambda^2 U = 0, \quad t > 0, \quad (2.9)$$

$$U(\lambda, 0) = F(\lambda); \quad \partial_t U(\lambda, 0) = G(\lambda), \quad (2.10)$$

where U , F , and G are the (zero-order) Hankel transforms of u , f , and g , respectively.

Solving (2.9)-(2.10) gives

$$U(\lambda, t) = F(\lambda) \cos(\lambda ct) + \frac{G(\lambda)}{\lambda c} \sin(\lambda ct).$$

Inverting the transform yields

$$u(r, t) = \int_0^\infty \lambda F(\lambda) \cos(\lambda ct) J_0(\lambda r) d\lambda + \frac{1}{c} \int_0^\infty G(\lambda) \sin(\lambda ct) J_0(\lambda r) d\lambda. \quad (2.11)$$

For specific choices of f , g , and c , it is possible to evaluate the integrals in closed form. For example, if

$$c = 1, \quad f(r) = \frac{\alpha}{\sqrt{\alpha^2 + r^2}}; \quad g(r) = 0, \quad \alpha = \text{constant},$$

then the functions F and G become

$$F(\lambda) = \frac{\alpha}{\lambda} e^{-\lambda \alpha}; \quad G(\lambda) = 0, \quad \alpha > 0.$$

Inserting these into (2.11) and evaluating the integral yields (letting \Re and \Im denote real and imaginary parts, respectively),

$$u(r, t) = \Re \left(\frac{\alpha}{\sqrt{(\alpha + it)^2 + r^2}} \right).$$

So, in terms of x and z , if

$$f_\alpha(x, z) = \left(\frac{\alpha}{\sqrt{\alpha^2 + x^2 + z^2}} \right), \quad (2.12)$$

then

$$u(x, z, t) = \Re \left(\frac{\alpha}{\sqrt{(\alpha + it)^2 + x^2 + z^2}} \right),$$

from which it follows that

$$\partial_t u = \Im \left(\frac{\alpha(\alpha + it)}{((\alpha + it)^2 + x^2 + z^2)^{3/2}} \right), \quad (2.13)$$

$$\partial_x u = -\Re \left(\frac{\alpha x}{((\alpha + it)^2 + x^2 + z^2)^{3/2}} \right),$$

and

$$\partial_z u = -\Re \left(\frac{\alpha z}{((\alpha + it)^2 + x^2 + z^2)^{3/2}} \right).$$

CHAPTER 3

First Order Linear Hyperbolic Systems

In this chapter, first order hyperbolic systems are examined. Motivated by model problems presented in the previous chapter, it will be assumed that these systems are linear with coefficients which may depend on spatial location, \mathbf{x} , but are independent of time, t . The first section contains a general discussion of such systems. In the second section, the first order system forms for some of the wave equations of the previous chapter are presented. When a dimensional, or Strang, splitting is applied to multi-dimensional systems, the problems are reduced to coupled one-dimensional systems. The diagonalization of the one-dimensional systems is discussed in the third section. The third section also discusses diagonalization of related systems which arise when a dimensional, or Strang, splitting is applied to these systems.

First Order Linear Hyperbolic Systems

The general *linear first order system* in two independent variables, x and t , is (using summation notation)

$$S_{ij}\partial_t v_j + A_{ij}\partial_x v_j = B_{ij}v_j + f_i, \quad i = 1, \dots, n. \quad (3.1)$$

The matrices \mathbf{S} , \mathbf{A} , and \mathbf{B} and the vector \mathbf{f} may be functions of x and t . However, it will be assumed that \mathbf{S} is invertible, so, without loss of generality, take \mathbf{S} to be the identity. Furthermore, \mathbf{A} and \mathbf{B} will be assumed to be independent of t . Equation (3.1) then reduces to

$$\partial_t \mathbf{v} + A(x)\partial_x \mathbf{v} = B(x)\mathbf{v} + \mathbf{f}(x, t). \quad (3.2)$$

Definition 3.1 System (3.2) is said to be hyperbolic if, for each x , the matrix $\mathbf{A}(x)$ is diagonalizable with real eigenvalues.

Suppressing the dependence on x , assume that

$$A = EDE^{-1},$$

where D is diagonal and E is a matrix whose columns are eigenvectors of A . The diagonal entries of D are the corresponding eigenvalues of A . Left multiplying (3.2) by E^{-1} and defining $\mathbf{w} = \mathbf{w}(x, t) = E^{-1}(x)\mathbf{v}(x, t)$,

$$\partial_t \mathbf{w} + D \partial_x \mathbf{w} = D(\partial_x E^{-1})E\mathbf{w} + E^{-1}B E\mathbf{w} + E^{-1}\mathbf{f}. \quad (3.3)$$

Note that the left-hand side is decoupled. One can obtain the *characteristic curves*, $x_i(t)$, for (3.2) by solving the initial value problems

$$\frac{dx_i}{dt} = D_{ii}(x_i), \quad i = 1, \dots, n. \quad (3.4)$$

Hyperbolicity guarantees (see [22]) that n characteristic curves pass through each point (x, t) .

In d space dimensions ($d = 2$ or 3), linear systems of the form

$$\partial_t \mathbf{v} + \sum_{m=1}^d A_m(\mathbf{x}) \partial_{x_m} \mathbf{v} = B(\mathbf{x})\mathbf{v} + \mathbf{f}(\mathbf{x}, t) \quad (3.5)$$

will be considered. In this case, characteristic curves become characteristic surfaces, and such a system will be hyperbolic if n characteristic surfaces pass through each point [22]. It will be assumed that the matrices, A_m , in (3.5) are diagonalizable. However, they need not be simultaneously diagonalizable.

First Order System Form for Some Wave Equations

The one-dimensional acoustic, or scalar, wave equation

$$\partial_t^2 u - c^2 \partial_x^2 u = 0 \quad (3.6)$$

is equivalent to the first order system,

$$\partial_t \mathbf{v} + A_1 \partial_x \mathbf{v} = \mathbf{0}, \quad (3.7)$$

where

$$\mathbf{v} = \begin{bmatrix} \partial_x u \\ \partial_t u \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & -1 \\ -c^2 & 0 \end{bmatrix}.$$

For each fixed x , the matrix A_1 in system (3.7) has two distinct eigenvalues, $c(x)$ and $-c(x)$, and, hence, is diagonalizable. Assuming $c(x)$ is not zero for any x , the system (3.7) has two distinct, real eigenvalue and is, therefore, hyperbolic.

The two-dimensional scalar wave equation,

$$\partial_t^2 u - c^2(\partial_x^2 u + \partial_y^2 u) = 0 \quad (3.8)$$

is equivalent to the first order system,

$$\partial_t \mathbf{v} + A_1 \partial_x \mathbf{v} + A_2 \partial_y \mathbf{v} = \mathbf{0}, \quad (3.9)$$

where now

$$\mathbf{v} = \begin{bmatrix} \partial_x u \\ \partial_y u \\ \partial_t u \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & 0 & -1 \\ 0 & 0 & 0 \\ -c^2 & 0 & 0 \end{bmatrix}, \quad A_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & -c^2 & 0 \end{bmatrix}.$$

It will be shown in the next section that each of the matrices A_1 and A_2 is diagonalizable. Similarly, the three-dimensional version,

$$\partial_t^2 u - c^2(\partial_x^2 u + \partial_y^2 u + \partial_z^2 u) = 0 \quad (3.10)$$

is equivalent to the first order system,

$$\partial_t \mathbf{v} + A_1 \partial_x \mathbf{v} + A_2 \partial_y \mathbf{v} + A_3 \partial_z \mathbf{v} = \mathbf{0}, \quad (3.11)$$

where

$$\mathbf{v} = \begin{bmatrix} \partial_x u \\ \partial_y u \\ \partial_z u \\ \partial_t u \end{bmatrix}, \quad A_1 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -c^2 & 0 & 0 & 0 \end{bmatrix},$$

$$A_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 \\ 0 & -c^2 & 0 & 0 \end{bmatrix}, \text{ and } A_3 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -c^2 & 0 \end{bmatrix}.$$

Again, each of A_1 , A_2 , and A_3 is diagonalizable.

The two-dimensional elastic wave equation,

$$\begin{aligned} \partial_t^2 u_i - c_1^2 \partial_x^2 u_i - c_2^2 \partial_y^2 u_i &= (c_1^2 - c_2^2) \partial_x \partial_y u_j \\ \partial_t^2 u_j - c_2^2 \partial_x^2 u_j - c_1^2 \partial_y^2 u_j &= (c_1^2 - c_2^2) \partial_x \partial_y u_i \end{aligned} \quad (3.12)$$

is equivalent to the first order system,

$$\partial_t \mathbf{v} + P_1 \partial_x \mathbf{v} + P_2 \partial_y \mathbf{v} = 0, \quad (3.13)$$

where

$$\mathbf{v} = \begin{bmatrix} \partial_x u_i \\ \partial_y u_i \\ \partial_x u_j \\ \partial_y u_j \\ \partial_t u_i \\ \partial_t u_j \end{bmatrix}, \quad P_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -c_1^2 & 0 & 0 & -c_1^2 & 0 & 0 \\ 0 & c_2^2 & -c_2^2 & 0 & 0 & 0 \end{bmatrix},$$

and

$$P_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & -c_2^2 & c_2^2 & 0 & 0 & 0 \\ -c_1^2 & 0 & 0 & -c_1^2 & 0 & 0 \end{bmatrix}.$$

Once again, the matrices P_1 and P_2 are diagonalizable and have real eigenvalues.

Diagonalization in the Acoustic Case

Motivated by dimensional splitting, this section deals with the transformation of the linear first order hyperbolic operator,

$$L_s \mathbf{v} \equiv \partial_t \mathbf{v} + A_m \partial_s \mathbf{v}, \quad (3.14)$$

into a diagonal (or essentially decoupled) form. The matrix A_m represents one of the matrices that arose in the first order systems (3.7), (3.9) and (3.11). The s in (3.14) is x , y , or z if $m = 1, 2$ or 3 , respectively. The only non-zero eigenvalues of A_m are c and $-c$ — independent of the number of spatial dimensions. As was done in equation (3.3), the change of variables,

$$\mathbf{w} = E^{-1}\mathbf{v},$$

transforms the operator in (3.14) into

$$E^{-1}L_s\mathbf{v} = \partial_t\mathbf{w} + D\partial_s\mathbf{w} - D(\partial_s E^{-1})E\mathbf{w} \equiv \tilde{L}_s\mathbf{w} \quad (3.15)$$

so that

$$L_s\mathbf{v} = E(\tilde{L}_s\mathbf{w}).$$

The matrices E (denoted E_m whenever clarity is needed) and D , utilized in the diagonalization of A_m , are recorded in Table 1.

The matrices $D(\partial_s E^{-1})E$ that appear in (3.15) are listed in Table 2. This result is independent of m . Note that each matrix in Table 2 contains the same 2×2 nonzero block structure. This structure will be exploited in the formulation of the numerical method.

The vector $\tilde{L}_s\mathbf{w}$ for the operator associated with the one-dimensional problem is thus

$$\tilde{L}_s\mathbf{w} = \begin{bmatrix} \partial_t w^{(1)} + c\partial_s w^{(1)} & -\frac{\partial_s c}{2}(w^{(2)} - w^{(1)}) \\ \partial_t w^{(2)} - c\partial_s w^{(2)} & -\frac{\partial_s c}{2}(w^{(2)} - w^{(1)}) \end{bmatrix}. \quad (3.16)$$

Operators defined by the A_m found in the two and three dimensional problems will have the same two entries as in (3.16) together with one or two additional entries. The additional entry in the operators associated with the two dimensional problem is $\partial_t w^{(3)}$. The operators associated with the three-dimensional problem contain (in addition to the two entries in (3.16)) the entries $\partial_t w^{(3)}$ and $\partial_t w^{(4)}$.

	$1 - D$	$2 - D$	$3 - D$
D	$\begin{bmatrix} c & 0 \\ 0 & -c \end{bmatrix}$	$\begin{bmatrix} c & 0 & 0 \\ 0 & -c & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} c & 0 & 0 & 0 \\ 0 & -c & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$
E_1	$\begin{bmatrix} 1 & 1 \\ -c & c \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 1 \\ -c & c & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -c & c & 0 & 0 \end{bmatrix}$
E_2		$\begin{bmatrix} 0 & 0 & 1 \\ 1 & 1 & 0 \\ -c & c & 0 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -c & c & 0 & 0 \end{bmatrix}$
E_3			$\begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 \\ -c & c & 0 & 0 \end{bmatrix}$

Table 1: Matrices arising in the diagonalization of the first order system operator (3.14).

	$1 - D$	$2 - D$	$3 - D$
$D(\partial_s E^{-1})E$	$\begin{bmatrix} -\frac{\partial_s c}{2} & \frac{\partial_s c}{2} \\ -\frac{\partial_s c}{2} & \frac{\partial_s c}{2} \end{bmatrix}$	$\begin{bmatrix} -\frac{\partial_s c}{2} & \frac{\partial_s c}{2} & 0 \\ -\frac{\partial_s c}{2} & \frac{\partial_s c}{2} & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} -\frac{\partial_s c}{2} & \frac{\partial_s c}{2} & 0 & 0 \\ -\frac{\partial_s c}{2} & \frac{\partial_s c}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$

Table 2: The matrices $D(\partial_s E^{-1})E$ in (3.15).

Diagonalization in the Elastic Case

Once again motivated by dimensional splitting, this section deals with the transformation of the linear first order hyperbolic operator,

$$L_s \mathbf{v} \equiv \partial_t \mathbf{v} + P_m \partial_s \mathbf{v}, \quad (3.17)$$

into a diagonal (or essentially decoupled) form. The matrix P_m represents the 6×6 matrices that arose in the first order system (3.13). The s in (3.17) is x or z if $m = 1$ or 2 , respectively. The non-zero eigenvalues of P_m are $\pm c_1$ and $\pm c_2$. As was done in equation (3.3), the change of variables, $\mathbf{w} = E^{-1} \mathbf{v}$, transforms the operator in (3.17) to (3.15). The matrices E_1 , E_2 , and D , utilized in the diagonalization of the P_m , are recorded in Table 3. The matrix $D(\partial_s E^{-1})E$ that appears in (3.15) is also listed in Table 3. This result is independent of m . Note that this matrix contains two 2×2 nonzero blocks, each of which are similar to the nonzero blocks in Table 2. The entries of the vector $\tilde{L}_s \mathbf{w}$ defined in (3.15), are thus

$$\tilde{L}_s \mathbf{w} = \begin{bmatrix} \partial_t w^{(1)} + c_1 \partial_s w^{(1)} & - \frac{\partial_s c_1}{2} (w^{(2)} - w^{(1)}) \\ \partial_t w^{(2)} - c_1 \partial_s w^{(2)} & - \frac{\partial_s c_1}{2} (w^{(2)} - w^{(1)}) \\ \partial_t w^{(3)} + c_2 \partial_s w^{(3)} & - \frac{\partial_s c_2}{2} (w^{(4)} - w^{(3)}) \\ \partial_t w^{(4)} - c_2 \partial_s w^{(4)} & - \frac{\partial_s c_2}{2} (w^{(4)} - w^{(3)}) \\ \partial_t w^{(5)} & \\ \partial_t w^{(6)} & \end{bmatrix}. \quad (3.18)$$

D	$\begin{bmatrix} c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -c_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & c_2 & 0 & 0 & 0 \\ 0 & 0 & 0 & -c_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
E_1	$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -c_1 & c_1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -c_2 & c_2 & 0 & 0 \end{bmatrix}$
E_2	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & -1 \\ 0 & 0 & -c_2 & c_2 & 0 & 0 \\ -c_1 & c_1 & 0 & 0 & 0 & 0 \end{bmatrix}$
$D(\partial_s E^{-1})E$	$\begin{bmatrix} -\frac{\partial_s c_1}{2} & \frac{\partial_s c_1}{2} & 0 & 0 & 0 & 0 \\ -\frac{\partial_s c_1}{2} & \frac{\partial_s c_1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{\partial_s c_2}{2} & \frac{\partial_s c_2}{2} & 0 & 0 \\ 0 & 0 & -\frac{\partial_s c_2}{2} & \frac{\partial_s c_2}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

Table 3: Matrices arising in the diagonalization of systems in (3.17).

CHAPTER 4

Dimensional Splitting Methods

The numerical scheme that is examined in Chapter 5 is designed to provide high resolution approximations to the diagonalized one dimensional systems presented in the previous chapter. In order to use the numerical scheme on problems of more than one spatial dimension, a splitting method is employed. The following section defines the one dimensional propagator, which is the operator that propagates the solution to a one dimensional problem from one point in time to the next. Once the propagators are defined for the one- and two-dimensional problems, the dimensional splitting methods due to Strang [20] are presented. The second section derives the Strang splittings as well as extensions yielding higher orders of accuracy. Finally, a second order splitting for three-dimensional problems is presented.

Propagators and Dimensional Splitting

Expanding the vector valued function, $\mathbf{v}(x, t)$, in a Taylor series about t yields

$$\mathbf{v}(x, t + \Delta t) = (1 + \Delta t \partial_t + \frac{1}{2} \Delta t^2 \partial_t^2 + \dots) \mathbf{v}(x, t). \quad (4.1)$$

If \mathbf{v} is a solution to

$$\partial_t \mathbf{v} + A \partial_x \mathbf{v} = 0, \quad (4.2)$$

then using (4.2) to replace the temporal derivatives with spatial derivatives, (4.1)

becomes

$$\mathbf{v}(x, t + \Delta t) = (1 - \Delta t A \partial_x + \frac{1}{2} \Delta t^2 (A \partial_x)^2 - \dots) \mathbf{v}(x, t). \quad (4.3)$$

Note that (4.3) can be expressed as

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

where

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

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

$$\partial_t \mathbf{v} + A \partial_x \mathbf{v} + B \partial_y \mathbf{v} = \mathbf{0}, \quad (4.4)$$

the propagator is expressed as the exponential term in

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

In many cases of interest (c.f., equation 3.9), the matrices A and B do not commute, so the two-dimensional propagator cannot be written as the product of two one-dimensional propagators, as the exponential form might suggest. It will be shown that the two-dimensional propagator can, however, be approximated by the product of one-dimensional propagators. The numerical implementation of the one-dimensional propagators is discussed in Chapter 5.

In [20], Strang showed

$$e^{-\Delta t (A \partial_x + B \partial_y)} = e^{-\Delta t A \partial_x} e^{-\Delta t B \partial_y} + O(\Delta t^2), \quad (4.5)$$

so that a first order accurate approximation is based on

$$\mathbf{v}(x, y, t^m + \Delta t) = e^{-\Delta t A \partial_x} (e^{-\Delta t B \partial_y} \mathbf{v}(x, y, t^m)) + O(\Delta t^2). \quad (4.6)$$

An extension of this result, also due to Strang, is

$$e^{-\Delta t (A \partial_x + B \partial_y)} = e^{-\frac{1}{2} \Delta t A \partial_x} e^{-\Delta t B \partial_y} e^{-\frac{1}{2} \Delta t A \partial_x} + O(\Delta t^3). \quad (4.7)$$

As the next section shows, higher order approximations can also be derived.

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

$$\partial_t \mathbf{v} + C \partial_s \mathbf{v} = \mathbf{0}, \quad (4.8)$$

where C is either A or B , corresponding to $s = x$ or $s = y$, respectively. \mathbf{v} is a function of x, y , and t , but one of x or y is treated as a parameter. For example, to implement (4.6), one first propagates data, $\mathbf{v}(x, y, t^m)$, from t^m to $t^m + \Delta t$, by numerically solving

$$\partial_t \mathbf{v} + B \partial_y \mathbf{v} = \mathbf{0} \quad (4.9)$$

for $\mathbf{v}(\cdot, y, t^m + \Delta t)$. This approximates the value of $e^{-\Delta t B \partial_y} \mathbf{v}(x, y, t^m)$ shown on the right in (4.6). Let $\mathbf{v}^{int} = e^{-\Delta t B \partial_y} \mathbf{v}(x, y, t^m)$ be this *intermediate* approximation. Then $\mathbf{v}(x, y, t^m + \Delta t)$ in (4.6) is obtained by solving

$$\partial_t \mathbf{v} + A \partial_x \mathbf{v} = \mathbf{0}, \quad (4.10)$$

with initial data \mathbf{v}^{int} . Specific details appear in Chapters 5 and 6.

Derivation and Extensions of Strang Splittings

This section examines the problem of accurately approximating a higher dimensional propagator by a product of one-dimensional propagators. To keep the notation simple, the problem will be restated. Assuming X and Y are non-commuting linear operators, determine the parameters a and b for which

$$e^{\tau a X} e^{\tau b Y} - e^{\tau(X+Y)} = O(\tau^p). \quad (4.11)$$

If each exponential on the left hand side of (4.11) is expanded about $\tau = 0$, and the terms involving $\tau^n, n = 0, 1, \dots$, are collected, then constraints may be found on a and b . The τ^0 term yields no constraints. The first order term is

$$\tau(aX + bY - X - Y).$$

In order for this first order term to be zero, it must be the case that

$$a = 1, \quad \text{and} \quad b = 1.$$

With these requirements met, the second order term is non-zero, so that (4.11) holds with $\rho = 2$.

Strang suggested that a higher order approximation was possible if the product of three first order propagators were used. In order to determine the parameters a, b , and c for which

$$e^{\tau a X} e^{\tau b Y} e^{\tau c X} - e^{\tau(X+Y)} = O(\tau^\rho), \quad (4.12)$$

the left hand side is once again expanded, and conditions are obtained for a, b , and c . Once again, the zero order term yields no constraints. So that the first order term is zero, it must be that

$$a + c = 1, \quad \text{and} \quad b = 1.$$

Setting $a = 1 - c$, the second order term is zero only if $c = \frac{1}{2}$, and with this choice, the third order term is non-zero. This implies (4.12) holds with $\rho = 3$. Note that the symmetry in the roles of X and Y also implies that

$$e^{\tau \frac{1}{2} Y} e^{\tau X} e^{\tau \frac{1}{2} Y} - e^{\tau(X+Y)} = O(\tau^3).$$

From these first two examples it might be inferred that there is a linear relationship between the order of the resulting approximation and the number of one-dimensional propagators required. Unfortunately, this is not the case. To illustrate, expand the left hand side of

$$e^{\tau a X} e^{\tau b Y} e^{\tau c X} e^{\tau d Y} - e^{\tau(X+Y)} = O(\tau^\rho),$$

in order to determine the parameters a, b, c , and d . In order that the first order term is zero, it must be the case that

$$a + c = 1, \quad \text{and} \quad b + d = 1.$$

Setting $a = 1 - c$ and $b = 1 - d$, the second order term is zero only if $c(1 - d) = \frac{1}{2}$. If $d = 1$, then $b = 0$, and since X does commute with itself, $e^{\tau a X} e^{\tau c X} = e^{\tau(a+c)X}$, and the problem reduces to the case of (4.11). Thus it must be that $c = \frac{1}{2(1-d)}$. With these constraints imposed, the third order term implies that d satisfies $3d^2 - 4d + 1 = 0$ and $d = \frac{1}{4}$. These cannot both hold, and hence, nothing may be gained by using the product of four one-dimensional propagators. In fact, nothing is gained by using the product of five one-dimensional propagators. However, the product of six propagators will result in a higher order method.

Imposing the necessary constraints, it can be shown that

$$e^{\tau a X} e^{\tau b Y} e^{\tau c X} e^{\tau d Y} e^{\tau e X} e^{\tau f Y} - e^{\tau(X+Y)} = O(\tau^4), \quad (4.13)$$

if

$$a = 1, \quad b = -\frac{1}{24}, \quad c = -\frac{2}{3}, \quad d = \frac{3}{4}, \quad e = \frac{2}{3}, \quad \text{and} \quad f = \frac{7}{24}.$$

Note also that this result, while not symmetric, implies that the reverse ordering

$$f = 1, \quad e = -\frac{1}{24}, \quad d = -\frac{2}{3}, \quad c = \frac{3}{4}, \quad b = \frac{2}{3}, \quad \text{and} \quad a = \frac{7}{24},$$

is also a solution.

In Glasner *et al.* [9], there is a constructive proof of the existence of parameters $a_{i,k}$ and b_i^k such that

$$\prod_{i=1}^m e^{\tau a_{i,k} X} e^{\tau b_{i,k} Y} - e^{\tau(X+Y)} = O(\tau^{k+1}) \quad (4.14)$$

for some m . While the method in [9] guarantees that a dimensional splitting method with any prescribed order of accuracy ρ may be obtained, the method may not be optimal. For example, (4.13) is obtained using the product of six one-dimensional propagators, and the method of [9] requires more one-dimensional propagators.

Approximations to the two-dimensional propagator may also be found using sums of products of one-dimensional operators, for example,

$$k_1 e^{\tau a X} + k_2 e^{\tau b Y} - e^{\tau(X+Y)} = O(\tau^2),$$

if

$$k_1 = k_2 = \frac{1}{2}, \quad \text{and} \quad a = b = 1.$$

As a final note, the ideas examined here may be extended to higher dimensional cases. For example if X , Y and Z do not commute,

$$e^{\tau \frac{1}{2} X} e^{\tau \frac{1}{2} Y} e^{\tau Z} e^{\tau \frac{1}{2} Y} e^{\tau \frac{1}{2} X} - e^{\tau(X+Y+Z)} = O(\tau^3).$$

This result may also be extended by the procedure outlined in this section.

CHAPTER 5

**Adaptive Stencil Finite Difference
Approximations**

This chapter outlines the adaptive stencil finite difference scheme. Since finite difference approximations are based on Taylor's Theorem, it is presented first. The second section uses Taylor's Theorem to determine various finite difference approximations to the first derivative of a function. In the third section, the need for adapting the finite difference approximations is examined. The fourth section deals with the CIR method. The fifth section presents the adaptive stencil finite difference scheme in order to extend the CIR method to higher spatial order. In the sixth section, essentially non-oscillatory (ENO) schemes are formally defined and the history of the ENO based schemes is outlined. The direct correspondence between the ENO polynomial interpolant selection scheme and the adaptive finite difference stencil selection scheme is also discussed. A means by which temporal accuracy may be extended is presented in the seventh section. The ENO based adaptive stencil method is highly non-linear since the solution is used to determine the choice of stencil. For this reason, linear stability analysis does not apply for this method. It is possible, however, to establish a CFL condition, a condition on the temporal to spatial mesh ratio that is necessary for convergence of a method, as shown in the eighth section. Evidence that the method is stable is given in the final section of this chapter. Further evidence of stability and a discussion of stability requirements for nonlinear methods will be presented in the final chapter. Numerical results are also presented.

Taylor's Theorem

Taylor's Theorem is a fundamental theorem in numerical analysis. The proof is well known and is contained in many texts. See, for example, [10], [3].

Theorem 5.1 *Suppose $f \in C^n[a, b]$ and $f^{(n+1)}$ exists on $[a, b]$. Let $x_0 \in [a, b]$. For every $x \in [a, b]$, there exists $\xi(x)$ between x_0 and x with*

$$f(x) = P_n(x) + R_n(x),$$

where

$$\begin{aligned} P_n(x) &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \dots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n \\ &= \sum_{k=0}^n \frac{f^{(k)}(x_0)}{k!}(x - x_0)^k, \end{aligned}$$

and

$$R_n(x) = \frac{f^{(n+1)}(\xi(x))}{(n+1)!}(x - x_0)^{n+1}.$$

The function $P_n(x)$ is called the n^{th} Taylor polynomial for f about x_0 , and $R_n(x)$ is called the remainder term (or truncation error) associated with $P_n(x)$.

Finite Difference Approximations

Suppose a function, f , satisfies the conditions of Taylor's theorem for $n = 1$ on some interval containing x_0 and $x_1 = x_0 + \Delta x$. Then

$$\begin{aligned} f(x) &= P_1(x) + R_1(x) \\ &= f(x_0) + f'(x_0)(x - x_0) + \frac{f''(\xi(x))}{2!}(x - x_0)^2. \end{aligned}$$

Thus

$$f(x_1) = f(x_0) + \Delta x f'(x_0) + \frac{\Delta x^2}{2!} f''(\xi(x)),$$

which implies

$$f'(x_0) = \frac{f(x_1) - f(x_0)}{\Delta x} + \mathcal{O}(\Delta x), \quad (5.1)$$

provided $f \in C^2[x_0, x_1]$.

Similarly, if f is C^2 on the interval $[x_{-1}, x_0]$, where $x_{-1} = x_0 - \Delta x$, then

$$f(x_{-1}) = f(x_0) - \Delta x f'(x_0) + \frac{\Delta x^2}{2!} f''(\xi(x)).$$

This implies that

$$f'(x_0) = \frac{f(x_0) - f(x_{-1})}{\Delta x} + \mathcal{O}(\Delta x). \quad (5.2)$$

If f is C^2 in a neighborhood of x_0 that contains both x_{-1} and x_1 , then either approximation may be used. Note, however, that the approximation to $f'(x_0)$ in (5.1) is actually an approximation to the right derivative of f at the point x_0 . This is due to the fact that f need not be defined to the left of x_0 for Taylor's theorem (and hence, this approximation) to be valid. Similarly, the approximation to $f'(x_0)$ in (5.2) is an approximation to the left derivative of f at x_0 .

It is possible to find other $\mathcal{O}(\Delta x)$ approximations to $f'(x_0)$. However, under the assumption that f can only be sampled at equi-spaced points $x_{-m}, \dots, x_0, \dots, x_M$, where $x_{i+1} - x_i = \Delta x$, (5.1) and (5.2) are the only two approximations that are $\mathcal{O}(\Delta x)$ accurate on intervals containing x_0 whose maximum length is Δx .

For $n = 2$, there are exactly three $\mathcal{O}(\Delta x^2)$ approximations on intervals containing x_0 whose maximum length is $2\Delta x$. Assuming f is C^3 on the interval $[x_0, x_2]$,

$$f(x) = f(x_0) + f'(x_0)(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \frac{f'''(\xi(x))}{3!}(x - x_0)^3. \quad (5.3)$$

An $\mathcal{O}(\Delta x^2)$ approximation may be obtained by determining constants a_0, a_1 , and a_2 such that

$$a_0 f(x_0) + a_1 f(x_1) + a_2 f(x_2) = (2!) \Delta x f'(x_0) + \mathcal{O}(\Delta x^3) \quad (5.4)$$

so that division by $(2!)\Delta x$ yields the approximation

$$f'(x_0) = \frac{a_0 f(x_0) + a_1 f(x_1) + a_2 f(x_2)}{2!} + \mathcal{O}(\Delta x^2).$$

Using (5.3) to write $f(x_1)$ and $f(x_2)$ in terms of $f(x_0)$, one can obtain conditions on the unknowns. The unknown coefficients on the resulting $f(x_0)$ terms require that $a_0 + a_1 + a_2 = 0$. The coefficients on the $f'(x_0)$ terms require $a_1 + 2a_2 = 2$, and the coefficients on the $f''(x_0)$ terms require that $a_1 + 4a_2 = 0$. Solving this system of equations yields

$$a_0 = -3, \quad a_1 = 4, \quad \text{and} \quad a_2 = -1,$$

so that

$$f'(x_0) = \frac{-3f(x_0) + 4f(x_1) - 1f(x_2)}{2!} + \mathcal{O}(\Delta x^2). \quad (5.5)$$

The other two $\mathcal{O}(\Delta x^2)$ approximations as well as higher order approximations may be found in a similar manner. The coefficients needed to determine the approximations

$$f'(x_i) = \frac{a_0 f(x_0) + a_1 f(x_1) + \dots + a_n f(x_n)}{n! \Delta x} + \mathcal{O}(\Delta x^n)$$

are listed in Table 4 for $n = 1, \dots, 5$.

The symbols on the right in Table 4 are shorthand that will be used to refer to each of the finite difference stencils. The symbol \circ denotes the point at which the derivative is being approximated. The symbols \circ denote additional points that are used. For example, the symbol, or stencil, for the finite difference approximation (5.2) would be $\circ \circ$, and for (5.5), $\circ \circ \circ$ will be used.

A Motivating Example

The approximations considered in the following example are each second order accurate (see $n = 2$ in Table 4). This example points out the need, and the potential benefits, for a method that adapts the finite difference stencil.

		i	a_0	a_1		
For $n = 1$,	0	-1	1	○	○	
	1	1	-1	○	○	

		i	a_0	a_1	a_2			
For $n = 2$,	0	-3	4	-1	○	○	○	
	1	-1	0	1	○	○	○	
	2	1	-4	3	○	○	○	

		i	a_0	a_1	a_2	a_3				
For $n = 3$,	0	-11	18	-9	2	○	○	○	○	
	1	-2	-3	6	-1	○	○	○	○	
	2	1	-6	3	2	○	○	○	○	
	3	-2	9	-18	11	○	○	○	○	

		i	a_0	a_1	a_2	a_3	a_4					
For $n = 4$,	0	-50	96	-72	32	-6	○	○	○	○	○	
	1	-6	-20	36	-12	2	○	○	○	○	○	
	2	2	-16	0	16	-2	○	○	○	○	○	
	3	-2	12	-36	20	6	○	○	○	○	○	
	4	6	-32	72	-96	50	○	○	○	○	○	

		i	a_0	a_1	a_2	a_3	a_4	a_5						
For $n = 5$,	0	-274	600	-600	400	-150	24	○	○	○	○	○	○	
	1	-24	-130	240	-120	40	-6	○	○	○	○	○	○	
	2	6	-60	-40	120	-30	4	○	○	○	○	○	○	
	3	-4	30	-120	40	60	-6	○	○	○	○	○	○	
	4	6	-40	120	-240	130	24	○	○	○	○	○	○	
	5	-24	150	-400	600	-600	274	○	○	○	○	○	○	

Table 4: Necessary constants in $\mathcal{O}(\Delta x^n)$ approximations to the first derivative.

The first frame of Figure 1 contains the graph of a function, $f(x)$ —the function whose derivative is to be approximated. It is clear that the function is only piecewise smooth. For this reason, attempts are made to approximate the left and/or right derivatives, which this function has at every point.

The small circles on the first frame of Figure 1 are the discrete data, $f(x_i)$. The continuous function, $f(x)$, and the circles are shown in each of the remaining frames for reference purposes.

The derivative of a function at a certain point corresponds to the slope of the line that is tangent to the function at that point. Thus, an approximate derivative gives an approximation of the slope of the tangent line. Presented in frame 2 of Figure 1 are the slopes that the derivative approximation, based on the stencil $\circ \circ \circ$, would assign. When the approximation is accurate, the line segment that passes through each circle, or node, should be nearly tangent to f at that node. Note that for $x = 0, 3, 4$ or 6 , the lines shown appear to be tangent, implying that the approximations are fairly accurate. At $x = 5$, a relative maximum, the line is nearly horizontal, implying that the approximation, while not exact, is nearly correct. Note also that at $x = 2$ and $x = 8$, the lines accurately approximate the slope of 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 because f is not smooth enough on the interval containing the nodes that are used; in this case the interval $[1, 3]$.

Frame 3 of Figure 1 presents results obtained when applying the centered stencil $\circ \bullet \circ$. From the figure, it is clear that this centered stencil provides a quite accurate approximate at most of the points. However, at $x = 2$ and $x = 8$, the slope of the approximate tangent line approximates neither the slope of the true left nor

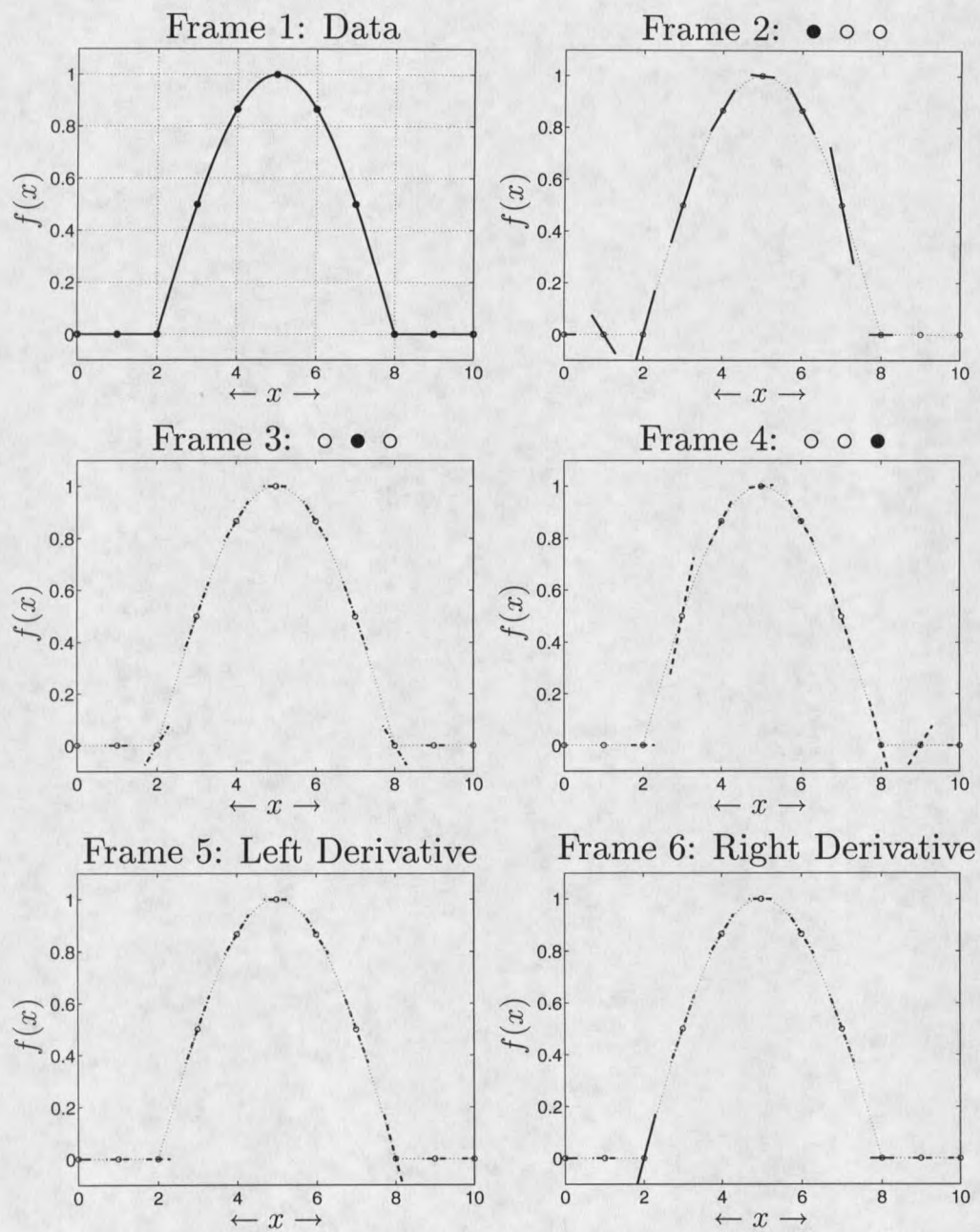


Figure 1: Various second order approximations to the derivative.

the true right derivative. Instead, it gives an average of the two. Also, this stencil may not be applied at $x = 0$ and $x = 10$.

Frame 4 contains similar information concerning the stencil $\circ \circ \circ$. Due to both the symmetry in the stencil and the symmetry of the problem, this information is a mirror of the $\circ \circ \circ$ approximation. Here again the stencil works well at certain points, and not so well at others. Note that it approximates the left derivative quite well at $x = 2$ and $x = 8$, the points of discontinuity of $f'(x)$.

Frames 2, 3, and 4 illustrate the inaccurate approximations which result if one fixed stencil is used on the interior points. Depending on how the derivative is used, the error in the derivative may be compounded in later calculations. Also note that all three of the stencils yield accurate approximations at $x = 4, 5$ and 6 , implying that it is not necessary to require the use of $\circ \circ \circ$ at these points to obtain the accurate approximations to the left and right derivative. The accurate approximations that were developed each used only two of the three possible stencils. For the approximation to the left derivative, there was no need to make use of the stencil that used only information to the right, $\circ \circ \circ$. Similarly, $\circ \circ \circ$ was not needed when approximating the right derivative.

Many numerical methods that utilize finite difference approximations for derivatives modify the stencil near a computational boundary. A common method for approximating the derivative would use $\circ \circ \circ$ at each interior point and use $\circ \circ \circ$ at $x = 0$ and $\circ \circ \circ$ at $x = 10$. The problem with this approach is that $\circ \circ \circ$ does not perform well at the interior points $x = 2$ and $x = 8$. However, if $\circ \circ \circ$ is used at $x = 2$ and $x = 8$ and $\circ \circ \circ$ everywhere else (except the endpoints), then one obtains an accurate approximation of the left derivative of f at every point (see Frame 5). Similarly, if $\circ \circ \circ$ is used at $x = 2$ and $x = 8$ then one obtains an equally accurate approximation of the right derivative. These and other boundary considerations will

be discussed in Chapter 6.

The CIR Method

Courant, Isaacson and Rees introduced what has become known as the CIR method in 1952 [6] (see also [10, p. 48]). In addition, they introduced the notion of *upwinding*. Upwinding insures the utilization of directional information in wave propagation problems. The adaptive stencil scheme presented later may be viewed as an extension of CIR.

The CIR method may be applied to first order systems in one space dimension to approximate the linear operators (3.14) and (3.17). CIR may also be applied to approximate the diagonalized operator (3.15). If the coefficients are constant, then the CIR method yields the same approximations to (3.14) and (3.17) as it does to the corresponding operator (3.15). If, however, the material parameters are not constant, then the approximations will differ. The numerical method that will be outlined later is an extension of the CIR method as it is applied to the diagonalized operator (3.15).

To illustrate the CIR method, consider the one-dimensional hyperbolic system

$$\partial_t \mathbf{w} + D \partial_x \mathbf{w} = F \mathbf{w}, \quad (5.6)$$

where D is an $n \times n$ diagonal matrix and F is an $n \times n$ matrix which couples the components of the n -vector, \mathbf{w} (for example, the systems in (3.3)). The left hand side of (5.6) is decoupled. With the spatial and temporal discretizations $\mathbf{w}_j^m \equiv [w_{1j}^m, \dots, w_{nj}^m]^T$, where $w_{ij}^m = w_i(m\Delta t, j\Delta x)$ and $m \geq 0$, $j = \dots, -1, 0, 1, \dots$, the CIR method yields

$$\frac{\mathbf{w}_j^{m+1} - \mathbf{w}_j^m}{\Delta t} + \begin{bmatrix} \lambda_1 \Delta_x(\lambda_1) w_{1j}^m \\ \vdots \\ \lambda_n \Delta_x(\lambda_n) w_{nj}^m \end{bmatrix} = F \mathbf{w}_j^m, \quad (5.7)$$

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

$$\Delta_x(\lambda)f_j = \begin{cases} \frac{f_j - f_{j-1}}{\Delta x}, & \text{if } \lambda > 0 \quad \circ \circ \\ \frac{f_{j+1} - f_j}{\Delta x}, & \text{if } \lambda < 0 \quad \circ \circ \end{cases} \quad (5.8)$$

for each j .

Note that the stencil *adapts* to the sign of λ . For this reason, CIR can be viewed as an adaptive stencil finite difference method. The requirement that the selection of the spatial difference stencils depends on the sign of λ_i insures (and in fact defines) upwinding. Numerical evidence (see [21, pp. 80-83]) shows that upwind schemes, including CIR, deal with numerical dispersion better than standard methods based directly on difference approximations to the second derivatives in the underlying second order problem.

The local truncation error for the CIR method is $O(\Delta x) + O(\Delta t)$ (see [10]). The low order accuracy of the CIR method results in undesirable smearing of wave forms. Higher order accuracy is required to overcome this smearing. Unfortunately, higher order approximations tend to introduce spurious oscillations, as the example in the previous section might suggest. The next section deals with an implementation of the ideas discussed in the previous section in order to obtain increased accuracy in the CIR method in a way which avoids both smearing and spurious oscillations.

Higher Order Adaptive Stencil Selection

The $O(\Delta x^n)$ approximations to the derivative $\partial_x w_i$ at the point x_j contained in Table 4 were obtained using values at $n + 1$ distinct grid points. There are $n + 1$ possible stencils containing x_j that yield approximations with an $O(\Delta x^n)$ truncation error. The example that was presented above indicates that there is no single fixed stencil which is appropriate for many problems. 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.

To determine which $(n + 1)$ spatial grid points to use in the finite difference stencil, the stencil is built up, starting with the point x_j and adding 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 2. Again, the solid dot \bullet refers to the point x_j , and the circles \circ refer to other points that are used in the stencil.

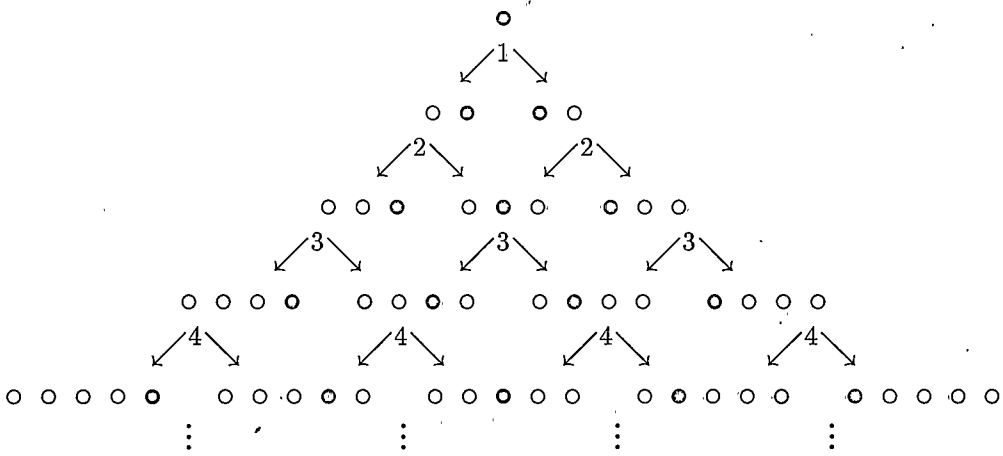


Figure 2: Tree of possible finite difference stencils.

The stencils corresponding to $O(\Delta x^n)$ approximations 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, the point that is in the upwind direction is added to the stencil. The upwind direction is determined by the sign of λ_i , as with the CIR method, c.f., (5.8). The arrow to the left at level 1 is taken (adding a point to the left end of the current stencil) if $\lambda_i > 0$. Otherwise the arrow to the right is followed. 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 [7, 21] for examples).

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 $n > 1$. The remaining decisions are based on the values contained in the vectors $\{D^k(f)\}_{k=2}^{n+1}$. Construct the adapted stencil as follows: for each level k , the arrow to the left is taken (i.e., a point is added to the left of the current stencil) if

$$|D^k(f)_i| < |D^k(f)_{i+1}|. \quad (5.9)$$

Otherwise the one on the right is taken.

The value of i in (5.9) 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.

Error analyses for finite difference methods are typically based on the remainder term $R_n(x)$. The purpose of developing an adaptive strategy (be it adaptive stencil or adaptive mesh) is to keep the remainder term small. Adaptive mesh methods do this by reducing the mesh spacing, h , in regions in which $R_n(x)$ is suspected to be large by making h^n small. On the other hand, adaptive stencil methods do this by selecting stencils in a manner which keeps estimates of the derivative term in $R_n(x)$ small.

Essentially Non-Oscillatory Schemes

An interpolation method is said to be *essentially non-oscillatory* (ENO) if for a piecewise C^{r+1} function, f , defined on the interval $[a, b]$, the interpolant, P_r , satisfies

$$TV(P_r) \leq TV(f) + O(\Delta x^r). \quad (5.10)$$

Here TV is the total variation, defined by

$$TV(f) \equiv \sup \sum_{j=1}^n |f(x_j) - f(x_{j-1})|,$$

where the sup is taken over all finite partitions of the interval $[a, b]$.

The basis for the ENO interpolation scheme is the non-linear (data dependent) approximation. This scheme can be adapted for the solution to hyperbolic partial differential equations. For example, consider how the ENO scheme would be applied to the problem

$$\partial_t v + a \partial_x v = 0$$

with initial data $v(x, 0) = v_0$. A time stepping scheme will require an approximation to $\partial_x v$ at each point in a discrete spatial mesh $\{x_j\}$. Let $f(x) = v(x, t^m)$. Then $\partial_x v = f'(x)$ is found at each x_j as follows:

- (i) If $a > 0$, let S be the set $\{x_j, x_{j-1}\}$. Otherwise, let S be the set $\{x_j, x_{j+1}\}$.
- (ii) Construct a polynomial that interpolates f at the points in S and the next mesh point to the left.
- (iii) Construct a polynomial that interpolates f at the points in S and the next mesh point to the right.
- (iv) Add to S the point to the left of S or the point to the right of S depending on which polynomial (ii), (iii), has the smaller k^{th} derivative (in magnitude), where k is the number of points in S .
- (v) Continue adding points to S by repeating steps (ii)-(iv) until the set contains $n + 1$ points. This will yield an $O(\Delta x^n)$ approximation to the first derivative.
- (vi) Use the derivative of the polynomial that interpolates the set S , evaluated at x_j , as the approximate for $f'(x_j) = \partial_x v(x_j, t^m)$.

The ENO scheme builds a number of interpolating polynomials of a fixed degree on each interval and chooses the one that most ensures that the solution will

be essentially non-oscillatory. This eliminates oscillations by forcing all derivatives (beyond the first derivative) of the interpolant to be as small as possible.

There is a unique n^{th} degree polynomial that interpolates u at the points x_0, \dots, x_n . This polynomial, $P_n(x)$, is, by Newton's interpolatory divided-difference formula (see [3], pages 100-101),

$$P_n(x) = u[x_0] + \sum_{k=1}^n u[x_0, x_1, \dots, x_k](x - x_0) \cdots (x - x_{k-1})$$

where $u[x_0, \dots, x_k]$ has the form

$$u[x_0, \dots, x_k] = \frac{\sum_{j=0}^k \alpha_j u_j}{(\Delta x)^k}. \quad (5.11)$$

The ENO approximation selection process requires the derivative of P_n evaluated at the point of interest, say x_m , which has the form

$$\frac{d}{dx} P_n(x_m) = \sum_{k=1}^n \frac{\sum_{j=0}^k \alpha_j u_j}{\Delta x} b_k,$$

where the b_k are constants.

If $u(x)$ has a bounded $(n+1)^{\text{st}}$ derivative, then since P_n is a polynomial interpolant of $u(x)$,

$$\begin{aligned} \frac{d}{dx} u(x_m) &= \frac{d}{dx} P_n(x_m) + \mathcal{O}(\Delta x^n) \\ &= \frac{\sum_{j=0}^n \beta_j u_j}{\Delta x} + \mathcal{O}(\Delta x^n). \end{aligned} \quad (5.12)$$

This implies that $\partial_x u$ has an $\mathcal{O}(\Delta x^n)$ accurate finite difference approximation. There is a one-to-one correspondence between the values presented in Table 4 and the values in (5.12), and therefore, with the polynomial interpolants used in the ENO scheme. Although no polynomials are used, the adaptive finite difference scheme described in the previous section was created using this correspondence to decide how to traverse the tree in Figure (2). To see this, note that numerator of each value in (5.11) is the same as one of the values $D^k(u)_i$ needed in (5.9). The ENO scheme adapts according

to the magnitude of the n^{th} derivatives of n^{th} degree polynomials, which is equal to $(n!)$ times the leading coefficient of the polynomial. In other words, the values that are compared at each point in the ENO scheme are scalar multiples of the values used in (5.9), implying the same selection criteria.

Increasing Temporal Accuracy

The adaptive finite difference scheme outlined above is used to obtain highly accurate spatial derivatives in first order hyperbolic systems of the form

$$\partial_t \mathbf{w} + D \partial_s \mathbf{w} = F \mathbf{w}. \quad (5.13)$$

In order to increase the temporal accuracy, the solution, \mathbf{w} , to (5.13) is expanded about time t and evaluated at time $t + \Delta t$, yielding

$$\mathbf{w}(s, t + \Delta t) = (I + \Delta t \partial_t + \frac{1}{2} \Delta t^2 \partial_t^2 + \dots) \mathbf{w}(s, t) \quad (5.14)$$

Gathering the spatial operators D , ∂_s , and F in (5.13), (5.13) can be rewritten as

$$(\partial_t) \mathbf{w} = (F - D \partial_s) \mathbf{w}. \quad (5.15)$$

Since the operators D , ∂_s , and F do not depend on time, they commute with the operator ∂_t , so applying (∂_t) to both sides of (5.15) implies

$$\begin{aligned} (\partial_t)(\partial_t) \mathbf{w} &= (\partial_t)(F - D \partial_s) \mathbf{w} \\ &= (F - D \partial_s)(\partial_t) \mathbf{w} \\ &= (F - D \partial_s)(F - D \partial_s) \mathbf{w} \\ &= (F - D \partial_s)^2 \mathbf{w}. \end{aligned} \quad (5.16)$$

Similarly,

$$(\partial_t)^k \mathbf{w} = (F - D \partial_s)^k \mathbf{w},$$

so if the temporal operators are replaced with spatial operators in (5.14), it is observed that

$$\mathbf{w}(s, t + \Delta t) = (I + \Delta t(F - D\partial_s) + \frac{1}{2}\Delta t^2(F - D\partial_s)^2 + \dots) \mathbf{w}(s, t) \quad (5.17)$$

Taking the first two terms on the right hand side of (5.17) and employing the adaptive spatial derivative approximation found in level 2 of the tree in Figure 2 gives the CIR method (5.7). One can derive higher order methods simply by taking additional terms in the right hand side of (5.17) (increasing temporal accuracy), and by continuing to lower levels in the tree in Figure 2 (increasing spatial accuracy).

The CFL Condition

In the early study of approximate solutions to time-dependent partial differential equations, Courant, Friedrichs, and Lewy [5] found that certain restrictions must be placed on the computational grid. Their idea was to define a sequence of approximate solutions, prove that they converge as the grid is refined, and then show that the limit function must satisfy the partial differential equation, giving existence of a solution. For fixed stencil methods, the Lax Equivalence Theorem [10] gives necessary and sufficient conditions for convergence of the method. For the nonlinear method considered here, this theorem does not apply. However, in the implementation of their idea, Courant, Friedrichs, and Lewy noted that a necessary stability condition for any numerical method is that the domain of dependence of the finite difference method should include the domain of dependence of the partial differential equation, at least in the limit as the mesh spacing goes to zero. This condition has become known as the CFL condition.

For fixed stencil methods, the CFL condition is easily determined. It provides necessary bounds on the the ratio of the temporal and spatial grid sizes. For example,

a finite difference method for

$$\partial_t u + a \partial_x u = 0, \quad (5.18)$$

which uses the values of u at (x_{j-1}, t_n) , (x_j, t_n) , and (x_{j+1}, t_n) in order to determine the value of u at the point (x_j, t_{n+1}) would have the computational triangle shown at the top of Figure 3. Each of the points (x_{j+q}, t_n) , $q = -1, 0, 1$ depend on three points from time t_{n-1} , which, in turn, depend on three others, etc. The triangle shown at the top of Figure 3 contains every point that can affect the value at the point (x_j, t_{n+1}) .

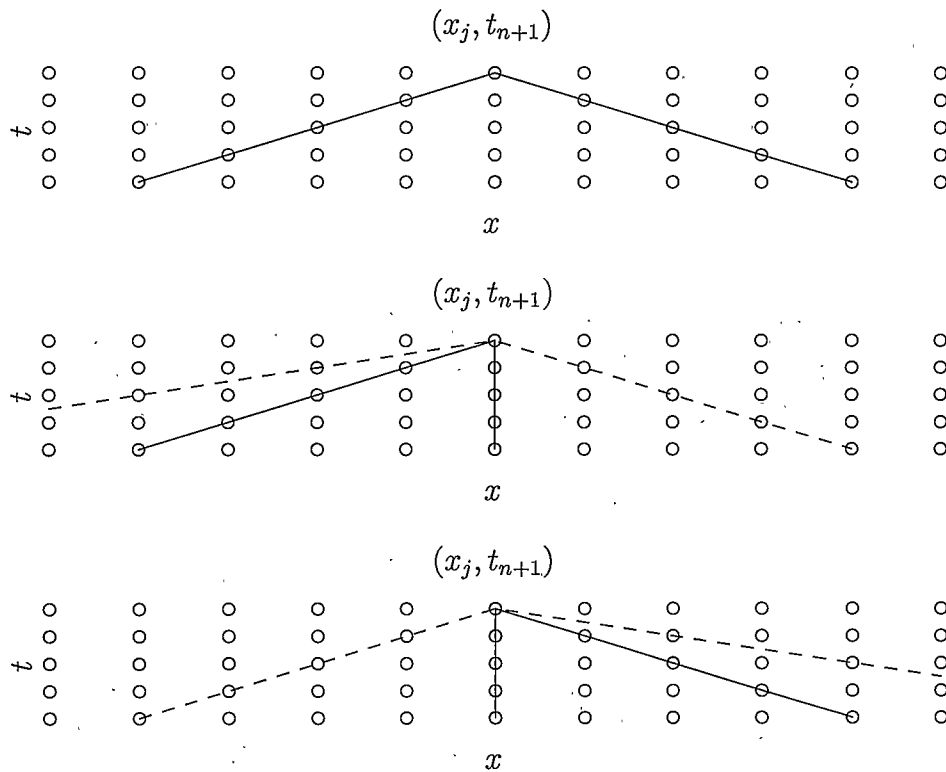


Figure 3: Illustration of the computational domain of dependence.

For this example, the computational domain of dependence satisfies [17]

$$D_k(x_j, t_{n+1}) \subset \left\{ x : |x - x_j| \leq (n+1)\Delta x = \frac{t_{n+1}\Delta x}{\Delta t} \right\}.$$

In order for the domain of dependence for the partial differential equation to be

contained in this set, one must require

$$|(x_j - at_{n+1}) - x_j| \leq \frac{t_{n+1}\Delta x}{\Delta t},$$

i.e.,

$$\left| \frac{a\Delta t}{\Delta x} \right| \leq 1. \quad (5.19)$$

Since a is known a priori, the computational grid may be selected to insure that this inequality will hold.

For higher order methods, more points are used (larger stencils), creating a larger computational domain of dependence. As the computational domain of dependence increases, the constant on the right of (5.19) increases, allowing for a larger Δt to Δx ratio.

Finding a CFL condition for the adaptive stencil finite difference methods is not as straightforward. For a second order method in the spatial derivative, the adaptive stencil finite difference method will choose one of three stencils at each spatial point (see the third level of the tree in Figure 2). The stencil that is chosen at each point is not known a priori, so the computational domain of dependence is not known. A subset of the computational domain may be determined, however. The selection criteria in the adaptive method assures upwinding. This implies that the points (x_{j-1}, t_n) and (x_j, t_n) will be used in order to determine the value of u at the point (x_j, t_{n+1}) if $a > 0$, and the points (x_j, t_n) and (x_{j+1}, t_n) will be used if $a < 0$. Moreover, if $a > 0$, the points (x_{j-2}, t_n) and (x_{j+1}, t_n) may possibly be used, and similarly, if $a < 0$, the points (x_{j-1}, t_n) and (x_{j+2}, t_n) may possibly be used.

The second and third frames of Figure 3 show the computational domains of dependence for the adaptive stencil method for $a > 0$ and $a < 0$ respectfully. The values of u at the points on or below the dotted line *may* affect the value of u at (x_j, t_{n+1}) , and the values of u at the points on or below the solid line are *guaranteed*

to affect the value of u . As the size of the stencil grows, the area under the dotted line will grow, but the solid lines will be fixed.

Due to the adaptive nature of the method, it is not known which of the points between the solid and dashed lines will be used. Since the values of the approximation at these points are not guaranteed to affect the solution, they cannot be included in the computational domain of dependence, and thus may be used when determining the CFL condition. For this reason, the CFL conditions for all of the adaptive stencil methods presented in this chapter are the same, namely

$$\left| \frac{a\Delta t}{\Delta x} \right| \leq 1. \quad (5.20)$$

Since a may be a function of the spatial variable x , in order to maintain a fixed grid, one must require

$$\frac{\Delta t}{\Delta x} \leq \frac{1}{\max_x |a(x)|}. \quad (5.21)$$

For systems of the form (5.6), it must be required that

$$\frac{\Delta t}{\Delta x} \leq \frac{1}{\max_x |\lambda_i(x)|} \quad \forall i = 1, \dots, n. \quad (5.22)$$

Propagation of Information about a Point

Due to the fact that the computational grid is finite, the numerical methods used to propagate information will suffer from resolution problems. If the grid spacing is too coarse, then certain features may be missed or misrepresented by the grid. If a feature is misrepresented, implying that there is some contribution to the data set, then method-to-method comparisons may be made.

For example, consider the data set $\{f(x_i)\}_{i=1}^N$, and assume the true underlying function, $f(x)$, is defined on the entire real line, but that its support is a short interval $[a, b]$. If the x_i are equally spaced with $x_i - x_{i-1} = \Delta x$, then for $\Delta x \gg b - a$, the

$f(x_i)$ may be all zeros. Clearly, in this case, all methods based on this data would yield the same approximation. On the other hand, for $\Delta x \approx b - a$, it may be the case that only one of the $f(x_i)$ is non-zero. Assume that $f_k = 1$ and $f_i = 0$, $i \neq k$, and consider the various approximations to the derivative generated by different methods. The *adaptive* stencil method yields approximations to $f'(x)$ whose support consists of exactly two points, regardless of the size of stencils in the selected stencil set. The *fixed* stencil finite difference schemes, however, produce an approximation whose support will contain the same number of points as are in the stencil, i.e., the support (which ideally would remain small) will grow if the stencil size is increased. This is somewhat disheartening, since the primary reason for increasing stencil size is to increase accuracy.

Consider a wave form that is “sharp” at x_k , say one that is represented (near the wave front) as a piecewise constant function in the data $\{f(x_i)\}_{i=1}^N$. For example, $f(x) = H(x - x_k)$ where H is the Heaviside function. The spatial derivative as approximated by a fixed stencil finite difference method would once again have a support that will contain the same number of points as are in the stencil and so it will grow if larger (presumably more accurate) stencils are used. The adaptive stencil finite difference method, on the other hand, will yield an approximate that has a (discrete) support at only one grid point (independent of the size of the stencil).

As the results in Chapter 7 will show, this (in part) results in a relatively small area in which the wave form is smoothed for the adaptive method, and a relatively large area in which the wave form is spread for the fixed stencil methods. Numerical evidence that the adaptive stencil method preserves sharp features will be given in Chapter 7.

CHAPTER 6

Implementation Issues

Issues concerning the computation of an approximation to the solution of initial value problems for first order hyperbolic systems using the adaptive stencil finite difference method are presented. A discussion of the approximation of one-dimensional first order systems is presented in the first section. An in-depth look at the one-dimensional acoustic wave equation is considered in the second section. Boundary conditions and their effect on calculating efficient approximations are considered in the third section. The adaptive stencil selection procedure is a key component to the schemes, and it will be discussed in the fourth section. Even though the method is nonlinear, it is still possible to vectorize the adaptive stencil selection procedure. This speeds the calculations and facilitates the use of multi-processor computers. The final section discusses computational complexity of certain components of the method.

An Algorithm for First Order Systems

The approximation of solutions of systems

$$\partial_t \mathbf{v} + A \partial_s \mathbf{v} = \mathbf{0}, \quad (6.1)$$

where the matrix A is diagonalizable and has elements that are functions of the spatial variable, s , (c.f. systems (3.14) and (3.17)), will be investigated in this section. The first step in approximating the solution to (6.1) is to make a change of variables. Observe that, from Chapter 3, the change of variables

$$\mathbf{w}(s, t) = E^{-1} \mathbf{v}(s, t),$$

where $A = EDE^{-1}$, transforms (6.1) to

$$\partial_t \mathbf{w} + D\partial_s \mathbf{w} = F \mathbf{w}, \quad (6.2)$$

where F is the matrix $D(\partial_s E^{-1})E$. The elements of F are functions of the spatial variable, s , and when a derivative is needed, the adaptive stencil finite difference scheme is used. Note that once the approximation for \mathbf{w} is determined, an approximation for \mathbf{v} is simply $E\mathbf{w}$.

In order to solve (6.2) numerically, a discrete grid is defined and an approximation to the solution of (6.2) restricted to this grid is sought. Assuming the solution to (6.1) is desired on the domain $\{(s, t) | s \in [a, b], t \in [0, T]\}$, then the computational grid is defined to be

$$\{(s_i, t^m) | s_i = a + i\Delta s \in [a, b], i = 0, \dots, N, t^m = m\Delta t \in [0, T], m = 0, \dots, M\}.$$

This process defines an equi-spaced computational grid. Note that it is necessary that condition (5.22) is met when choosing the size of Δs and Δt . Let W^m denote the mesh function, whose components are the approximations to $\mathbf{w}_i^m = \mathbf{w}(s_i, t^m)$, $i = 0 \dots, N$. The matrices D and F do not depend on time, so the values of $D_{ii} = D(s_i)$ and $F_i = F(s_i)$ for each i may be determined a priori.

Initially (at time $t_0 = 0$), W^0 is determined by the initial data, $E^{-1}\mathbf{v}_i^0$. Once this is determined, the process of time stepping may begin. From (5.17),

$$W_i^{m+1} = W_i^m + \Delta t(F_i - D_i\partial_s)W_i^m + \frac{1}{2}\Delta t^2(F_i - D_i\partial_s)^2W_i^m + \dots \quad (6.3)$$

for $i = 1, \dots, N$. The adaptive stencil finite difference method (see Chapter 5) is used to approximate the terms involving ∂_s , and determines the spatial accuracy of the method. Implementation of the adaptive stencil finite difference scheme is discussed in further detail in the third section of this chapter. The number of terms taken on

the right in (6.3) determines the temporal accuracy of the method (see the seventh section of Chapter 5).

To approximate the solution to the one-dimensional problem (3.6) using the above algorithm, the system form (3.7) of this problem is diagonalized using the matrices in Table 1. Equation (6.3) is used (with $s = x$) to determine the approximation at times $t > 0$. Note that in this specific case, the system (6.2) is the coupled system (3.16). This particular system occurs when considering multidimensional problems as well, and will be discussed in greater detail in the next section.

To solve the two-dimensional problem (3.8), the equivalent system (3.9) is split using one of the Strang splittings (4.5) or (4.7), or an extended splitting such as (4.13). Each splitting requires an approximation to repeated applications of exponential operators like those in (4.6). Each application of an exponential operator involving an x derivative requires the approximation of systems of the type (4.10). Similarly, applications of exponentials involving a y derivative requires the approximation of the systems of the type (4.9), as described in the discussion of (4.8)-(4.10). Each of the systems (4.10) and (4.9) may be put in the form of (6.1), and the above algorithm may be used to approximate solutions. Note that the computational grid $\{(s_i, t^m)\}$ is not the same when considering (4.10) as it is when considering (4.9). The grid function W_m , D , and F must be defined at each (x_i, y_j) , $i = 0, \dots, N_x$, $j = 0, \dots, N_y$ in the computational domain. When considering (4.10), the independent variable, y , is held constant and, rather than evaluating the single system (6.3), the N_y systems

$$W_{i,j}^{m+1} = W_{i,j}^m + \Delta t(F_{i,j} - D_{i,j}\partial_x)W_{i,j}^m + \frac{1}{2}\Delta t^2(F_{i,j} - D_{i,j}\partial_x)^2W_{i,j}^m + \dots, \quad j = 0, \dots, N_y \quad (6.4)$$

must be evaluated. Similarly, when considering (4.9), the independent variable, x , is

held constant and rather than evaluating the single system (6.3), the N_x systems

$$W_{i,j}^{m+1} = W_{i,j}^m + \Delta t (F_{i,j} - D_{i,j} \partial_y) W_{i,j}^m + \frac{1}{2} \Delta t^2 (F_{i,j} - D_{i,j} \partial_y)^2 W_{i,j}^m + \dots, \quad i = 0, \dots, N_x \quad (6.5)$$

must be evaluated. For systems involving more than two spatial variables, the process is extended similarly.

Note that if the components of the n -vector \mathbf{w} are $w^{(i)}(s, t)$, $i = 1, \dots, n$, then the components of the system that appears in (6.2) are

$$\partial_t w^{(i)} + \lambda_i \partial_s w^{(i)} = F(i, :) \mathbf{w} \quad i = 1, \dots, n, \quad (6.6)$$

where $F(i, :)$ is the i^{th} row of the matrix F and where λ_i is the (i, i) entry of the diagonal matrix, D .

The adaptive finite difference selection procedure described in the section containing Figure 2 is a key component of the approximation of the solution of (6.6) and will be discussed in more detail in the next section. For notational purposes, define $(\partial^n u)$ to be the $O(\Delta s^n)$ adaptive stencil approximation to $\partial_s u$.

The CIR (5.7) approximation of each $w^{(i)}$ at the point s_k and time $t^{m+1} = t^m + \Delta t$ is then given by

$$w_k^{(i),(m+1)} = w_k^{(i),m} - \Delta t \left(\lambda_i (\partial^1 w^{(i),m})_k + F(i, :)_k \mathbf{w}_k^m \right). \quad (6.7)$$

This is equivalent to the system (5.17) if only the first two terms are taken on the right and the derivative operator is replaced with the adaptive stencil approximation. In the systems derived in Chapter 3, an occurrence of a zero on the diagonal in D implies that $\lambda_i = 0$ and it also implies that the right hand side, $F(i, :)\mathbf{w}$, is zero, so (6.7) reduces to

$$w_k^{(i),(m+1)} = w_k^{(i),m}, \quad (6.8)$$

and no further computations are required.

Numerical Solutions of the Acoustic Wave Equation

In this section, an algorithm for solving initial value problems for the one-dimensional acoustic wave equation is presented. Consider

$$\partial_t^2 u - c^2 \partial_x^2 u = 0, \quad -\infty < x < \infty, t > 0 \quad (6.9)$$

with initial data

$$u(x, 0) = f(x), \quad \partial_t u(x, 0) = 0. \quad (6.10)$$

Further, assume that $c(x)$ is constant outside of some finite interval, and that $f(x)$ has finite support that is contained in this interval. The computational grid may then be chosen such that c is constant outside the computational grid. This second order problem is rewritten as the first order hyperbolic system (3.7). The initial conditions imply that, initially, $\mathbf{v} = [f'(x), 0]^T$. Using the matrices in Table 1 and Table 2, the linear operator in (3.7) is rewritten in terms of $\mathbf{w} = E^{-1}\mathbf{v}$, yielding the operator (3.3).

Numerically, it is not necessary to change variables. The appropriate combination of the original variables may be used, saving computation time, as well as storage space. To this end, note that

$$\mathbf{w} = \begin{bmatrix} \partial_x u + \frac{1}{c} \partial_t u \\ \partial_x u - \frac{1}{c} \partial_t u \end{bmatrix}$$

Also, if the matrix $D(\partial_x E^{-1})E$ in (3.16) is applied to this representation of \mathbf{w} ,

$$D(\partial_x E^{-1})E\mathbf{w} = \begin{bmatrix} -\frac{c'}{c} \partial_t u \\ -\frac{c'}{c} \partial_t u \end{bmatrix}$$

so that

$$\begin{bmatrix} \hat{\partial}_t(\partial_x u + \frac{1}{c} \partial_t u) + c \partial_x^{n+}(\partial_x u + \frac{1}{c} \partial_t u) \\ \hat{\partial}_t(\partial_x u - \frac{1}{c} \partial_t u) - c \partial_x^{n-}(\partial_x u - \frac{1}{c} \partial_t u) \end{bmatrix} = \begin{bmatrix} -\frac{c'}{c} \partial_t u \\ -\frac{c'}{c} \partial_t u \end{bmatrix}. \quad (6.11)$$

Again, the c' on the right hand side of the (6.11) is determined by the adaptive stencil finite difference scheme (the final section of Chapter 5 showed that the scheme will yield an appropriate approximation even if c is only piecewise differentiable). The operators $\hat{\partial}_t$, ∂_x^{+n} , and ∂_x^{-n} , are the approximation operators to the temporal and spatial derivatives, respectively, that were discussed in the previous section (the $+$ and $-$ indicate positive and negative wave velocities). The CIR method in (6.7) can be reduced to a simple time stepping scheme. Let

$$a_j^m = \partial_x^{+1}(\partial_x u_j^m + \frac{1}{c} \partial_t u_j^m) \quad \text{and} \quad (6.12)$$

$$b_j^m = \partial_x^{-1}(\partial_x u_j^m - \frac{1}{c} \partial_t u_j^m). \quad (6.13)$$

Then solving (6.11) for $\partial_x u$ and $\partial_t u$ at time level t^{m+1} in terms of information at time level t^m yields,

$$\partial_x u_j^{m+1} = \partial_x u_j^m + r_j(b_j^m - a_j^m) - \Delta t \frac{c_j'^m}{c_j^m} \partial_t u_j^m \quad (6.14)$$

$$\partial_t u_j^{m+1} = \partial_t u_j^m + r_j(b_j^m + a_j^m), \quad (6.15)$$

where $r_j = \frac{c_j \Delta t}{2}$.

This is a single step method; only information at the previous time step is needed to determine the approximate solution at the next time step. A numerical scheme based on this method simply loops on (6.12)-(6.14), starting with the initial data $\partial_x u_j^0 = f'(x_j)$, $\partial_t u_j^0 = 0$ for each j . The ratio of Δx to Δt must be chosen to satisfy the CFL condition (5.21) with $a(x) = c(x)$.

Note that the spatial boundaries were not mentioned here. The problem was posed on the entire real line. Boundary conditions called *absorbing boundary conditions* [4] may be imposed at the computational boundaries. These boundary conditions are imposed so that the values of the approximate solution on the computational domain will approximate the restriction of the true solution to the subset of the true

domain that contains the computational domain. It must also be assumed that the wave speed in this example, $c(x)$, must be constant outside the computational domain. If c were not constant, then a reflected wave would eventually return to the computational boundary, violating a basic assumption that no disturbances enter the computational domain from the exterior.

Boundary Conditions

Consider limiting the computational domain for the one-dimensional problem,

$$\partial_t u + a(x)\partial_x u = 0 \quad -\infty < x < \infty,$$

$$u(x, 0) = f(x).$$

Under the assumption that the initial wave form, $f(x)$, has finite support contained in the computational boundaries, and the assumption that $a(x)$ does not change sign, it is possible to absorb the wave form at the downwind boundary. Near the downwind boundary, the stencil is adapted so that only information from the interior of the computational domain is used.

If $a > 0$, then the wave is traveling to the right. At the left endpoint in the computational domain, $\partial_x u$ is set to 0. (i.e., there is no energy transmitted into the computational domain from the 'region' to the left of the computational domain.) Near the left boundary, the finite difference stencil may still need to be modified so that the information used is contained in the computational domain. At the point to the right of the left endpoint, the upwind stencil ($\circ \circ \circ \dots \circ$) is applied. At the next point to the right, there are two upwind stencils, and the adaptive stencil selection procedure is used to select the appropriate one.

At the right endpoint, the **totally** upwinding stencil is used. Thus, at the right endpoint, the stencil ($\circ \dots \circ \circ \circ$) is applied. At the point to the left of the endpoint,

there is a choice of two possible stencils, and, again, the adaptive procedure is used to make the selection.

It is possible to make these boundary modifications an automated part of the adaptive stencil selection procedure. Suppose $\vec{v} = \{v_i\}_{i=1}^N$ is a one-dimensional vector of data. If the adaptive procedure were then applied, stencils that require data outside of the computational domain may be selected. If, however, the same procedure is applied to

$$[\infty, v_1, \dots, v_N, \infty],$$

then the procedure will 'choose' not to use the ∞ values; hence, no other data outside the domain would be chosen.

The first order system resulting from the one-dimensional acoustic wave equation was considered in the previous section. The diagonalized form (3.16) decouples in regions of the domain in which a is constant. The natural requirement that a is constant outside of the computational boundaries was discussed above. So that the waves propagated by the operator (3.16) may be absorbed, it must be required that a is not only constant outside the computational boundary, but also *near* the computational boundary. One stencil length is enough to ensure that none of the energy of the out-going wave will be transferred to the in-going wave. With this added assumption, the absorbing boundary conditions will be exact for the method (i.e., there will be *no* reflection from the computational boundary).

Other boundary conditions are easily handled. 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 instance, in the 1-D case, $\partial_x u$ may be set equal to zero at the end points at each time step (recall that $\partial_x u$ and $\partial_t u$, rather than u , are carried along at each time step). Standard methods require additional, often quite complicated, difference approximations near boundaries in order to ensure that the

resulting approximation will have the correct approximate derivatives.

Efficient Stencil Selection

In this section, an efficient implementation of the adaptive finite difference stencil procedure is outlined. As discussed above, there is only one stencil of a given size that may be applied at the computational boundaries. Near a computational boundary, the adaptive stencil selection procedure outlined in Chapter 5 must be modified so that only stencils which utilize data from within the computational domain are chosen. The modifications to the "near-boundary" selection procedure will effect the realized boundary conditions. In this section, it is assumed that absorbing boundary conditions are sought.

In order to determine an adaptive stencil approximation to the derivative of a discrete mesh function, f , certain information is required. Besides the function values ($\{f_i\}_{i=1}^N$), the a priori information required consists of the spatial step size (Δx), the order of the desired approximation, and which direction is the upwind direction. Recall from Figure 2 that upwinding completely determines the first order stencil.

To ensure that only function values from the given data set will be required by the selection procedure, large values are attached to each end of the set, $\{f_i\}_{i=1}^N$. This implies that the working data set consists of $\{f_i\}_{i=0}^{N+1}$ where $f_0 = f_{N+1} = \infty$. In practice, ∞ is taken to be a large constant (relative to the data set). Note that, by the discussion in the previous section, this will also result in absorbing boundary conditions.

In two-dimensional problems, the dimensional splitting implies that, at each step in the approximation, one of the independent variables is held constant, and

the procedure outlined here need not be modified (as long as Δx is replaced by the appropriate dimensional spacing).

Let $\{F_i\}$ be the set that contains approximations of the derivatives of the set $\{f_i\}$, so the $F_i \approx f'(x_i)$. If a first order approximation is sought and the upwind direction is to the left (f_i being left of f_{i+1}), then the adaptive approximation, F_i , is simply

$$F = \frac{[0, \Delta f]}{\Delta x},$$

where Δf is the vector whose entries are $\Delta f_i = f_{i+1} - f_i$, $i = 0, \dots, N$. The notation $[a\vec{v}]$ denotes the vector, or set, obtained by appending the scalar a to the left end of the vector, or set, \vec{v} . On the other hand, if the upwind direction is to the right, then

$$F = \frac{[\Delta f, 0]}{\Delta x}.$$

These choices of F correspond to the CIR method presented in (5.8).

In determining second order approximations, there are three possible stencils for each point (see Figure 2). Define

$$d0 = \Delta f, \quad d1 = \Delta d0,$$

and \vec{q} to be the vector whose i^{th} entry is

$$q_i = \begin{cases} 1, & \text{if } |d1_i| \leq |d1_{i+1}|, \\ 0, & \text{otherwise.} \end{cases}$$

The values of \vec{q} and the upwind direction is the only information that is required to select the finite difference stencil that must be applied to the data in order to determine F_i .

If the upwind direction is to the left, then $F_1 = 0$. If $q_i = 1$, then F_{i+1} for $i > 0$ is the value determined by applying the stencil $\circ \circ \circ$ to the data $\{f_j\}_{j=i-2}^i$. If $q_i = 0$, F_{i+1} is the value determined by applying the stencil $\circ \circ \circ$ to $\{f_j\}_{j=i-1}^{i+1}$. On

the other hand, if right is the upwind direction, then $F_N = 0$. If $q_i = 1$, then F_i for $i < N$ is the value determined by applying the stencil $\circ \bullet \circ$ to the data $\{f_j\}_{j=i-1}^{i+1}$. If $q_i = 0$, F_i is the value determined by applying the stencil $\bullet \circ \circ$ to $\{f_j\}_{j=i}^{i+2}$.

Note that it appears possible that f_0 and f_{N+1} may be used to determine values for F_i . Placing the ∞ values into the data, as indicated above, will force the values of q to be such that the *inward* looking stencils are chosen. Also note that the stencil $\circ \bullet \circ$ may be selected in two ways. This corresponds to the fact that there are two paths leading to this stencil in the tree in Figure 2. For higher order approximations, this procedure is simply extended to incorporate each of the decisions that are implied by the tree in Figure 2, incorporating the growing number of possible paths into the procedure.

This procedure yields approximations to either the left or right derivatives at each point, depending on whether the upwind direction is to the right or left. At the upwind boundary, the derivative is assumed to be zero, implying no change at the boundary. Other boundary conditions may be implemented. For example, suppose f is periodic, and that the appropriate boundary conditions are needed. Further, suppose that $\{f_i\}_{i=1}^N$ is the discrete data for which an approximate derivative must be determined. In this situation, the method should adapt at every point, including the areas near the computational boundary. The data set must be modified so that the selection procedure has access to the pertinent data. The data set

$$[f_{N-2}, f_{N-1}, f_1, \dots, f_N, f_2, f_3]$$

contains the information that is needed to determine the adaptive finite difference approximation to the derivative at each point in the computational domain when periodic boundary conditions are imposed and three point stencils are utilized. More points must be added to the data set in the obvious way if larger stencils are to be

utilized.

Computational Complexity

Increasing temporal accuracy requires the approximation of terms of the form

$$(\partial_t)^k \mathbf{w} = (F - D \partial_s)^k \mathbf{w},$$

see (5.17). Since, in general, F , D and ∂_s do not commute,

$$(F - D \partial_s)^2 \mathbf{w} = F^2 \mathbf{w} - F D (\partial_s \mathbf{w}) - D \partial_s (F \mathbf{w}) + D \partial_s (D (\partial_s \mathbf{w})).$$

Hence, additional applications of the approximation to ∂_s are needed. Note that since the approximation to ∂_s is nonlinear and data dependent, more than one call to the approximation procedure will be required. The first and the last ∂_s on the right hand side both act on the same data; therefore, they do not require separate calls to the adaptive approximation procedure. They are also the same as the operator that occurred in the first order term. However, if the elements of D are non-constant, then $\partial_s D \mathbf{w} = \partial_s (D \mathbf{w})$ is not the same as $D \partial_s \mathbf{w}$. Moreover, $\partial_s D \mathbf{w} = \partial_s (D \mathbf{w})$ is not $D \partial_s \mathbf{w} + (\partial_s D) \mathbf{w}$. Although this implies more computations, as the motivating example presented above pointed out, the finite difference stencil should be adapted depending on the data in order to avoid certain errors. And, since \mathbf{w} differs from $D \mathbf{w}$, the approximation to the operator ∂_s used to determine $\partial_s \mathbf{w}$ should not, in general, be the same as the approximation to the operator ∂_s used to determine $\partial_s (D \mathbf{w})$.

When approximating $\partial_s f$, 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 associated trees, and, therefore, the elements $D^n(f)_i$, (see Figure 2) 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. As the previous section shows, this selection process may be vectorized, implying that very efficient computer code may be formulated.

CHAPTER 7

Numerical Results

This chapter contains a number of numerical examples. Included are approximations to first order scalar equations in one spatial dimension and first order systems in one and two spatial dimensions. Many of the examples include either discontinuous initial data or discontinuous wave speed parameters.

First Order, One Space Dimension

Consider the problem of finding $u(x, t)$ satisfying

$$\partial_t u + \partial_x u = 0, \quad -\infty < x < \infty \quad t > 0$$

$$u(x, 0) = f(x).$$

Apply a second order adaptive spatial stencil and a first order time marching scheme to this problem by taking $W = u$, $D = 1$, and $F = 0$ in (5.13). If the first order adaptive stencil operator is replaced with the second order adaptive stencil operator, this is equivalent to taking $w^{(2)} = u$ and $\lambda_i = 1$ in the CIR approximation, (6.7). Letting $u^0 = f(x)$, an approximation to u may be obtained at any time $t = k\Delta t$ by performing k iterations of the scheme,

$$u_i^{m+1} = u_i^m - \Delta t \partial_x^{+2}(u_i^m), \quad i = 1, \dots, N, \quad (7.1)$$

where $\partial_x^{+2}(u)$ is the approximation to the derivative obtained by using a second order adaptive stencil. The second order stencil is selected by the procedure outlined in Chapter 5 when left is the upwind direction. If

$$\frac{\Delta t}{\Delta x} = \frac{2}{3}, \quad (7.2)$$

and the initial data, $f(x)$, is taken to be the piecewise constant function shown on the left in Figure 4, then it is easily shown that this method is exact. Note that the ratio in (7.2) satisfies the CFL condition (5.19).

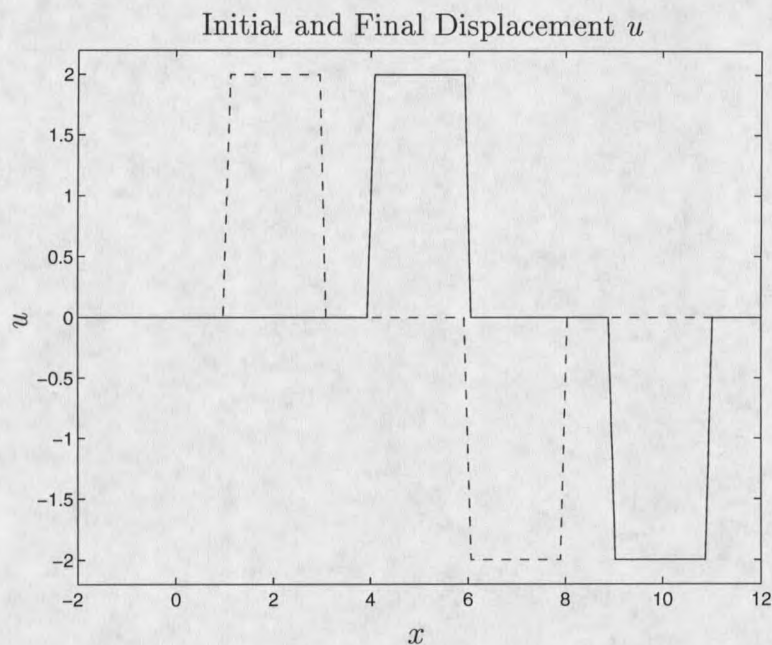


Figure 4: Initial data and (exact) numerical approximation at time $t = 3$.

Let \vec{u}^m be the vector whose i^{th} entry is u_i^m and let $M^{(m)}$ be the $N \times N$ matrix whose values correspond to those of the operator ∂_x^{+2} . In other words, define $M^{(m)}$ so that the vector $M^{(m)}\vec{u}^m$ agrees with the vector whose i^{th} entry is $\partial_x^{+2}(u_i^m)$ for $i = 1, \dots, N$. Due to the nature of the adaptive stencil selection procedure, the matrix $M^{(m)}$ will change from one time step to the next; hence, the superscript. With this notation, (7.1) can be rewritten as

$$u_i^{m+1} = (I - \Delta t M^{(m)})u^m, \quad (7.3)$$

and if the matrix $P^{(m)}$ is defined to be $I - \Delta t M^{(m)}$, then $P^{(m)}$ is adaptive stencil finite difference approximation of the propagator, $e^{-\Delta t \partial_x}$, which takes \vec{u} from time t^m to time t^{m+1} .

If N is taken to be 100, the non-zero entries of the matrix $P^{(m)}$ for a particular m are shown in Figure 5. The adaptive nature of the method may be seen by noting the differences in the locations of the stencils relative to the diagonal. Recall that the adaptive procedure (given the upwind direction) chooses from among two stencils at each point, and these two stencils may be observed in the figure. The points at which the stencil changes correspond directly with the discontinuities in the solution, \bar{u}^m , at the point in time for which the matrix was recorded.

For fixed stencil methods, the matrices $M = M^{(m)}$ and $P = P^{(m)}$ are data-independent, and thus, will be fixed. By the Lax-Equivalence Theorem, a necessary condition for the stability of a method based on (7.3) is that the norm of the matrix P be less than one. The norm of $P^{(m)}$ illustrated in the Figure 5 example is 1.8385, which violates the necessary condition of this theorem. However, this theorem also assumes that P is fixed, so that the \bar{u}^m may be determined by finding

$$u^m = P u^{m-1} = P(P(u^{m-2})) = P^m u^0 = P^m f.$$

Note that if the norm of P is greater than one, the norm of the solution will tend to infinity as the m increases. Due to the adaptive nature of the adaptive stencil procedure, the matrices $P^{(m)}$ may not be determined a priori, and thus, no direct relationship between the initial data and the solution at a given time may be determined a priori. The matrices $P^{(m)}$ may not be determined a priori, but for a given problem, they may be recorded for the purpose of analysis.

In Figure 6, the non-zero entries of the solution matrix,

$$\hat{P} = \prod_{m=1}^k P^{(m)},$$

are shown, where k is the number of time steps needed to arrive at the solution shown in Figure 4. The norm of the matrix \hat{P} is 43568. Since \hat{P} takes the initial data to the exact solution, it appears that standard linear analysis is not applicable

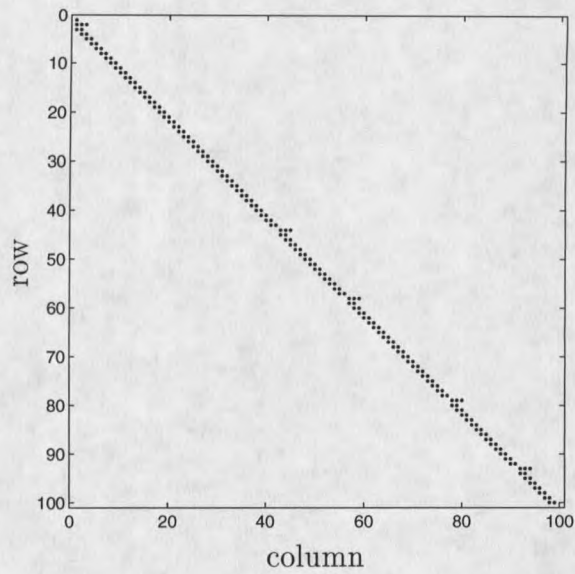


Figure 5: Non-zero entries of the matrix M^m for a certain m .

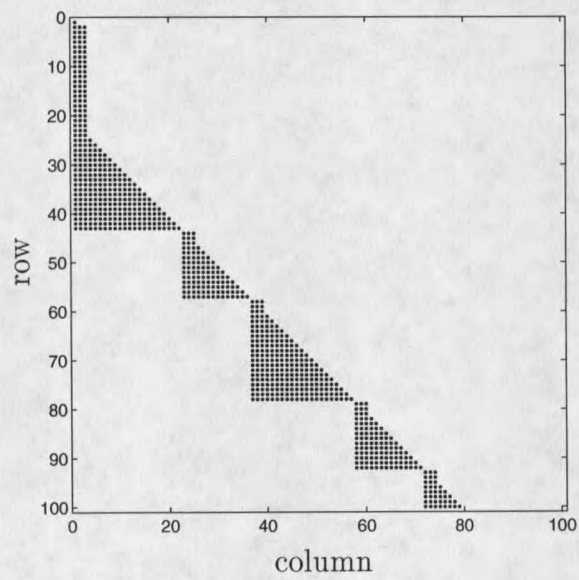


Figure 6: Non-zero entries of \hat{P} .

(one would expect via stability analysis that this matrix would, since it is generated by an exact method, have a norm near 1). It is also interesting to note the effects that adaptivity has on the matrix \hat{P} . The pattern of non-zeros elements is shown in Figure 6. Note that the M , corresponding to a second order centered finite difference method, would be tri-diagonal, which means that P^k would have bandwidth $2k + 1$ and, thus, eventually produce a full matrix.

The above method will not be exact if (7.2) does not hold. However, the method is still second order accurate in space. Figure 7 shows the numerical solution to the above problem when $\frac{\Delta t}{\Delta x} = \frac{1}{3}$. Note that there is some smoothing of the wave form, but that no spurious oscillations are present.

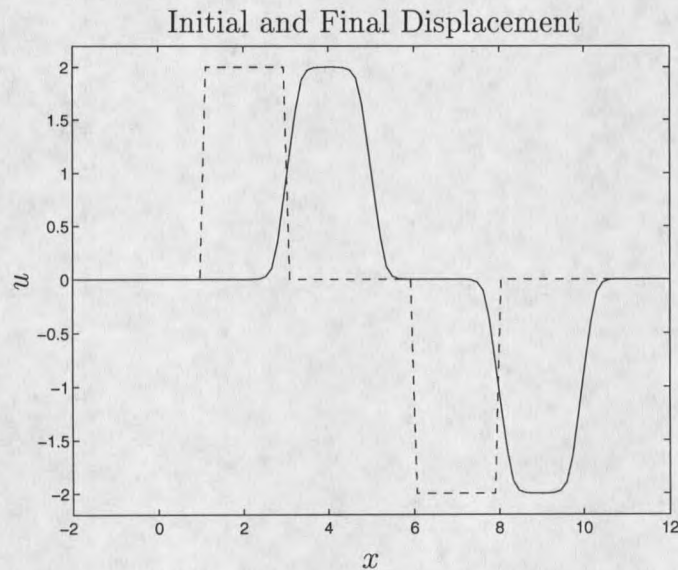


Figure 7: Initial and final displacement for an $O(\Delta x^2)$ method.

Since linear stability analysis does not apply, numerical evidence of stability is presented. If the noise in Figure 8 is added to the initial data of the above example (again with $\frac{\Delta t}{\Delta x} = \frac{1}{3}$) the method yields the results in Figure 9. The noise is random and has a maximum range that is equal to 2.5 percent of the range of the original

initial data. Note that the solution does not exhibit oscillations and maintains the overall wave form of the original data.

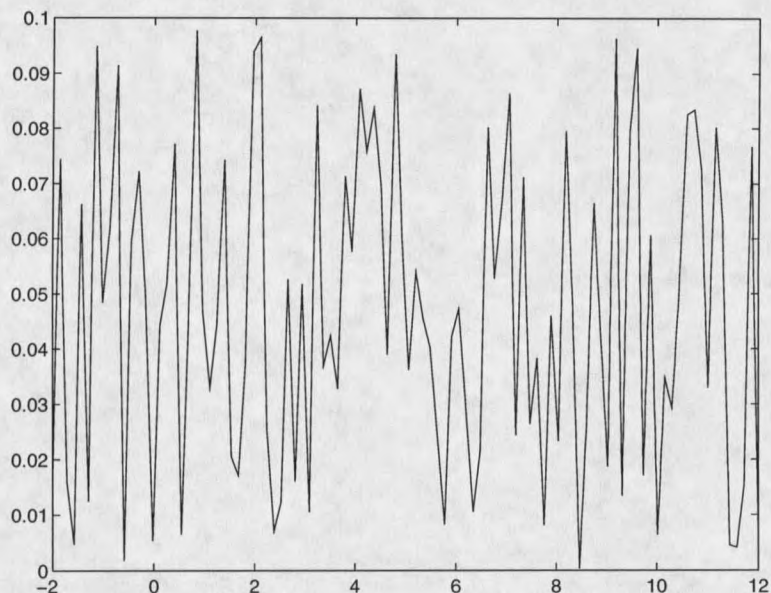


Figure 8: Random noise.

Fixed stencil methods applied to the above problem perform poorly, due to the non-smooth initial data. The approximation utilizing the second order centered stencil with the noise free initial data is shown in Figure 10. The approximation shown is for time $t = .5$. The approximation for $t = 3$ (as in the above examples) contains very large oscillations, and is not shown.

Increasing the size of the stencil in a fixed stencil finite difference scheme on similar initial data affects the solution very little. Figure 11 shows the results when a five point centered stencil is applied to the noise free initial data. On the other hand, the adaptive stencil finite difference method yields greater accuracy when stencil size is increased. In Figure 12 the results for stencil size 4 is shown. Note how the amount of smoothing is reduced, compared to the solution in Figure 7.

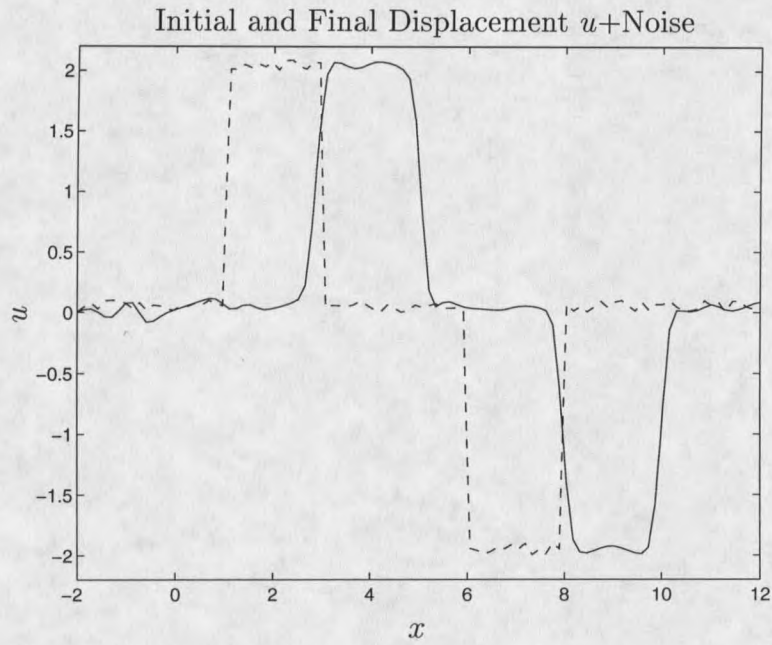


Figure 9: Initial data (including noise) and numerical solution at time $t = 3$

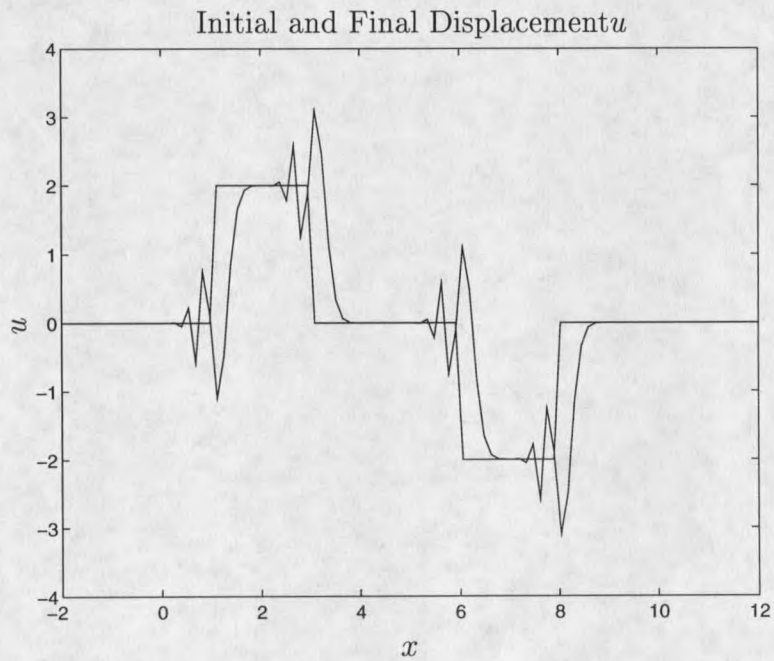


Figure 10: Initial data and 3-point fixed stencil solution at time $t = .5$

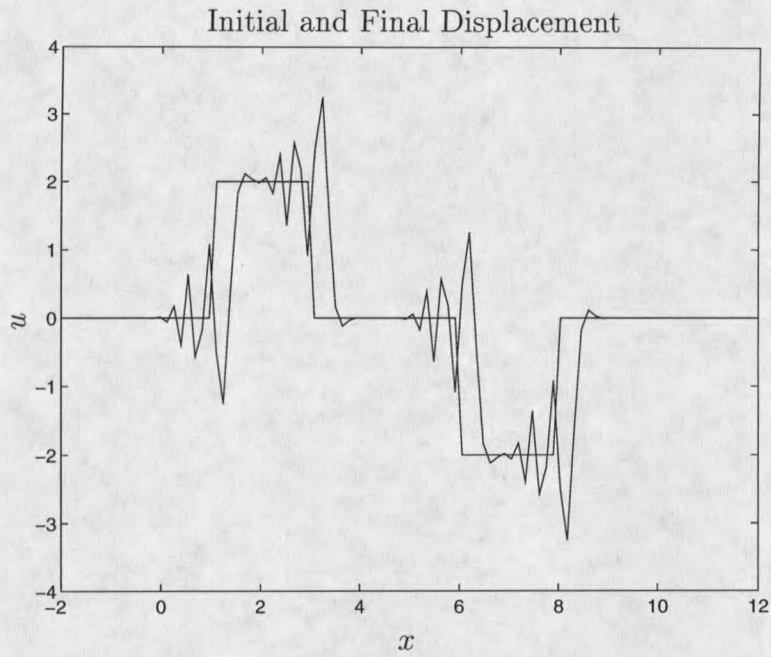


Figure 11: Initial data and 5-point fixed stencil solution at time $t = .5$

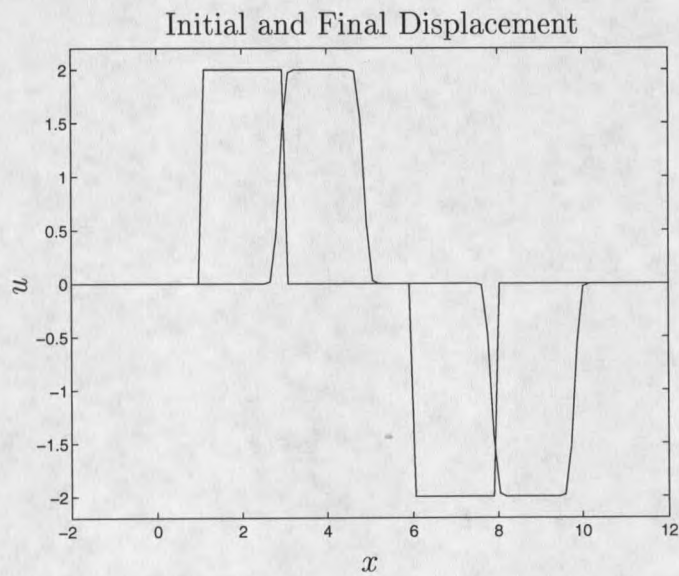


Figure 12: Initial and final displacement for an $O(\Delta x^3)$ method.

Systems in One Space Dimension

This section contains two examples that illustrate the attributes of the adaptive stencil method as applied to first order linear systems. The procedure is outlined in Chapter 6, yielding second order temporal accuracy and, unless otherwise stated, an $O(\Delta x^2)$ adaptive spatial stencil (level 3 in the tree in Figure 2) is used, and Δt is taken to be $\Delta x/(2 \max_x c(x))$.

The first example is the one-dimensional initial value problem (IVP) for the scalar wave equation

$$\partial_t^2 u - \partial_x^2 u = 0, \quad -\infty < x < \infty, \quad t > 0, \quad (7.4)$$

$$u(x, 0) = f(x), \quad -\infty < x < \infty,$$

$$\partial_t u(x, 0) = 0, \quad -\infty < x < \infty,$$

where $f(x)$ is the piecewise linear function on the left in Figure 13. The piecewise constant derivative, $f'(x)$, is shown on the right.

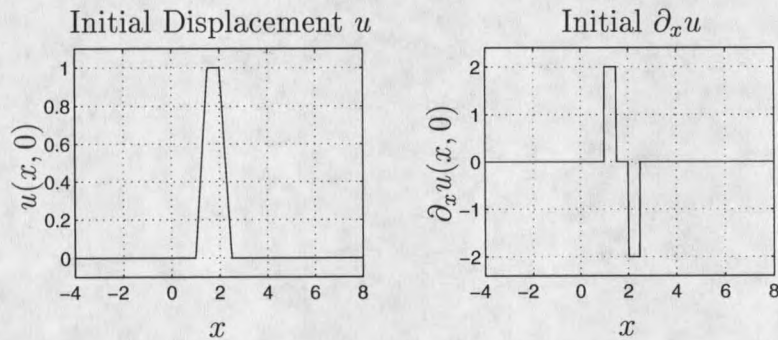


Figure 13: Initial data for a one-dimensional, homogeneous IVP.

Recall that the method approximates the vector of derivatives $\vec{v} = [\partial_x u, \partial_t u]^T$.

The D'Lambert solution [14] of (7.4) is

$$u(x, t) = \frac{1}{2}f(x - t) + \frac{1}{2}f(x + t), \quad (7.5)$$

and so

$$\partial_x u(x, t) = \frac{1}{2}f'(x - t) + \frac{1}{2}f'(x + t), \quad (7.6)$$

i.e., $\partial_x u$ is the sum of translates of components of the piecewise constant function shown on the right in Figure 13.

In Figure 14, the numerical approximation of $\partial_x u$ at various times is presented. In Figure 15, results obtained using approximations that are $O(\Delta x^n)$ for $n = 1, 2$ and 3 , are presented. For clarity, only the subinterval $3 \leq x \leq 6$ at time $t = 3$ is shown. Note that the approximations become increasingly sharp as n increases, and that none exhibit spurious oscillations.

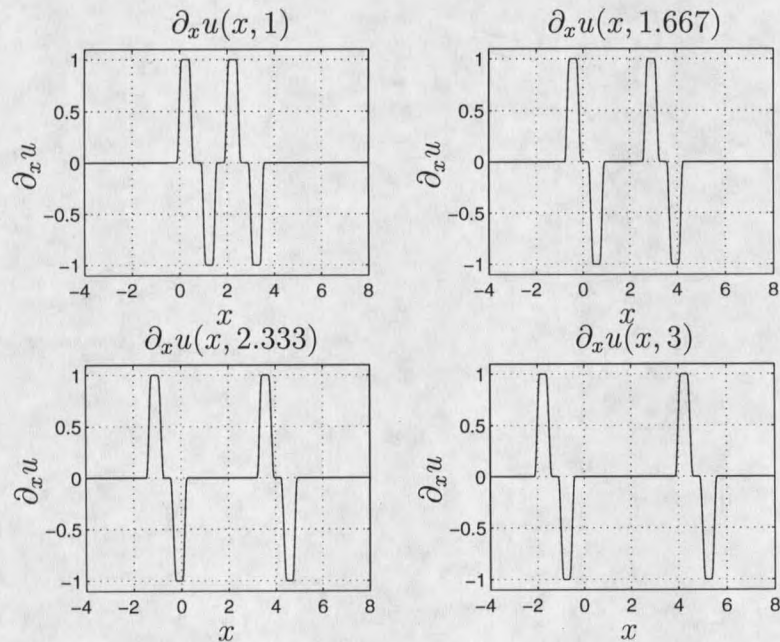


Figure 14: Adaptive stencil solutions for a one-dimensional, homogeneous, scalar wave IVP at times $t = 1, \frac{5}{3}, \frac{7}{3}$, and 3 .

A fixed stencil finite difference approximation to (7.4) is presented in Figure 16. The method consists of formulating second order approximations to the second

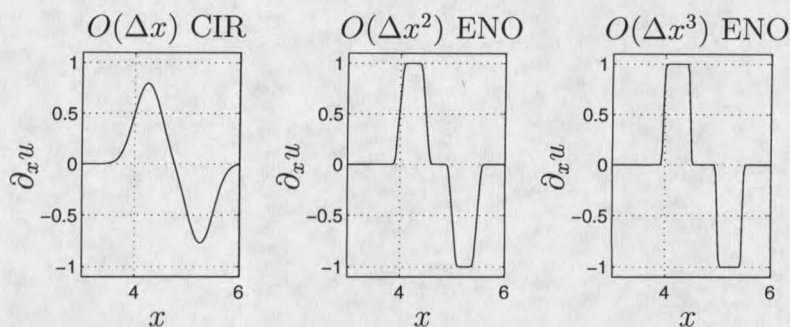


Figure 15: Effects of stencil size on adaptive stencil solutions.

derivatives in (7.4),

$$\frac{u_j^{m+1} - 2u_j^m + u_j^{m-1}}{\Delta t^2} - \frac{u_{j+1}^m - 2u_j^m + u_{j-1}^m}{\Delta x^2} = 0, \quad (7.7)$$

and is standard [10]. The Δt and Δx are the same as in the previous approximations to (7.4), and the CFL condition was met. On the left in Figure 16 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 . Approximating the derivative, $\partial_x u$, greatly magnifies the oscillations.

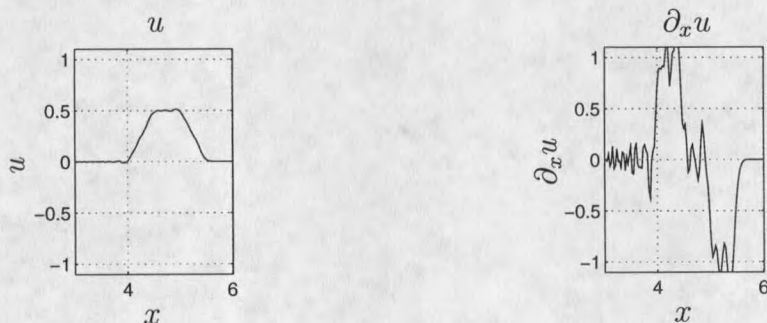


Figure 16: Standard finite difference solution to a one-dimensional, constant coefficient, homogeneous, scalar wave equation.

The following test case was taken from a paper [7] in the journal *Geophysics*.

Consider the IVP

$$\partial_t^2 u - c(x)^2 \partial_x^2 u = f(t) \delta(x - x_0), \quad -\infty < x < \infty, \quad t > 0, \quad (7.8)$$

$$u(x, 0) = \partial_t u(x, 0) = 0, \quad -\infty < x < \infty,$$

with the wave speed, $c(x)$, the forcing function, $f(t)$, and the source location, x_0 , given in Figure 17 below. Note that $c(x)$ is piecewise constant with a jump discontinuity at position $x = 5203$ meters, where the wave speed doubles. Figure 18 shows the 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 [7], high order (up to 10^{th} order in space and 4^{th} in time) difference approximation schemes were advocated in order 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. In contrast, the second order adaptive stencil approach is relatively easy to implement, and gives results which compare very favorably with those in [7].

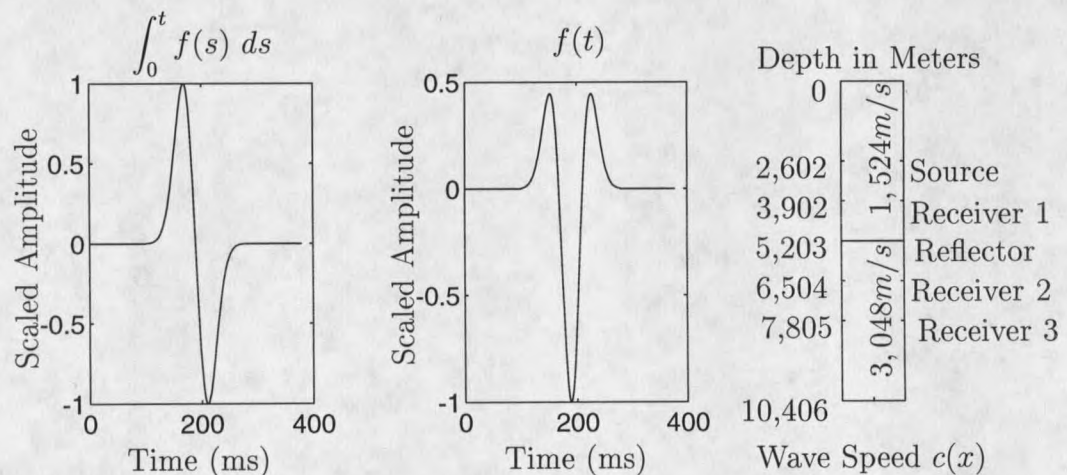


Figure 17: The forcing function, $f(t)$, and discontinuous wave speed, $c(x)$.

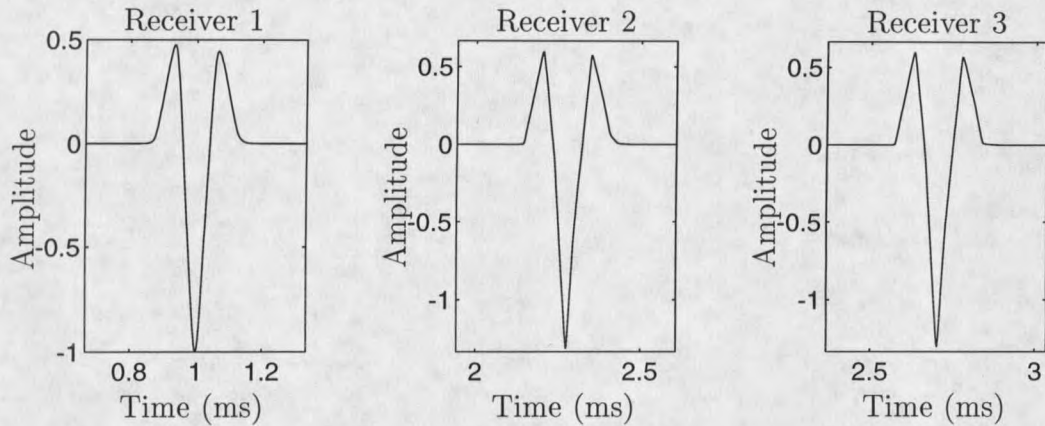


Figure 18: Time traces at various spatial locations obtained using a second order adaptive stencil method.

Numerical Results in Two Space Dimensions

In two spatial dimensions, closed form solutions to the acoustic wave equation problems are known only in certain cases. In Chapter 2, a family of solutions was derived for the problem

$$\partial_t^2 u - \partial_x^2 u + \partial_z^2 u = 0, \quad (7.9)$$

$$u(x, z, 0) = f_\alpha(x, z), \quad \partial_t u(x, z, 0) = 0, \quad (7.10)$$

where $f_\alpha(x, z)$ is given in (2.12). For $\alpha = .25$, $f_\alpha(x, z)$ is shown in Figure 19.

A numerical approximation of the solution of this problem was found using the Strang splitting (4.7) and second order adaptive spatial derivatives. The spatial mesh size was reduced a number of times and comparisons with the exact solution were made using the energy norm [23].

The results of the comparison are shown in Figure 20, and the slope of the line is approximately 2 for small values of Δx , implying $O(\Delta x^2)$ convergence of the method. So as to avoid the issue of boundary conditions, the energy norm was taken on a subset of the computational domain which could not be affected by the treatment

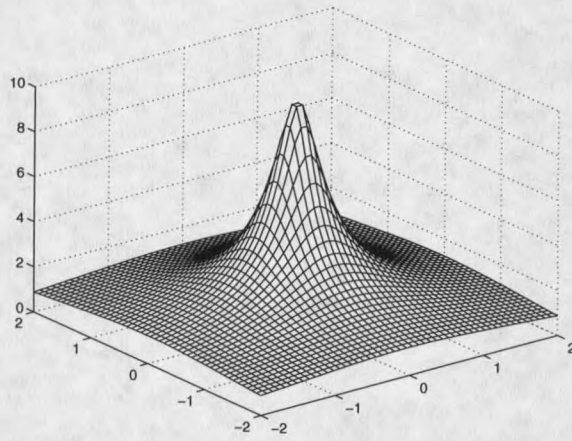


Figure 19: Initial data for a two-dimensional problem.

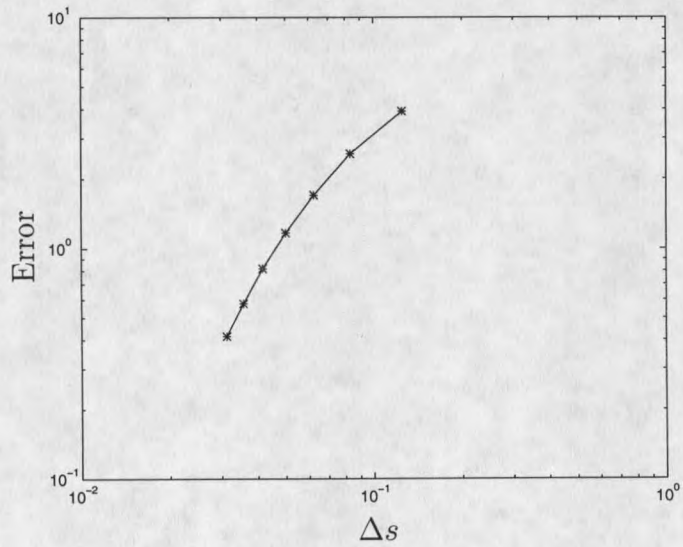


Figure 20: Energy norm of the error versus spatial step size.

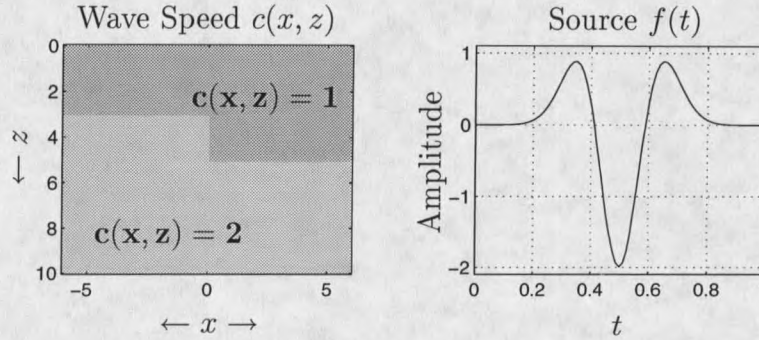


Figure 21: Wave speed and forcing function for a two-dimensional discontinuous coefficient problem.

near the boundaries. The initial data in Figure 19 is smooth, and since the wave speed is constant, the media is homogeneous, so this example does not test the ability of the method to handle discontinuities, but it does show the convergence of the Strang splitting when used in conjunction with ENO. The examples presented below attest to the method's ability to handle discontinuities in the wave speed.

Each of the next two examples illustrate the method when a second order accurate Strang splitting (4.7) and a second order adaptive spatial stencil is used. In both examples, absorbing boundary conditions are employed, $\Delta x = \Delta z$, and Δt is taken to be $\Delta x / (2 \max_{(x,z)} c(x, z))$.

Consider

$$\begin{aligned} \partial_t^2 u - c^2 (\partial_x^2 u + \partial_z^2 u) &= f(t) \delta(x) \delta(z), \\ u = \partial_t u &= 0 \quad \text{at } t = 0, \end{aligned} \quad (7.11)$$

where $c(x, z)$ is the piecewise constant function shown on the left in Figure 21 and $f(t)$ is the Ricker wavelet commonly used in geophysical problems [7], shown on the right in Figure 21.

In the first frame the wave propagates from a point source at the top center of the media. In the second frame the wave reaches the left interface. Note both the

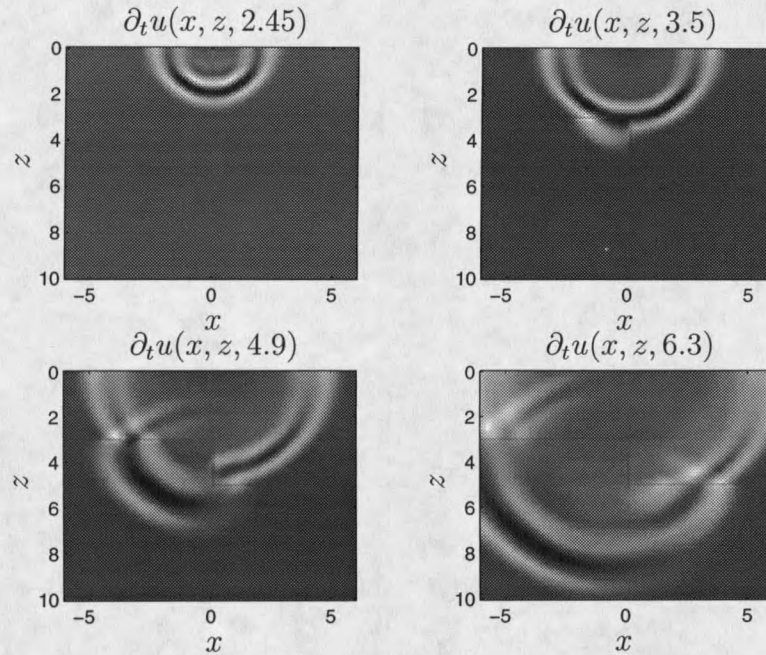


Figure 22: Adaptive stencil approximations to $\partial_t u$ at various times, t , for a two-dimensional discontinuous coefficient problem.

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 final frame. The effect of the absorbing boundary conditions can also be seen in this final frame.

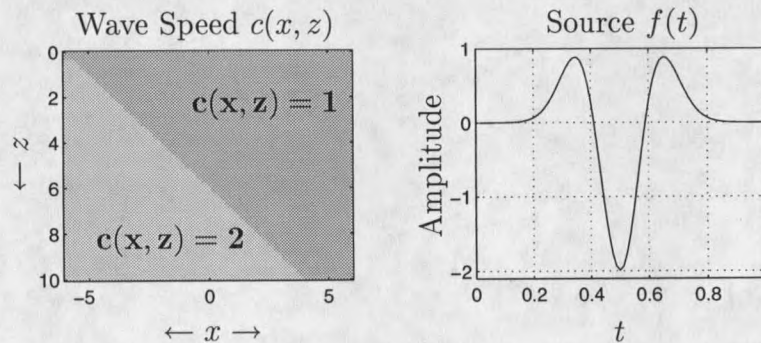


Figure 23: Wave speed and forcing function for a two-dimensional problem with a slanted interface.

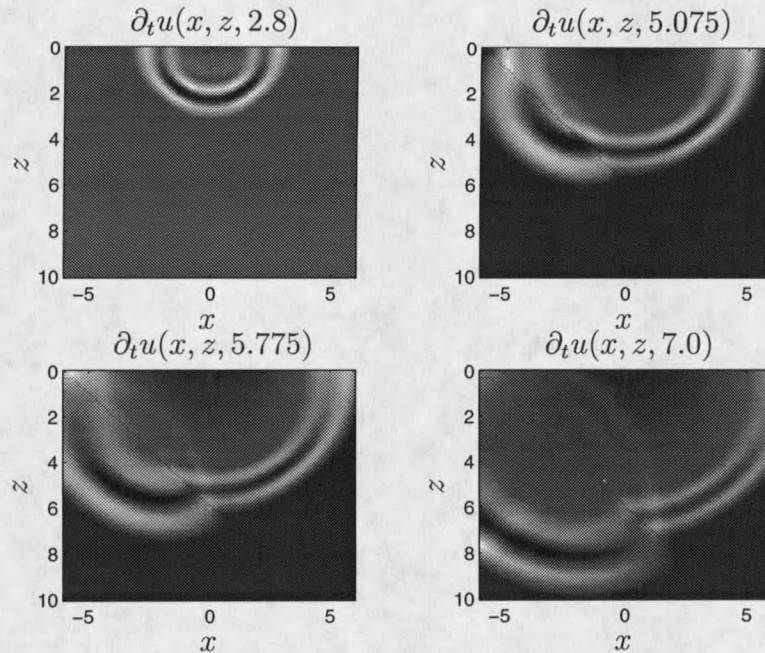


Figure 24: Adaptive stencil approximations to $\partial_t u$ at various times, t , for the slanted interface problem.

For a final example, consider (7.11) with a different wave speed as given in Figure 23. The first frame of Figure 24 shows a propagating wave emanating from a point source at the top center of the media. In the second frame the wave reaches the slanted interface. The next frame shows that part of the transmitted wave reaches the computational boundary and is absorbed, and the reflected wave also becomes apparent. Finally, both the transmitted and the reflected wave are being absorbed. The test case seems to indicate that no difficulties arise when discontinuities are not oriented along the computational grid.

The examples above imply that the method can propagate nonsmooth initial data and can propagate wave forms in materials that are modeled with discontinuous parameters. Although the examples listed above were taken from systems generated by certain second order equations, the method is applicable to a much broader class.

REFERENCES CITED

- [1] T. Belytschko and R. Mullen. On dispersive properties of finite element solutions. *Modern problems in elastic wave propagation*, 67(82), 1978.
- [2] Bleistein. *Mathematical Methods for Wave Phenomena*. Academic Press, Inc., 1984.
- [3] R. L. Burden and J. D. Faires. *Numerical Analysis*. Prindle, Weber & Schmidt, Boston, 3rd edition, 1985.
- [4] R. Clayton and B. Engquist. Absorbing boundary conditions for acoustic and elastic wave equations. *Bulletin of the Seismological Society of America*, 67(6):1529–1540, 1977.
- [5] R. Courant, K. O. Friedrichs, and H. Lewy. On the partial differential equations of mathematical physics. *IBM Journal*, 11:215–234, 1967.
- [6] R. Courant, E. Isaacson, and M. Rees. On the solution of nonlinear hyperbolic differential equations by finite differences. *Comm. Pure Appl. Math.*, 5:243–255, 1952.
- [7] M. A. Dablain. The application of high-order differencing to the scalar wave equation. *Geophysics*, 51:54–66, 1985.
- [8] B. Engquist, E. Fatemi, and S. Osher. Numerical solution of the high frequency asymptotic expansion for the scalar wave equation. CAM Report 93-05 UCLA Computational and Applied Mathematics, April 1993.
- [9] M. Glasner, D. Yevick, and B. Hermansson. Computer generated generalized propagation techniques. *Appl. Math. Lett.*, 4(5):85–90, 1991.
- [10] A. Hall and T. A. Porsching. *Numerical Analysis of Partial Differential Equations*. Prentice-Hall, New Jersey, 1st edition, 1990.
- [11] A. Harten. ENO schemes with subcell resolution. Report No. 87-56, ICASE, August 1987. Also published in *J. Comput. Phys.*
- [12] A. Harten and S. Osher. Uniformly high-order accurate nonoscillatory schemes I. *SIAM J. Numer. Anal.*, 24(2):279–309, 1987.

- [13] A. Harten, S. Osher, B. Engquist, and S. R. Chakravarthy. Some results on uniformly high-order accurate essentially nonoscillatory schemes. *Applied Numerical Mathematics*, 2:347–377, 1986.
- [14] F. John. *Partial Differential Equations*. Springer-Verlag, New York, 4th edition, 1982.
- [15] R. J. Krueger and Jr. R. L. Ochs. A Green's function approach to the determination of internal fields. *Wave Motion*, 11(6):525–543, 1989.
- [16] L. D. Landau and E. M. Lifshitz. *Theory of Elasticity*. Pergammon Press, Oxford, 1970.
- [17] R. J. LeVeque. *Numerical Methods for Conservation Laws*. Birkhauser Verlag, Basel, 1992.
- [18] S. Osher. Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations. Report No. 87-66, ICASE, September 1987.
- [19] S. Osher and C-W. Shu. High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations. *SIAM J. Numer. Anal.*, 28(4):907–922, August 1991.
- [20] G. Strang. On the construction and comparison of difference schemes. *SIAM J. Numer. Anal.*, 5(3):506–517, September 1968.
- [21] R. Vichnevetsky and J. B. Bowles. *Fourier analysis of Numerical approximations of Hyperbolic Equations*. SIAM, Philadelphia, 1st edition, 1982.
- [22] G. B. Whitham. *Linear and Nonlinear Waves*. John Wiley & Sons, 1973.
- [23] Erich Zauderer. *Partial Differential Equations of Applied Mathematics*. John Wiley & Sons, New York, 1983.
- [24] Chaoming Zhang. *Immersed interface method for wave equations with discontinuous coefficients*. PhD thesis, In progress at University of Washington, 1995.

MONTANA STATE UNIVERSITY LIBRARIES



3 1762 10245176 0

**HOUCHEN
BINDERY LTD
UTICA/OMAHA
NE.**