Old Dominion University

ODU Digital Commons

An Extension of Essentially Non-Oscillatory Shock-Capturing Schemes to Multi-Dimensional Systems of Conservation Laws

Jay Casper
Old Dominion University

Recommended Citation

AN EXTENSION OF ESSENTIALLY NON-OSCILLATORY SHOCK-CAPTURING SCHEMES TO MULTI-DIMENSIONAL SYSTEMS OF CONSERVATION LAWS

by

Jay Casper

B.A., Virginia Wesleyan College, 1977

M.S., Old Dominion University, 1987

A Dissertation Submitted to the Faculty of Old Dominion University
in Partial Fulfillment of the Requirements for the Degree of

DOCTOR OF PHILOSOPHY
in
COMPUTATIONAL AND APPLIED MATHEMATICS

December 1990

Approved by:

J.M. Dorrepaal (Director)

ABSTRACT

AN EXTENSION OF ESSENTIALLY NON-OSCILLATORY SHOCK-CAPTURING
SCHEMES TO MULTI-DIMENSIONAL SYSTEMS OF CONSERVATION LAWS

Jay Casper

Old Dominion University, 1990
Director: Dr. J. M. Dorrepaal

In recent years, a class of numerical schemes for solving hyperbolic partial differential equations has been developed which generalizes the first-order method of Godunov to arbitrary order of accuracy. High-order accuracy is obtained, wherever the solution is smooth, by an essentially non-oscillatory (ENO) piecewise polynomial reconstruction procedure, which yields high-order pointwise information from the cell averages of the solution at a given point in time. When applied to piecewise smooth initial data, this reconstruction enables a flux computation that provides a time update of the solution which is of high-order accuracy, wherever the function is smooth, and avoids a Gibbs phenomenon at discontinuities.

The promising results of Harten *et al.*, in the use of ENO schemes in solving the one-dimensional Euler equations of gas dynamics, have aroused considerable interest in the aerodynamic community. However, the application of these schemes to areas of scientific and industrial interest, such as aircraft configuration, obviously requires compressible flow solutions in more than one spatial dimension. It is this extension of ENO schemes to multi-dimensional application to which this study

is dedicated. In particular, a two-dimensional extension is proposed for the Euler equations of gas dynamics. Among the issues to be considered in this extension are achieving formal high-order two-dimensional spatial accuracy, the implementation of boundary conditions and application to general curvilinear coordinates.

Acknowledgements

The author is greatly appreciative to all who made possible the research that led to this dissertation.

As this research was in large part a computational effort, the support of the management and personnel at NASA Langley Research Center was indispensable to this project. In particular, I am indebted to Harold Atkins, my advisor at NASA LaRC. I would also like to thank Manuel Salas and V. Venkatakrishnan for their advice and encouragement. My three years at NASA LaRC have been both fruitful and enjoyable.

At Old Dominion University, I am grateful to a faculty which provided meaningful course work and guidance. As for Mark Dorrepaal, his help through some difficult moments will always be remembered.

Finally, nothing can replace the support that I have received at home from Cecilia Wozlek, mother of my son Jason, and my wife of thirteen years.

ii

Contents

List of Tables

Chapter 6

List of Figures

vii

Chapter 6

Chapter 1

Introduction

Solutions of equations which govern the flow of a compressible fluid have been the subject of intense scientific interest for many years. In particular, solutions of the Euler equations of gas dynamics have proven extremely useful in the transonic flow regime. Moreover, since the advent of the high–speed computer, the development of numerical schemes for discretizing the Euler equations has become a growing area of academic interest.

Because the quality of a given numerical solution is largely dependent upon the scheme one chooses for its computation, some *a priori* judgements must be made. These decisions might include the speed of the computation or the accuracy of solution. On a more qualitative level, as the hyperbolic nature of the Euler equations can give rise to discontinuous solutions, we might desire that our approximation avoid oscillatory behavior near steep gradients.

Clearly, on a more global level, it is desirable that an approximation to the solution of a hyperbolic system mimic the behavior of the true solution, as well as it can be mathematically understood. In Chapter 2, we review some of the more prominent aspects of the theory concerning the solutions of hyperbolic equations (largely due to Lax [30]), as well as the results pertaining to the existence of such solutions.

In Chapter 3, we review some of the theory concerning the stability and conver-

1

gence of discrete approximations to weak solutions of conservation laws. We then discuss some of the properties we desire in a numerical shock–capturing scheme, most notably high–order accuracy and monotonicity near discontinuities. Perhaps most important in this chapter is the discussion of what will become the design foundation of the class of *essentially non–oscillatory* (ENO) schemes. The basis of our design is due to Godunov [13], whose numerical scheme relied upon the characteristic structure of the solution of a hyperbolic system. Though not the first characteristic–based shock–capturing scheme (See [6]), Godunov was the first to propose the ingenious use of the Riemann problem as a means of detecting characteristic signal propagation.

A detailed discussion of ENO schemes in one spatial dimension is presented in Chapter 4. Unlike TVD schemes, in which we require that the solution's total variation be a non–increasing function in time, the definition of ENO schemes requires that we release such rigid control. Only in this manner can we escape the inherent limit of second–order accuracy. The willingness to accept such a weakened version of control of the total variation of the numerical solution is attributed to Harten and Osher [25] and represented a major conceptual change in the area of non–linear shock–capturing schemes.

In Chapter 5, we achieve the stated goal of this dissertation, in that the ideas of the previous chapter are extended to the solutions of hyperbolic equations in two spatial variables. As opposed to a one–dimensional setting, there are particular areas of concern in two dimensions. Perhaps of foremost concern is the high–order pointwise evaluation of the solution which is required for the numerical flux computation. Also, we must consider the implementation of boundary conditions and the extension to curvilinear co–ordinates.

Results of numerical experiments are presented in Chapter 6. Scalar equations are used in a grid refinement study in order to support the claim in Chapter 5

2

concerning the formal order of accuracy of our two–dimensional ENO scheme. Particular attention is paid to the two–dimensional Euler equations of gas dynamics. The test cases we discuss are demanding in that they involve non–trivial geometries and develop complicated shock structures. These calculations represent the first successful application of ENO schemes to boundary–value problems with solid walls.

Some concluding remarks are made in Chapter 7. In addition, some comments, although speculative at this point, are given concerning the areas of future use for high–order accurate shock–capturing methods.

Chapter 2

The Mathematical Theory of Shock Waves

In this chapter, for the sake of completeness, we introduce the reader to some of the basic theory involving the solutions of hyperbolic systems of partial differential equations. We define some basic terms and discuss the characteristic structure of these solutions. We then discuss a specific solution, that of the Riemann problem, which will be fundamental to the development of a large class of numerical shock-capturing schemes.

2.1 Hyperbolic Systems of Conservation Laws

A *conservation law*, in words, simply states that the rate of change of the total amount of substance within a domain D must be equal to the *flux* of that quantity through the boundary of D. For the purpose of mathematical formulation, let u denote the pointwise distribution of some property of that substance within D at a given point in time, and the flux by f. The conservation law, then, can be written

$$\frac{\partial}{\partial t} \int_D u \, dV = - \int_{\partial D} f \cdot \hat{n} \, dS \;, \qquad (2.1)$$

where dV is the volume element of D and dS is the surface element on ∂D, the boundary of D. We choose $\hat{n}$ to denote the outward normal to ∂D, so that the right-hand integral in (2.1) measures outflow, hence the minus sign.

4

Whereas (2.1) represents the conservation of u on the entire domain of interest, we desire an equation which specifies the conservation of u pointwise throughout D. To this end, we take the time derivative under the left-hand integral and then apply the Divergence Theorem on the right, obtaining

$$\int_D (u_t + div\, f)\, dV = 0 .$$
(2.2)

If we then divide by the volume V and take the limit of both sides of (2.2) as $V \to 0$, we obtain the differential conservation law

$$u_t + div\, f = 0 .$$
(2.3)

Now suppose that u in (2.3) represents more than one conserved quantity, *i.e.* is a vector of m components, each of which has a corresponding flux associated with it. Equation (2.3) then denotes a *system* of conservation laws. Imposing a coordinate system $\{ x_1, x_2, \ldots, x_k \}$ on the domain D, we then have k directional components to each of m fluxes. Carrying out the divergence of f within this framework, we write our system as

$$\frac{\partial}{\partial t} u^j + \sum_{i=1}^{k} \frac{\partial}{\partial x_i} f_i^j = 0 , \qquad j = 1, 2, \ldots, m , \quad (2.4)$$

where each f^j is a nonlinear function of $u^1, u^2, \ldots, u^m$. This dependence of f upon u enables us to finally write our non-linearly coupled system of conservation laws in the quasi-linear form

$$u_t + \sum_{i=1}^{k} A_i u_{x_i} = 0 ,$$
(2.5)

where A_i is the Jacobian matrix of f_i with respect to u. We call the system (2.5) *hyperbolic* (in the strict sense) if, for each x, t, u, and unit vector ω, the matrix

$$\sum_{i=1}^{k} \omega_i A_i$$

has real and distinct eigenvalues $\lambda^j(x, t, u, \omega)$, $j = 1, 2, \ldots, m$.

5

2.2 Weak Solutions and Entropy Conditions

We now concern ourselves with finding the solution of the system (2.4), subject to given initial and/or boundary conditions. At first glance, we would simply assume that any solution u of (2.4) would be continuously differentiable in each x_i, for all $t > 0$, and proceed accordingly. However, in general, such a C^1 solution does not exist. In fact, the solution u itself can become discontinuous within a finite time interval, even with smooth initial data. Due to this peculiar nature of hyperbolic equations, we develop the notion of a more generalized solution.

A solution u is called a *weak solution* of the system of conservation laws (2.4) if u satisfies the integral form of these laws, *i.e.*, if

$$\int_G u \, dV \Big|_{t_1}^{t_2} + \int_{t_1}^{t_2} \int_{\partial G} f \cdot \hat{n} \, dS \, dt = 0 \tag{2.6}$$

holds for every smoothly bounded domain G in D and for every time interval (t_1, t_2). Equation (2.6) is valid if u is merely bounded and measurable. If u happens to be C^1 in G, then it solves (2.4) in the classical sense.

Let $z(t)$ be a smooth surface moving with time t, and u be a weak solution of (2.4) which is continuously differentiable on either side of z, but which is discontinuous across z. Then on each point of z, it can be shown that this weak solution satisfies

$$s[u] = [f] \cdot \hat{n}_z \,, \tag{2.7}$$

where $[u]$ and $[f]$ denote the difference between values of u and f, respectively, on the two sides of z, $\hat{n}_z$ is the unit normal to z, and s is the speed with which z propagates with respect to $\hat{n}_z$. Relation (2.7) is a generalization of the *Rankine-Hugoniot jump condition*.

It is well-known that, in general, weak solutions are not uniquely determined from their initial data. This loss of uniqueness occurs in the class of piecewise smooth solutions, *i.e.* solutions which develop discontinuities. It turns out that cer-

6

tain types of discontinuities, across which (2.6) and (2.7) are satisfied, are nonetheless "undesirable." For instance, in the context of gas dynamics, the Rankine-Hugoniot conditions admit a "rarefaction shock" as a possible weak solution. Such a phenomenon (an obvious oxymoron!) is physically impossible, and therefore some condition of a qualitative or structural nature is needed to make such a solution inadmissible. An argument involving the way in which characteristics converge on a curve of discontinuity can be used to determine this condition. (See [30] for details.)

In order to formally express this condition, we first denote by R_1 and R_2 the regions of G which are separated by the surface z. Choose a point P on z, and let u_1 and u_2 denote the values of u at P, as limits from R_1 and R_2, respectively. Let $\{\lambda^j\}$ denote the eigenvalues of the matrix

$$\sum_{i=1}^{k} n_i A_i\,,$$

where $\{n_i\}$ are the components of $\hat{n}_x$, and are evaluated, as are the $\{A_i\}$, at the point P. Furthermore, we assume that these eigenvalues are labeled in increasing order, $\lambda^1 < \lambda^2, < \ldots, < \lambda^m$. Seeking to rule out "physically unrealizable" solutions, we require, for some k, $1 \le k \le m$, that the following inequalities hold:

$$\lambda^k(u_1) \; > \; s \; > \; \lambda^k(u_2)\,, \tag{2.8a}$$

$$\lambda^j(u_1) \; < \; s\,,\ \text{for } j < k\,, \qquad s \; < \; \lambda^j(u_2)\,,\ \text{for } j > k\,. \tag{2.8b}$$

Thus, for one and only one index k is the speed of the discontinuity intermediate to the characteristic speeds on either side. When considering the equations of gas dynamics, the relations (2.8) are equivalent to requiring that *entropy* increase across a shock. Hence we shall refer to these as *entropy conditions*, and to any solution which satisfies (2.6), (2.7), and (2.8) as an *entropy solution*. Relations (2.8) are also known as *Lax's shock conditions*.

We now move on to some of the theoretical results concerning weak solutions, their existence and whether or not they satisfy entropy conditions. In order to facil-

7

itate our presentation, until further notice, we restrict our discussion to hyperbolic systems in one spatial variable.

2.3   The Structure of Entropy Solutions

Consider the initial-value problem (IVP) for a one-dimensional hyperbolic system of differential conservation laws

$$u_t + f(u)_x = 0, \quad t > 0, \quad -\infty < x < \infty, \quad (2.9a)$$

$$u(x,0) = u^0(x), \quad (2.9b)$$

where $u(x,t)$ is a vector of m unknowns and each of the m components of f is a nonlinear function of $u^1, u^2, \ldots, u^m$. We assume that the IVP (2.9) is well-posed and that $u^0(x)$ is of bounded variation on $(-\infty, \infty)$. Equation (2.9a) can be written in quasi-linear form

$$u_t + A u_x = 0, \quad (2.10)$$

where A is the Jacobian matrix of f with respect to u. We assume that the system is hyperbolic, and that the eigenvalues of A are labeled in increasing order, i.e. $\lambda^1 < \lambda^2 < \ldots < \lambda^m$. Let r^k denote the k-th column vector of the right eigenvector matrix associated with A, and l^k the k-th row vector of the left eigenvector matrix.

Because the system (2.9) admits discontinuities, we therefore seek weak solutions. A weak solution of (2.9) must satisfy, for all rectangles $(a,b) \times (t_1,t_2)$, the relation obtained by integrating (2.9a) over the rectangle :

$$\int_a^b u(x,t_2)\,dx - \int_a^b u(x,t_1)\,dx + \int_{t_1}^{t_2} f(u(b,t))\,dt - \int_{t_1}^{t_2} f(u(a,t))\,dt = 0. \quad (2.11)$$

The solution $u(x,t)$ in (2.14) satisfies (2.9) in each smooth region, while across each discontinuity, it will satisfy the jump condition (2.7), which in this case, can be written

$$f(u_R) - f(u_L) = s\,(u_R - u_L), \quad (2.12)$$

8

where s is the speed of propagation of the discontinuity, and u_L and u_R are the states on the left and right, respectively. Furthermore, in order to rule out "undesirable" discontinuous solutions we require, *a priori*, that our solution satisfy the entropy conditions (2.8), *i.e.* for some index k, $1 \le k \le m$,

$$\lambda^k(u_L) > s > \lambda^k(u_R) \,, \tag{2.13a}$$

$$\lambda^j(u_L) < s \,, \text{ for } j < k \,, \qquad s < \lambda^j(u_R) \,, \text{ for } j > k \,. \tag{2.13b}$$

We define the *k-th characteristic field* by the family of curves that satisfies

$$\frac{dx}{dt} = \lambda^k \,. \tag{2.14}$$

We will call the field generated by (2.11) *genuinely nonlinear* if

$$\nabla \lambda^k \cdot r^k \ne 0 \,, \tag{2.15}$$

or *linearly degenerate* if

$$\nabla \lambda^k \cdot r^k \equiv 0 \,, \tag{2.16}$$

where the gradient is with respect to u. A *Riemann invariant* of the kth field is a quantity $\sigma(u)$ which satisfies

$$r^k \cdot \nabla \sigma = 0 \,, \tag{2.17}$$

where the gradient is with respect to u.

A physically acceptable discontinuity in a genuinely nonlinear field satisfies the entropy inequalities (2.13). Pictorially, characteristics converge on either side of the discontinuity $dx/dt = s$, as shown in Figure 1a. Henceforth, in analogy to gas dynamics, we shall refer to such a discontinuity as a *shock*.

Now, suppose that the kth characteristic field is genuinely nonlinear and that u is continuously differentiable in some region D of the $x - t$ plane. If, in addition, all Riemann invariants are constant in D, then we refer to u as a *centered simple wave* or *rarefaction wave*. All characteristics within this region (Figure 1b) are divergent

9
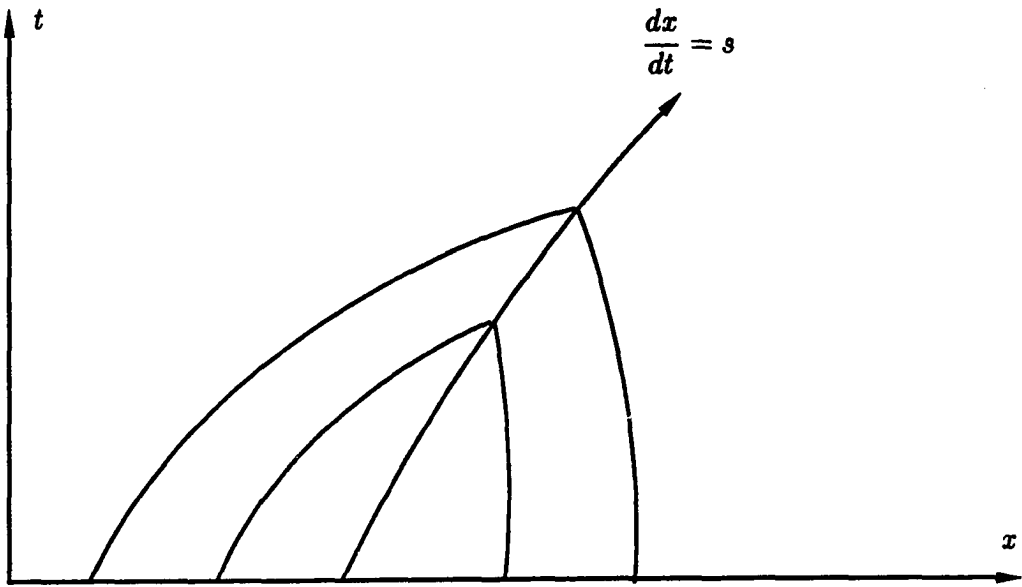
Figure 1a.   Characteristic structure of a shock
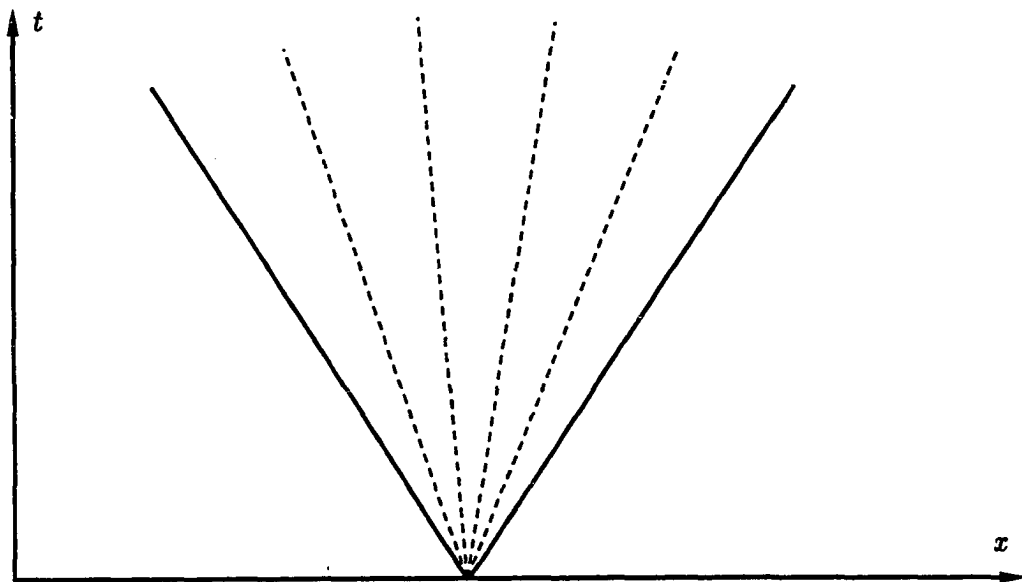
Figure 1b.   Characteristic structure of a rarefaction wave

10

straight lines, along which u is constant. Throughout D, u is a function depending only on $(x - x_0)/(t - t_0)$, where (x_0, t_0) is the center of the wave. These waves are the classical C^1 solutions of (2.9). We note that a rarefaction wave is continuous at its endpoints, but is not, in general, differentiable there.

If the kth field is linearly degenerate and contains a discontinuity propagating at speed s, then across this discontinuity, we have

$$\lambda^k(u_L) = s = \lambda^k(u_R) . \tag{2.18}$$

This situation is illustrated in Figure 1c. Again, in analogy to gas dynamics, we shall henceforth refer to this type of discontinuity as a *contact discontinuity*. In this case, the characteristics on either side of the discontinuity do not converge and the curve $dx/dt = s = \lambda^k$ is itself one of the characteristics.

2.4   The Riemann Problem

Because it is so fundamental to our nonlinear numerical schemes, we will now consider the solution of the *Riemann problem*. This problem is a special case of the IVP (2.9) which can be stated

$$u_t + f(u)_x = 0 , \tag{2.19a}$$

$$u(x,0) = \begin{cases} u_L , & x < 0 \\ u_R , & x > 0 , \end{cases} \tag{2.19b}$$

where u_L and u_R are constants.

It is proven in [53] that, provided the "initial jump" $|u_R - u_L|$ is sufficiently small, the Riemann problem (2.19) has a unique solution consisting of at most $m + 1$ constant states $\{u_L = u_0, u_1, \ldots, u_m = u_R\}$, separated by shocks, contact discontinuities, and rarefaction waves, centered at the origin. The characteristic structure of this solution is pictured in Figure 2. The manner in which a state u_L can be connected on the right, through constant states, to a state u_R makes the solution of (2.19) a

11

$$\frac{dx}{dt} = \lambda_k$$
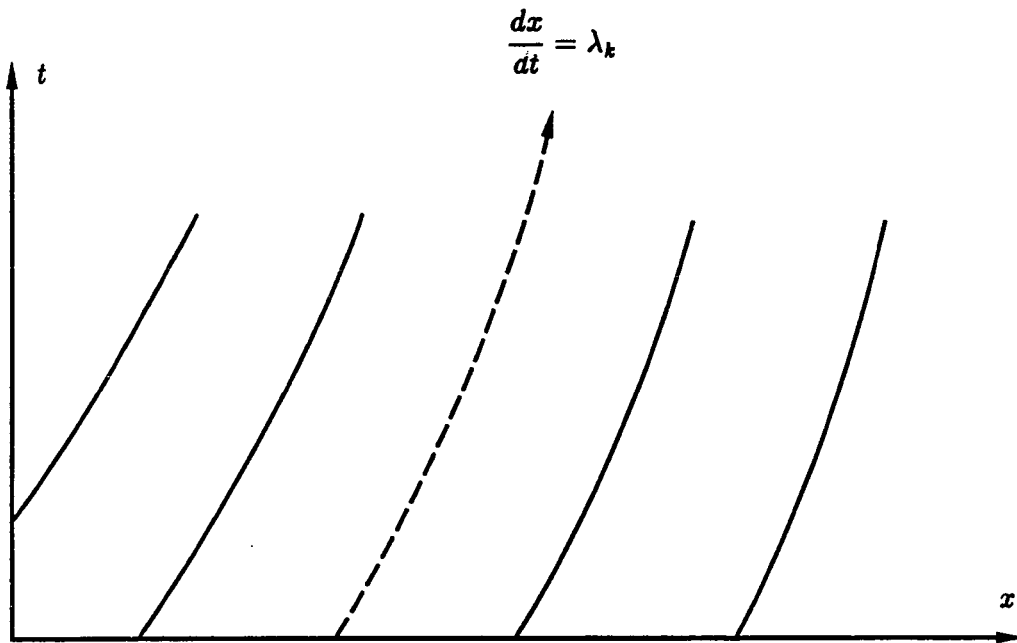
Figure 1c.   Characteristic structure of a contact discontinuity
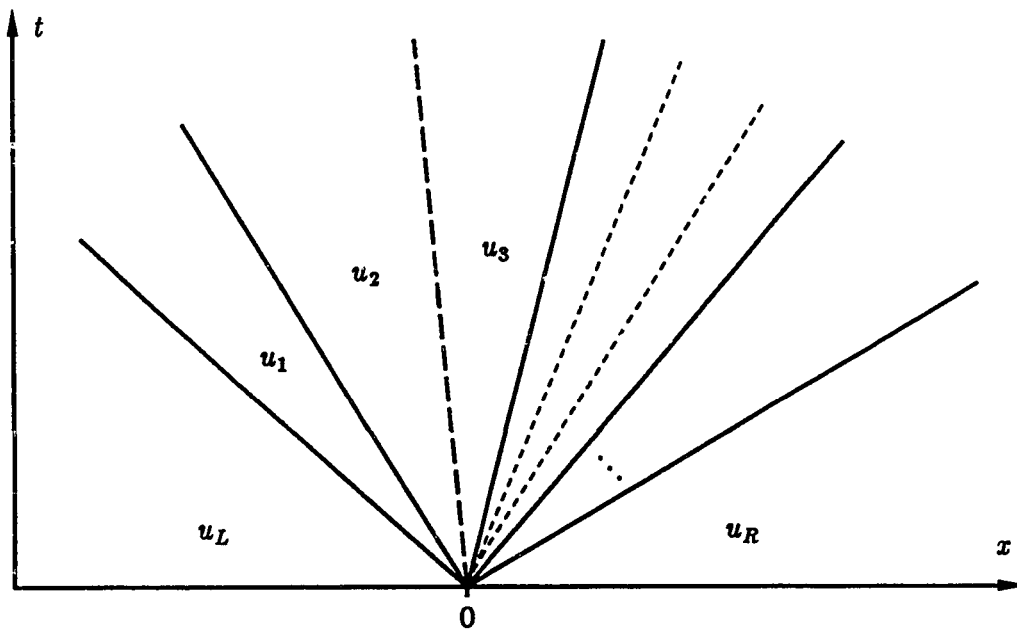
Figure 2.   Characteristic structure of a solution of the Riemann IVP (2.19)

12

function only of u_L, u_R, and the ratio x/t. Furthermore, in the specific case where equation (2.19a) represents the Euler equations of gas dynamics, it can be shown (See [53].) that $|u_R - u_L|$ can be arbitrary and the conclusion concerning the solution remains the same.

Chapter 3

Numerical Shock-Capturing Schemes

We now begin our discussion of numerical approximations of weak solutions of hyperbolic equations. In this chapter, we present theoretical results concerning the convergence of discrete approximations and establish criteria for stability. In addition, we seek to unify these ideas with our desire for high-order accurate, non-oscillatory schemes.

3.1 Discrete Approximations and Convergence

Now that we have developed the basic ideas concerning the existence of an entropy solution, we now wish to actually solve the IVP (2.9). Due to the nonlinearity of f, we cannot, in general, explicitly determine this solution and therefore seek to approximate it in a discrete fashion. To this end, we divide the spatial and temporal domains of (2.9) into finite intervals, where

$$x_{i-1/2} < x < x_{i+1/2} \, , \qquad -\infty < i < \infty \, ,$$

defines the ith spatial interval with center x_i and

$$t^n < t < t^{n+1} \, , \qquad 0 < n < \infty \, ,$$

represents the time discretization. Let Δx_i and Δt^n denote the (positive) lengths of the ith interval and n^{th} time-step, respectively.

14

In order that a discrete solution u be a weak solution of (2.9), we require that u satisfy equation (2.11) for all rectangles $(x_{i-1/2}, x_{i+1/2}) \times (t^n, t^{n+1})$, which we write in the form

$$\bar{u}_i^{n+1} = \bar{u}_i^n - \frac{\Delta t^n}{\Delta x_i} \left[\hat{f}_{i+1/2} - \hat{f}_{i-1/2} \right], \qquad (3.1a)$$

where

$$\bar{u}_i^n = \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t^n) \, dx, \qquad (3.1b)$$

and will be referred to as the *cell average* of u on the ith interval (or cell), and

$$\hat{f}_{i+1/2} = \frac{1}{\Delta t^n} \int_{t^n}^{t^{n+1}} f(u(x_{i+1/2}, t)) \, dt, \qquad (3.1c)$$

is the flux through the cell interface $x_{i+1/2}$. We note here that there is nothing "approximate" about the discrete solution $\{\bar{u}_i^n\}$ given by (3.1). The problem arises, in the general nonlinear case, in determining explicitly the flux (3.1c). It is ultimately here that we must rely on some type of numerical scheme.

Let v_i^n denote a numerical approximation to $u(x_i, t^n)$. The classical approach to the design of numerical methods for partial differential equations is to obtain a solvable set of equations for $\{v_i^n\}$ by replacing derivatives in the system (2.9a) with appropriate discrete approximations. Immediately, we have a conceptual difficulty using such an approach to compute solutions which may become discontinuous. We therefore follow the lead of Lax [29] who overcame this difficulty by considering numerical approximations to the *weak formulation* (3.1a), rather than to the system of PDE's (2.9a). Therefore, we consider schemes in the form

$$v_i^{n+1} = v_i^n - \frac{\Delta t^n}{\Delta x_i} \left[\bar{f}_{i+1/2} - \bar{f}_{i-1/2} \right], \qquad (3.2a)$$

where v_i^n approximates $u(x_i, t^n)$ and $\bar{f}$ is the *numerical flux*. It is natural to expect that the information that ultimately determines this approximate flux will come from the values of the $\{v_i^n\}$ which are nearby. In particular, we assume that $\bar{f}$ is a

15

Lipschitz continuous function of $2k$ arguments :

$$\bar{f}_{i+1/2} = \bar{f}(v^n_{i-k+1}, \ldots, v^n_{i+k}) .$$ (3.2b)

Furthermore, we require that the numerical flux be consistent with the "true flux" $f(u)$ in the sense that, when all $2k$ arguments are equal, $\bar{f}$ reduces to f :

$$\bar{f}(w, w, \ldots, w) = f(w) .$$ (3.2c)

It is through the recursive use of (3.2) as a numerical scheme, $n = 1, 2, \ldots,$ that the values $\{v^n_i\}$ can be successively updated from the previous time level.

We will say that a scheme of the form (3.2) is *consistent with the conservation law* (2.9a), whose solution it will approximate. A scheme in this form is also said to be in *conservative form.* The conservation character of (3.2) is expressed by the fact that, though the fluxes are approximated, the flux which is computed to enter the ith cell during the time interval (t^n, t^{n+1}) through the left endpoint is exactly equal to the approximate flux leaving the $(i - 1)$st cell through its right endpoint during the same time interval.

We have now established a set of prerequisites which enables us to determine a "desirable" numerical scheme for the approximation of weak solutions of conservation laws. The following result is due to Lax and Wendroff [31].

Theorem 3.1 *Suppose the difference scheme (3.2) is consistent with the conservation law (2.9a) with initial condition $\{v^0_i\} = u^0(x_i)$, $-\infty < i < \infty$. Further suppose that as Δx and Δt tend to zero, $\{v^n_i\}$ converges boundedly almost everywhere to some function $u(x,t)$. Then $u(x,t)$ is a weak solution of (2.9) with initial values $u^0(x)$.*

Furthermore, we know from Chapter 2 that the converged solution in Theorem 3.1 will satisfy the jump condition (2.12) across discontinuities. Hence, the term *shock-capturing*, a phrase coined by H. D. Lomax.

16

Though this theorem yields a desirable result, the assumption of convergence is quite stringent. In fact, the convergence of an approximate solution is itself an issue of greater concern, and which we now consider. For simplicity, we assume that the spatial and temporal discretizations are uniform; $\Delta x_i = h$, for all i, and $\Delta t^n = \Delta t$, for all n. We also assume that $\Delta t = O(h)$. Let $v_h(x, t)$ be a globally defined numerical approximation associated with the discrete values $\{v_i^n\}$. Furthermore, denote by E_h the numerical solution operator which advances $v_h(x, t)$ in time. We express the operation of E_h on the discrete solution $\{v_i^n\}$ by rewriting (3.2a) as

$$v_i^{n+1} = (E_h v^n)_i . \tag{3.3}$$

The classical approach (e.g. [44]) to the question of convergence leads us to the requirement of *linear stability*. Let us consider the scalar *constant coefficient case*

$$u_t + a u_x = 0 , \qquad a = \text{constant} . \tag{3.4}$$

Equation (3.4) is also known as the *linear advection equation*, and its solution, with initial data $u^0(x)$, is explicitly given by

$$u(x, t) = u^0(x - at) . \tag{3.5}$$

When applied to equation (3.4), many schemes (e.g. Cole–Murman [39], MacCormack [38]) take the form

$$v_i^{n+1} = \sum_{l=-k}^{k} C_l v_{i-l}^n , \tag{3.6}$$

where the $\{C_l\}$ are constants independent of i. Thus, in the constant-coefficient case the numerical solution operator E_h becomes a linear operator, and we shall refer to such schemes as "essentially linear" or just *linear schemes*.

We say that the numerical scheme (3.6) is *stable* if the operator E_h is uniformly bounded on $0 \le t^n \le T$, where $\Delta t = O(h)$. The classical linear analysis leads us to the conclusion that the scheme (3.6) is stable if and only if it satisfies von

17

Neumann's condition :

$$\left| \sum_{l=-k}^{k} C_l \, e^{-il\varepsilon} \right| \leq 1 \,, \qquad\qquad 0 \leq \varepsilon \leq \pi \,, \qquad (3.7)$$

where, in this equation, i is the imaginary unit.

Now consider $v_h(x,t)$ as a sequence of approximations whose limit we seek, as $h \to 0$. The notion of linear stability is related to convergence through *Lax's equivalence theorem*, which states that a consistent linear scheme is convergent if and only if it is stable.

Now suppose that a numerical scheme does not take the form (3.6) in the constant-coefficient case, or for any other reason the application of local linear stability analysis cannot be justified. The design of such *nonlinear* schemes that have desirable properties is the subject of the following sections, and these ideas will be carried throughout the remainder of this dissertation.

We close this section with an important theoretical result concerning the application of nonlinear schemes to hyperbolic systems. This result, due to Harten and Lax [23], gives a set of sufficient conditions under which we may expect convergence of a discrete solution to a weak solution of (2.9), the most crucial of which is the notion of *total variation stability*. Preparatory to this, we note that the *total variation* (TV) of the approximate solution $v_h(x,t)$ plays a critical role and, in the discrete case, can be defined

$$TV(v) \;=\; \sum_i \, \left| v_{i+1}^n - v_i^n \right| \,.$$

Furthermore, we say that a numerical scheme is *total-variation stable* if the total variation in x of $v_h(x,t)$ is uniformly bounded in $t, \Delta t$, and h. Recalling our assumption of bounded variation of the initial data u^0, we can express total-variation stability by

$$TV(v) \;\leq\; k \, TV(u^0) \,, \qquad\qquad 0 < t < T \,, \qquad (3.8)$$

where k is a constant, independent of x. We may now state the following.

18

Theorem 3.2 *Suppose that a numerical scheme is in conservation form (3.2). If the resulting approximation is total-variation stable, then the scheme is convergent, and its limit is a weak solution of (2.9).*

If, in addition to the theorem's hypothesis, it is assumed that consistency with an entropy condition implies uniqueness of the IVP (2.9), then the limit solution will be unique. However, uniqueness in the general case has yet to be proven.

3.2 Accuracy and Monotonicity

Let E_h denote the numerical solution operator (3.3) and let $E(t)$ denote the evolution operator of the exact solution u of the IVP (2.9). We say that the numerical scheme (3.2) is r^{th}-order accurate (in a pointwise sense) if its local truncation error satisfies

$$E(\Delta t)\, u \; - \; E_h\, u \; = \; O(h^{r+1}) \tag{3.9}$$

for sufficiently smooth h, where $\Delta t = O(h)$.

It might seem reasonable that any higher-order accurate scheme would be more desirable than any lower-order accurate one. However, a formally high order of accuracy is not the only desirable property of a numerical scheme. It is well-known (See [19,34,35].) that some schemes which yield excellent results for smooth solutions exhibit spurious $O(1)$ oscillations in the presence of steep gradients.

Therefore, in addition to high-order accuracy in regions where the solution is smooth, we also desire that the scheme be capable of propagating discontinuities without such oscillations. In the scalar case, this can be accomplished by designing schemes to be *monotonicity-preserving*, *i.e.* whenever v is a monotone mesh function, so is $E_h\, v$. Godunov [13] considered this approach in the constant-coefficient case (3.4), and was able to prove that monotonicity-preserving linear schemes are necessarily only first-order accurate. Therefore, if we are to achieve our objectives

19

of high-order accuracy and monotonicity in our design, we must do so with *non-linear* schemes. Throughout the remainder of this dissertation, we concentrate on the design of such schemes, which even in the constant-coefficient case (3.4) are nonlinear.

In [16], Harten was able to achieve second-order accuracy by rigorously developing a class of nonlinear schemes which are required to be *total-variation diminishing* (TVD). A TVD numerical solution operator E_h is defined by

$$TV(E_h v) \leq TV(v), \qquad 0 < t < T. \qquad (3.10)$$

As has become customary in the literature, the word "diminishing" is used loosely as a synonym for "non-increasing." We note that TVD schemes are automatically total-variation stable, since (3.10) satisfies (3.8) with $k = 1$. Also it is shown in [16] that TVD schemes are necessarily monotonicity-preserving. Finally, it can be shown that TVD schemes are at most second-order accurate, in the L_1 norm. In fact, regardless of its form, any TVD scheme must be only first-order accurate, in a pointwise sense, near local extrema, and therefore is globally first-order accurate in the L_∞ norm. This is due to the fact that, in order to satisfy (3.10) at every time step, the scheme's design must ensure that the value of an isolated local maximum may only decrease in time, while that of a local minimum may only increase. It is this perpetual damping of extrema that ultimately determines the pointwise truncation error, and is necessary in order that the total variation of the solution be so strongly controlled.

Harten and Osher [25] later introduced a larger class of schemes, in which the numerical solution operator is required only to diminish the number of local extrema, rather then their relative size. These schemes are referred to as (strictly) *non-oscillatory*, and are defined by

$$N_0(E_h v) \leq N_0(v), \qquad 0 < t < T. \qquad (3.11)$$

20

where $N_0(v)$ denotes the number of local extrema in v.

Unlike TVD schemes, which are a subset of this class, non-oscillatory schemes are not required to damp the value of each local extrema at every time step, but are allowed to occasionally accentuate them. Harten and Osher were able to design these schemes to be uniformly second-order accurate in smooth regions (including at extrema) as well as monotonicity preserving. However, the accuracy of a non-oscillatory scheme is limited to second order. Nonetheless, the willingness to accept a weaker notion of control over the possible growth in the total variation of the numerical solution introduced major conceptual changes. Indeed it was the new ideas in [25] that would lead to the development of the high-order accurate shock-capturing schemes which we further develop and extend in this dissertation.

3.3 Godunov–Type Schemes

In this section we begin to design a class of nonlinear schemes, which we desire to be high-order accurate and capable of propagating discontinuities without spurious oscillations. As there is more than one approach to the actual implementation of such schemes, here we present an abstract design, leaving details to later sections.

First and foremost, we require our schemes to be in conservative form (3.2). Secondly, even though our initial interpretation of the discrete approximation $\{v_i^n\}$ was in a pointwise sense, through association with a globally defined numerical solution $v_h(x,t)$, the scheme (3.2) is easily extended to the notion of cell averages (3.1b) by simply writing our scheme

$$\bar{v}_i^{n+1} = \bar{v}_i^n - \frac{\Delta t^n}{\Delta x_i}\left[\bar{f}_{i+1/2} - \bar{f}_{i-1/2}\right], \qquad (3.12)$$

where

$$\bar{v}_i^n = \frac{1}{\Delta x_i}\int_{x_{i-1/2}}^{x_{i+1/2}} v_h(x,t^n)\,dx, \qquad (3.13)$$

21

is an approximation to the exact cell average $\bar{u}_i^n$ of the weak formulation (3.1). Thus, we choose to update approximate cell averages $\{v_i^n\}$ of the exact solution $\{\bar{u}_i^n\}$. This "finite–volume" approach was introduced by Godunov. Because the first–order accurate scheme he presented in [13] contains the basic elements upon which we build our class of high–order schemes, we find it prudent to discuss Godunov's scheme in some detail, before proceeding with our general design.

Godunov begins the derivation of his scheme by considering the numerical approximation $v_h(x, t^n)$ at the discrete time level t^n to be a piecewise constant function in x (Figure 3a), where each constant is a cell average, *i.e.*

$$v_h(x, t^n) \;=\; v_i^n \,, \qquad\qquad x_{i-1/2} < x < x_{i+1/2} \,, \qquad (3.14\text{a})$$

for all i. There are two steps in calculating the approximate solution at the next time level $t^{n+1} = t^n + \Delta t$. First, we solve exactly the initial-value problem

$$u_t + f(u)_x = 0 \,, \qquad\qquad u(x, t^n) = v_h(x, t^n) \qquad\qquad (3.14\text{b})$$

for the time interval $t^n \leq t \leq t^{n+1}$. Denote this solution globally as $u^n(x, t)$.

The piecewise-constant form of the initial conditions (3.14a) creates discontinuities at each cell interface $x_{i+1/2}$, with the value v_i^n on the left and v_{i+1}^n on the right. Thus the IVP (3.14b) is a sequence of Riemann problems (Figure 3b). Recall from Section 2.4 that the solution of the Riemann problem (2.19) consists of at most $m + 1$ constant states separated by the various types of waves discussed in that section, and can be expressed as a function of the left and right states u_L and u_R, and the ratio x/t. Denote the general solution of (2.19) by $u(x/t; u_L, u_R)$.

Now if we restrict the size of the time step Δt by

$$\frac{\Delta t}{\Delta x} |\lambda|_{\max} \leq \frac{1}{2} \,,$$

where $|\lambda|_{\max}$ is the magnitude of the globally largest eigenvalue, then there is no interaction amongst neighboring Riemann problems. In this way, the global solution

22
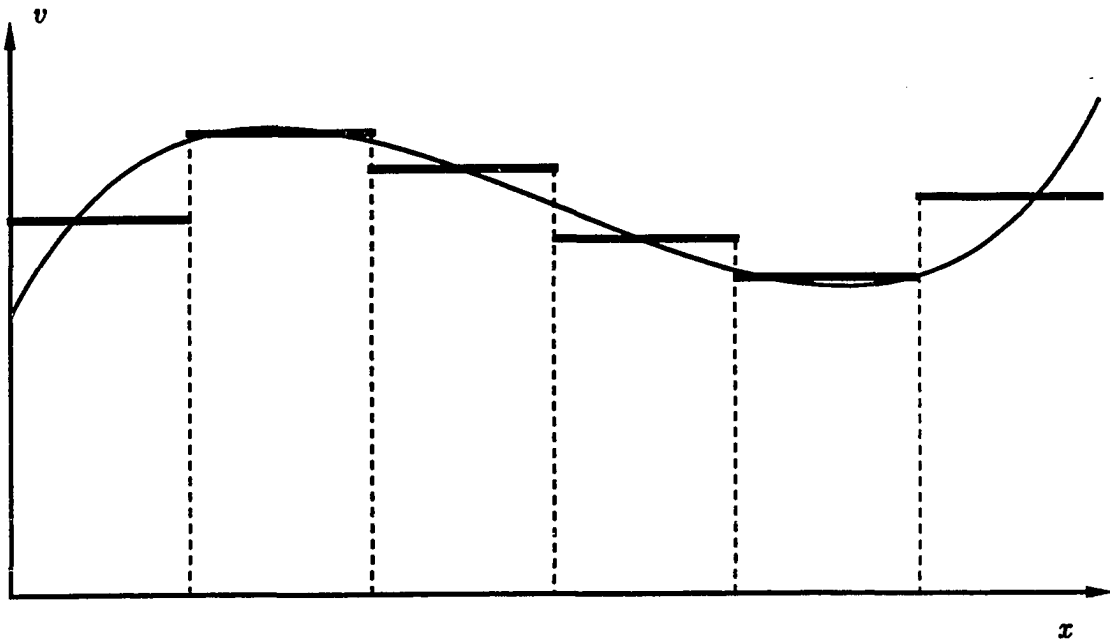
Figure 3a.   The horizontal bars represent the initial data of equation (3.14a)
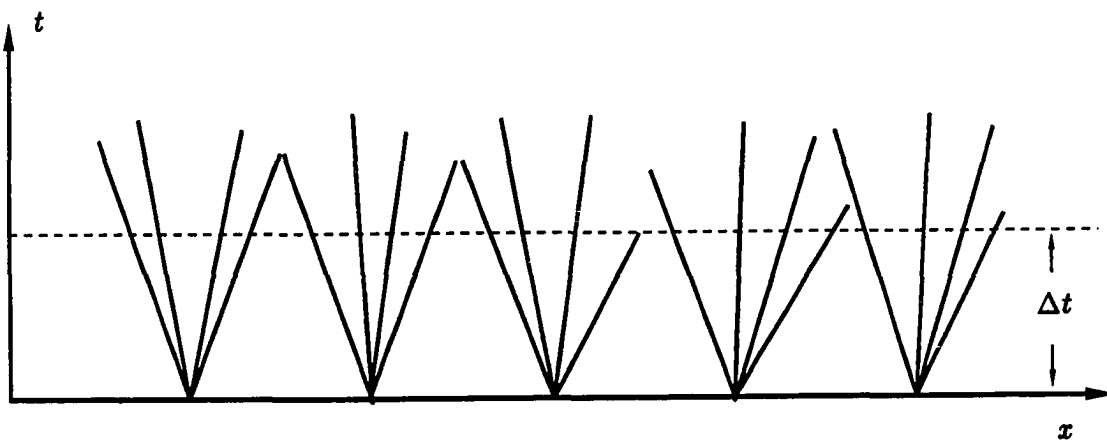
Figure 3b.   "Solutions in the small" of IVP (3.14b)

23

$u^n(x,t)$ of (3.14b) can be expressed exactly in terms of the solutions of the local Riemann problems :

$$u^n(x,t) = u(\,(x - x_{i+1/2})/(t - t^n)\,;\, \bar{v}_i^n,\, \bar{v}_{i+1}^n)\,, \qquad (3.14c)$$

for $x_i < x < x_{i+1}$, for all i, and $t^n \le t \le t^{n+1}$.

The second step in Godunov's scheme actually updates the $\{\bar{v}_i^n\}$. We obtain a piecewise-constant approximation $v_h(x, t^{n+1})$ by averaging the solution (3.14c) at time t^{n+1}, $i.e.$

$$\bar{v}_i^{n+1} = \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} u^n(x, t^{n+1})\, dx \,. \qquad (3.14d)$$

Thus, on the interval $x_{i-1/2} < x < x_{i+1/2}$, the update $\bar{v}_i^{n+1}$ is an average of the contribution, at time t^{n+1}, due to the right-running waves emanating from $x_{i-1/2}$, and that of the left-running waves emanating from $x_{i+1/2}$. Using this interpretation, we rewrite (3.14d) as

$$\bar{v}_i^{n+1} = \frac{1}{\Delta x_i} \int_0^{\Delta x_i/2} u(\,x/\Delta t\,;\, \bar{v}_{i-1}^n,\, \bar{v}_i^n)dx + \frac{1}{\Delta x_i} \int_{-\Delta x_i/2}^0 u(\,x/\Delta t\,;\, \bar{v}_i^n,\, \bar{v}_{i+1}^n)dx \,. \quad (3.15)$$

We can show that Godunov's scheme is conservative by noting that $u^n(x,t)$ is an exact solution of the system of conservation laws in (3.14b) and therefore must satisfy the weak formulation (2.11) on the rectangle $(x_{i-1/2}, x_{i+1/2}) \times (t^n, t^{n+1})$. Using this formulation and (3.14d), we can write Godunov's scheme in the form

$$\bar{v}_i^{n+1} = \bar{v}_i^n - \frac{\Delta t^n}{\Delta x_i} \left[\bar{f}_{i+1/2} - \bar{f}_{i-1/2} \right] \,, \qquad (3.16a)$$

which is the conservative form (3.12) and the Godunov flux function is

$$\bar{f}_{i+1/2} = f(\,u(0\,;\, \bar{v}_i^n,\, \bar{v}_{i+1}^n)\,)\,. \qquad (3.16b)$$

Furthermore, in this form, because we require the solution of the local Riemann problem only at $x = x_{i+1/2}$, we may weaken our time-step restriction to

$$\frac{\Delta t}{\Delta x}|\lambda|_{max} \le 1 \,.$$

24

As previously noted, Godunov's scheme is first-order accurate. We now wish to generalize the procedure (3.14) to an arbitrary order of accuracy r. However, if we never deal with the numerical solution in any form but that of its cell averages, we see that there is a limitation on the order of accuracy, by the following argument. For some $h > 0$, define a global "sliding-average" value function $\bar{w}$ in x by

$$\bar{w}(x) = \frac{1}{h} \int_{-h/2}^{h/2} w(x + \xi) d\xi.$$ (3.17)

If we apply the mid-point rule to the integral in (3.17), we get, for smooth $\bar{w}(x)$,

$$\bar{w}(x) = w(x) + O(h^2),$$

which shows that, as a pointwise approximation, a cell average is never any better than first-order accurate (or second-order accurate in the L_1 sense). Therefore, if we desire a higher-order accurate scheme, the cell averages themselves will not do. We must be able to somehow extract high-order pointwise information from these averages, in a manner that does not produce oscillations near discontinuities.

To this end, define an operator $R = R(x; \bar{w})$, which reconstructs a discrete set $\{\bar{w}_i\}$, and yields a piecewise smooth function that approximates $w(x)$ to some order of accuracy. We define a Godunov-type scheme by essentially the same procedure as (3.14), except that the piecewise-constant approximation (3.14a) is replaced by the more general approximation determined by the operator R. Its abstract form can be expressed as follows:

(i) Reconstruction: Define

$$v_h(x, t^n) = R(x; \bar{v}^n).$$ (3.18a)

(ii) Solution in the Small: For $t^n \leq t \leq t^{n+1}$, define

$$v_h(\cdot, t) = E(t - t^n) v_h(\cdot, t^n),$$ (3.18b)

25

where $E(t)$ is the exact evolution operator of the IVP (2.9).

(iii) <u>Cell</u>–<u>Averaging</u>: Close the time loop of the algorithm by defining

$$\bar{v}_i^{n+1} = \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} v_h(x, t^{n+1}) \, dx \,. \tag{3.18c}$$

Clearly, the reconstruction operator R is the most important feature in the design of these schemes. In fact, it is not immediately clear how to approach the "solution-in-the-small" step (3.18b), when left and right states at each interface are not constant values, as in the classical Riemann problem, but are higher-order pointwise approximations to $v_h(x, t)$. This issue, as well as the details of this reconstruction operator, the conservation of the schemes, and an analysis of their truncation error will be discussed in Chapter 4.

3.4 Approximate Riemann Solvers

Before moving on to the rigorous development of high-order schemes of Godunov type, we make a general observation of these schemes at the abstract level presented in the last section. In step (ii) of these schemes (3.18), all of the local Riemann problems are solved exactly. This process, depending on a given system of equations, could be long and difficult. However, all of these local solutions are then averaged in the update step (iii). Consequently, many of the fine details of the exact solution in (3.18b) are later ignored in evaluating $\bar{v}_i^{n+1}$ by averaging the exact solution over the interval $(x_{i-1/2}, x_{i+1/2})$. We therefore consider replacing the exact solution $u(x/t; u_L, u_R)$ in (3.15) with an approximation.

In conservation form (3.16a), the numerical flux we desire contains only that part of the solution of the local Riemann problem which exists at $x = x_{i+1/2}$ in the time interval $t^n \leq t \leq t^{n+1}$. So we first desire an expression that will express the flux $\bar{f}_{i+1/2}$ in terms of the known initial conditions u_L, u_R of the Riemann problem. Once again, for ease of presentation, we assume a uniform computational mesh, and note

26

that the Riemann solution $u(x/t; u_L, u_R)$ must satisfy the weak formulation (2.11) on the rectangle $(-h/2, 0) \times (t^n, t^{n+1})$, i.e.

$$\int_{-h/2}^{0} u(x/t^{n+1}; u_L, u_R)dx - \frac{h}{2} u_L + \Delta t \, \bar{f}_{i+1/2} - \Delta t \, f(u_L) = 0 \, ,$$

which gives the relation

$$\bar{f}_{i+1/2} = f(u_L) - \frac{1}{\Delta t} \int_{-h/2}^{0} u(x/t^{n+1}; u_L, u_R)dx + \frac{h}{2\Delta t} u_L \, . \qquad (3.19a)$$

Performing the same operation over the rectangle $(0, h/2) \times (t^n, t^{n+1})$ yields

$$\bar{f}_{i+1/2} = f(u_R) + \frac{1}{\Delta t} \int_{0}^{h/2} u(x/t^{n+1}; u_L, u_R)dx - \frac{h}{2\Delta t} u_R \, . \qquad (3.19b)$$

Thus, the only nontrivial operation in the flux computation is the integration of the Riemann solution through either the left–running waves or through the right–running waves. We therefore seek an approximate solution which will simplify the integrals in (3.19).

To this end, consider the IVP (2.9) in matrix form

$$u_t + A u_x = 0 \, , \qquad\qquad u(x,0) = u^0(x) \, , \qquad (3.20)$$

where A is the Jacobian of f with respect to u. Recall from Section 2.3 that the matrix A has real and distinct eigenvalues $\{\lambda^k\}$, which therefore give rise to a complete set of right eigenvectors $\{r^k\}$. Thus the matrix R of right eigenvectors is invertible and the rows of the inverse L form a complete set of left eigenvectors $\{l^k\}$. We may also assume that, upon suitable normalization, the orthonormal dot-product relation

$$l^i \cdot r^j = \delta_{ij} \qquad (3.21a)$$

is satisfied, as well as the matrix product relation

$$LAR = \Lambda \, , \qquad (3.21b)$$

where the elements of the diagonal matrix Λ are given by

$$\Lambda_{ij} = \lambda^i \delta_{ij} \, . \qquad (3.21c)$$

27

Now consider the IVP (3.20) in the constant-coefficient case, *i.e.* let $f = Au$, so that A is a constant matrix, as are R, L, and Λ. Let w be the column vector of characteristic variables, defined by

$$w = Lu\ , \qquad \text{or} \qquad w^k = l^k \cdot u\ , \qquad k = 1, 2, \dots, m\ . \tag{3.22a}$$

By multiplying (3.20) on the left by L and using the relations (3.21) and the definition (3.22a), we see that the system in (3.20) decouples into m scalar characteristic equations, and the IVP becomes

$$w_t^k + \lambda^k w_x^k = 0\ , \qquad\qquad w^k(x, 0) = l^k \cdot u^0(x)\ . \tag{3.22b}$$

Each of the m differential equations in (3.22b) is identical to the linear advection equation (3.4), whose solution (3.5) is easily obtained. The desired solution u can then be recovered by noting the relation

$$\sum_{k=1}^{m} w^k r^k = u\ . \tag{3.22c}$$

We now consider the Riemann problem (2.19) in the case of a constant co-efficient matrix A. Here the solution (Figure 4) is composed of constant states $\{u_L = u_0, u_1, u_2, \dots, u_m = u_R\}$, separated by discontinuities only, whose characteristic paths are straight lines determined by

$$\frac{dx}{dt} = \lambda^k\ , \qquad k = 1, 2, \dots, m\ .$$

Using (3.22c), the difference in the initial conditions, $u_R - u_L$, can be expressed in terms of the differential characteristic variables $\alpha^k = l^k \cdot (u_R - u_L)$,

$$u_R - u_L = \sum_{k=1}^{m} \alpha^k r^k\ . \tag{3.23a}$$

An intermediate state u_j then can be calculated by

$$u_j = u_L + \sum_{k=1}^{j} \alpha^k r^k\ , \tag{3.23b}$$

28
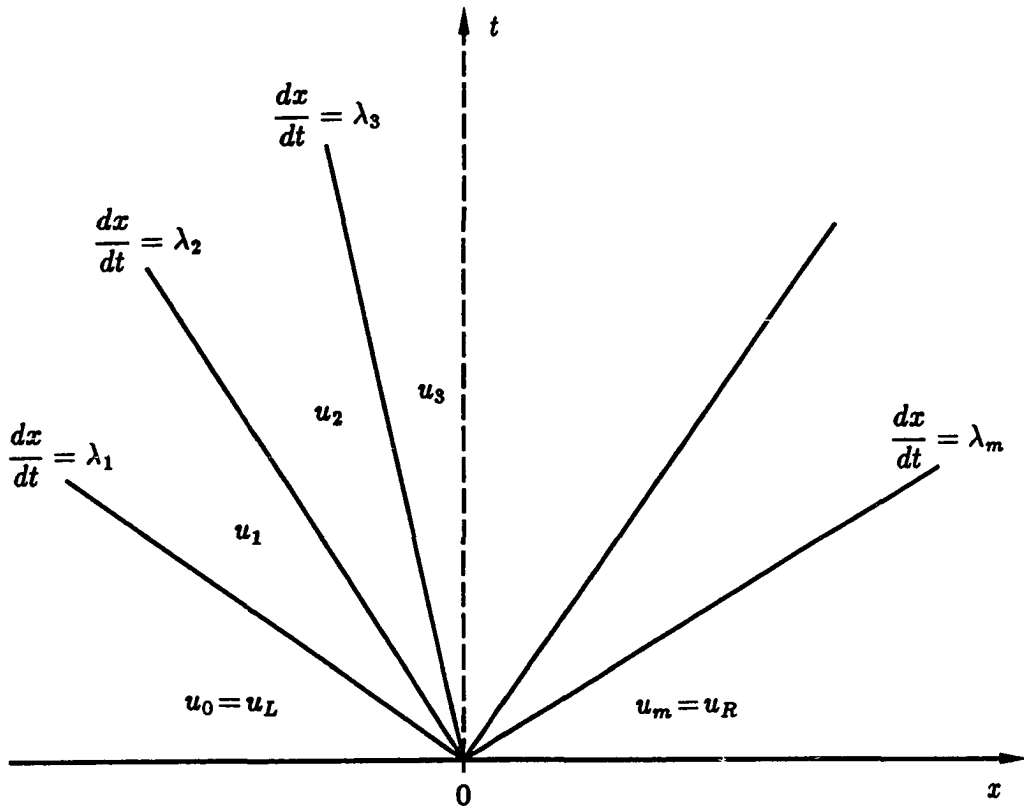
Figure 4.   Solution of the constant-coefficient Riemann problem

29

or

$$u_j \;=\; u_R - \sum_{k=j}^{m} \alpha^k r^k \,, \tag{3.23c}$$

for $j = 1, 2, \ldots, m-1$.

Let u_j $(0 \le j \le m)$ be the value of the Riemann solution at $x = 0$, and for $t^n \le t \le t^{n+1}$. Because of the simple structure of the solution in this case, we see that the integration at $t = t^{n+1}$ in (3.19) greatly simplifies. Upon substitution of the solution (3.23c) for the integrand in (3.19a), we can write the integral

$$\frac{1}{\Delta t} \int_{-h/2}^{0} u(x/t^{n+1} \,; u_L, u_R) dx \;=\; \frac{h}{2\Delta t} u_L \;+\; \sum_{k=1}^{j} \frac{h_k}{\Delta t} \alpha^k r^k \,,$$

where $\{h_k\}_1^j$ are points in the interval $(-h/2, 0)$, as shown in Figure 5. Also, because the characteristics are straight lines in this case, we have

$$\frac{h_k}{\Delta t} \;=\; \lambda^k \,, \qquad k = 1, 2, \ldots, j \,.$$

Thus, in the constant coefficient case, we can express the integral in (3.19a) as a discrete sum of the contributions of the Riemann solution, summed through the waves with negative speeds. Using this interpretation, we can write the flux (3.19a) in the form

$$\bar{f}_{i+1/2} \;=\; f(u_L) + \sum_{k=1}^{m} \alpha^k \lambda^{k(-)} r^k \,, \tag{3.24a}$$

where

$$\lambda^{k(-)} \;=\; \min\left(\lambda^k, 0\right) \;=\; 1/2\left(\lambda^k - |\lambda^k|\right) \,.$$

A symmetrical argument allows us to substitute (3.23c) into (3.19b), and we obtain

$$\bar{f}_{i+1/2} \;=\; f(u_R) - \sum_{k=1}^{m} \alpha^k \lambda^{k(+)} r^k \,, \tag{3.24b}$$

where

$$\lambda^{k(+)} \;=\; \max\left(\lambda^k, 0\right) \;=\; 1/2\left(\lambda^k + |\lambda^k|\right) \,.$$

Summing (3.24a) and (3.24b) yields the particularly convenient result

30
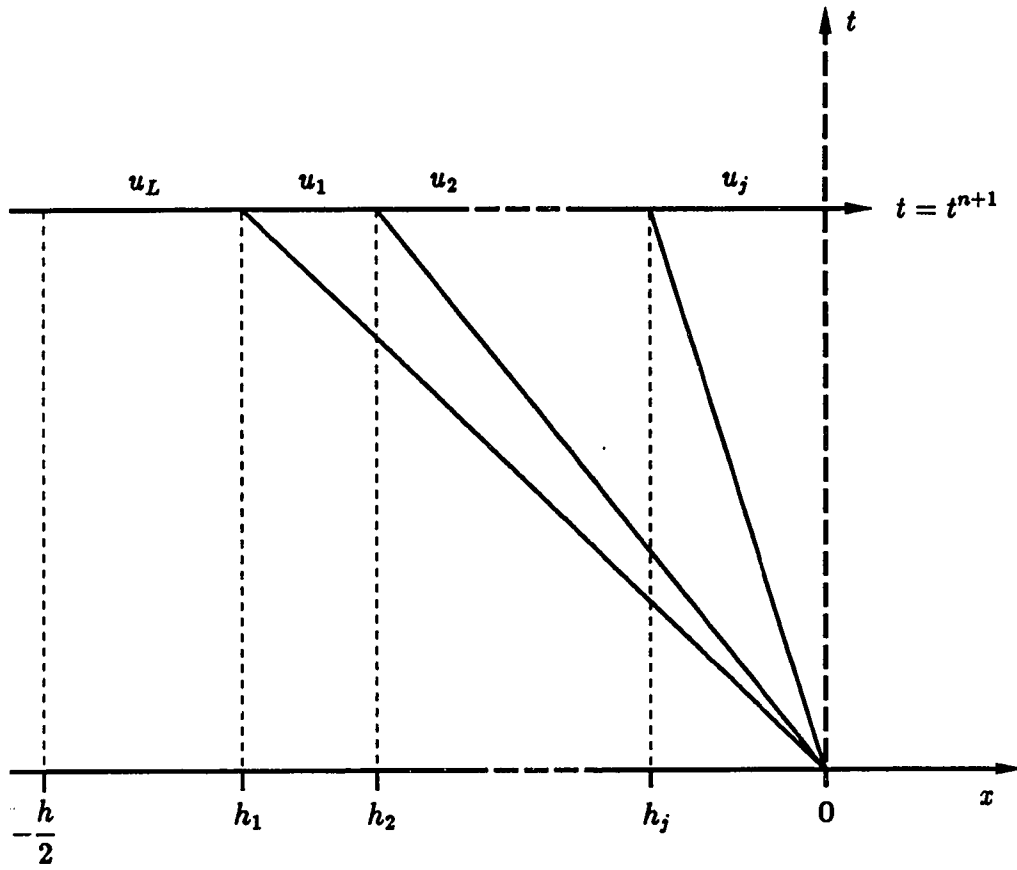
Figure 5. Integration of the solution of the constant–coefficient Riemann problem at $t = t^{n+1}$ through its left–running waves.

31

$$\tilde{f}_{i+1/2} = \frac{1}{2} \left[f(u_L) + f(u_R) \right] - \frac{1}{2} \sum_{k=1}^{m} \alpha^k |\lambda^k| r^k . \tag{3.24c}$$

It has been suggested by Roe [47] that we account for the nonlinearity in the solution of a given hyperbolic system by "locally linearizing" the system at each point where the computation of a flux is desired. Thus, at the "solution-in-the-small" step (3.18b), we solve at each interface $x_{i+1/2}$, the IVP

$$u_t + \hat{A}_{i+1/2} u_x = 0 , \qquad u(x,0) = \begin{cases} u_L , & x < x_{i+1/2} \\ u_R , & x > x_{i+1/2} , \end{cases} \tag{3.25}$$

where $\hat{A}_{i+1/2}$ is the original matrix A whose value has been fixed relative to the left and right states at $x_{i+1/2}$.

There are many ways to define $\hat{A}$, the most convenient of which would be some type of average of left and right states. Roe has proposed such a matrix $\hat{A}(u_1, u_2)$ with the following properties:

(i) $\hat{A}(u_1, u_2)$ has real eigenvalues and a complete set of eigenvectors.

(ii) For any two states u_1, u_2,

$$f(u_2) - f(u_1) = \hat{A}(u_1, u_2) (u_2 - u_1) . \tag{3.26a}$$

(iii) $\hat{A}(u_1, u_2)$ is consistent in the sense

$$\hat{A}(u, u) = A(u) . \tag{3.26b}$$

Once a particular $\hat{A}_{i+1/2}$ is determined by the values u_L, u_R at an interface $x_{i+1/2}$, the corresponding quantities of the matrices $\hat{R}_{i+1/2}$, $\hat{L}_{i+1/2}$, and $\hat{\Lambda}_{i+1/2}$ are then used in the numerical flux formula (3.24c). We will use Roe's ideas, "Roe's scheme" as it is called, in the application of our new schemes, and discuss the details when appropriate.

Roe's Riemann solver contains a great amount of detail including $m-1$ intermediate states. Furthermore, even when a characteristic separating two constant

32

states lies upon the t-axis, the numerical flux (3.24c) is still valid in the sense of a weak solution. Such a characteristic satisfies $dx/dt = \lambda^k = 0$, in which case, the discontinuity between u_{k-1} and u_k is stationary, and the jump condition (2.12) is still satisfied, i.e. $f(u_{k-1}) = f(u_k)$. Thus, Roe's scheme is capable of perfectly resolving a stationary discontinuity. If, in addition, the relation $\lambda^k(u_{k-1}) > 0 > \lambda^k(u_k)$ is satisfied, then such a discontinuity is a stationary shock. However, in the case where $\lambda^k(u_{k-1}) < 0 < \lambda^k(u_k)$, equation (3.24c) still results in $f(u_{k-1}) = f(u_k)$, and again a stationary discontinuity satisfying the jump condition (2.12). But in this case, the discontinuity violates the entropy condition (2.13), the solution being a "rarefaction shock." The admission of such a physically unrealizable solution is due to the linearization, by which A is a constant matrix, all characteristics are discontinuities, and the finite spread of a rarefaction is not allowed. It should be noted that this problem usually only arises in practice when a stationary rarefaction wave is symmetrically aligned with a cell interface. Examples of entropy "fixes" are discussed in [22,16,17]. We will make note of such a "fix" when necessary for our later application.

There have been many other approaches to the solution of the Riemann problem in recent years, as "upwind" schemes have come to play a more prominent role in numerical shock–capturing. For examples of some of these approaches and their comparisons, the interested reader is referred to [3,12,33,41,46].

33

Chapter 4

Essentially Non-Oscillatory Schemes

In this chapter we develop the notion of essentially non- oscillatory (ENO) shock-capturing schemes. This large class of schemes is designed to approximate weak solutions of hyperbolic systems of conservation laws to an arbitrarily high-order accuracy in smooth regions, as well as avoid a "Gibb's-like" phenomenon near discontinuities. Though our goal is to develop these ENO schemes for application to multi-dimensional systems, we find it prudent to present the fundamental ideas first within a one-dimensional framework, before extending these notions to the approximation of functions of more than one spatial variable.

4.1   Review and Overview

We wish to design high-order accurate schemes for the numerical approximation of weak solutions of hyperbolic systems of conservation laws

$$u_t + f(u)_x = 0 \,, \tag{4.1a}$$

subject to given initial conditions

$$u(x,0) = u^0(x) \,. \tag{4.1b}$$

The function $u = (u^1, u^2, \ldots, u^m)^T$ is a state vector and $f(u)$, the flux, is a vector-valued differentiable function of m components. We assume that the system

34

(4.1a) is *hyperbolic* in the slightly weakened sense that the $m \times m$ Jacobian matrix

$$A(u) = \frac{\partial f}{\partial u}$$

has m real eigenvalues

$$\lambda^1(u) \leq \lambda^2(u) \leq \ldots \lambda^m(u)$$

and a complete set of m linearly independent right eigenvectors $\{r^k(u)\}$ and left eigenvectors $\{l^k(u)\}$. Under suitable normalization, we assume that $l^i r^j = \delta_{ij}$.

We assume that the initial-value problem (4.1) is well-posed in the sense that the solution u depends continuously on the initial data, and that this solution is generically piecewise smooth, with at most a finite number of discontinuities.

As previously noted in Section 3.3, we wish to approximate weak solutions of (4.1) in the "finite-volume" sense. To this end, we note that on any rectangle $(x_{i-1/2}, x_{i+1/2}) \times (t^n, t^{n+1})$, a weak solution of (4.1) satisfies

$$\bar{u}_i^{n+1} = \bar{u}_i^n - \frac{\Delta t^n}{\Delta x_i} \left[\hat{f}_{i+1/2} - \hat{f}_{i-1/2} \right], \tag{4.2a}$$

where

$$\bar{u}_i^n = \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} u(x, t^n)\, dx, \tag{4.2b}$$

is the cell average of u at time $t = t^n$ on the ith interval, and

$$\hat{f}_{i+1/2} = \frac{1}{\Delta t^n} \int_{t^n}^{t^{n+1}} f(u(x_{i+1/2}, t))\, dt, \tag{4.2c}$$

is the flux through the cell interface $x = x_{i+1/2}$. Desiring to approximate equation (4.2a), we write our numerical scheme in the conservation form

$$v_i^{n+1} = v_i^n - \frac{\Delta t}{\Delta x} \left[\bar{f}_{i+1/2} - \bar{f}_{i-1/2} \right], \tag{4.3a}$$

where v_i^n approximates $\bar{u}_i^n$ in (4.2b), and the numerical flux $\bar{f}_{i+1/2}$ is a Lipschitz continuous function of $2k$ variables

$$\bar{f}_{i+1/2} = \bar{f}(v_{i-k+1}^n, \ldots, v_{i+k}^n), \tag{4.3b}$$

35

which is consistent with the flux $f(u)$ in the sense

$$\bar{f}(w, w, \ldots, w) = f(w) . \tag{4.3c}$$

Denote by E_h the numerical solution operator defined by the right-hand side of (4.3a), whose operation can be expressed

$$v_i^{n+1} = (E_h v^n)_i . \tag{4.4}$$

Also, recall that E_h is said to be r^{th}-order accurate if its local truncation error satisfies

$$\bar{u}_i^{n+1} - (E_h \bar{u}^n)_i = O(h^{r+1}) \tag{4.5}$$

for sufficiently smooth u, where $\Delta t = O(h)$.

We desire that the scheme (4.3) approximate (4.2) to high-order accuracy, while being capable of propagating discontinuities without producing spurious $O(1)$ oscillations. The first step taken in this direction was due to the work of Van Leer [36], who proposed a second-order extension of Godunov's scheme. Harten [16,17] later provided a rigorous mathematical foundation for the ideas presented in [36], and developed the class of TVD schemes (See Section 3.2.), which are required to satisfy inequality (3.10). We recall though, that such schemes are inherently limited to second–order accuracy in the L_1 sense. Harten and Osher [25] later developed a class of *non-oscillatory* schemes which could achieve uniform second-order accuracy (See Section 3.2.), again an inherent limit. In order to enable the design of even higher-order accurate schemes, Harten *et al.* [26] introduced the notion of *essentially non-oscillatory* (ENO) schemes, which exclude $O(1)$ Gibbs-like oscillations, but do allow for the production of spurious oscillations on the level of the truncation error. These schemes satisfy, in the scalar case,

$$TV(E_h \bar{v}) \leq TV(v) + O(h^r) . \tag{4.6}$$

Within this framework of numerical schemes which are abstractly defined by (4.6), we will describe a high-order generalization of Godunov's scheme.

36

4.2 ENO Schemes of Godunov Type

Let A_h denote the cell-averaging operator defined by the right-hand side of (4.2a). We respectively denote the sliding average $\bar{w}(x)$ or a discrete value $\bar{w}_i$ as

$$\bar{w}(x) = (A_h\, w)(x)\,, \quad \text{or} \quad \bar{w}_i = (A_h\, w)_i\,. \tag{4.7}$$

As previously noted in Section 3.3, at points of smoothness in $w(x)$, we have the relationship

$$\bar{w}(x) = w(x) + O(h^2)\,,$$

which immediately demonstrates a limit on the order of accuracy if we choose to deal with the solution only on the level of cell averages. Therefore, when $f(u)$ is a nonlinear function of u, the high-order accurate approximation of $\hat{f}_{i+1/2}$ in (4.2b) will require *pointwise* information of the solution to a correspondingly high-order accuracy. We must be able to extract high-order accurate pointwise information from the given $\{\bar{v}_i^n\}$, which are approximations to $\{\bar{u}_i^n\}$, the cell averages (4.2b) of the exact solution. Furthermore, we wish to do so without introducing $O(1)$ spurious oscillations at points of discontinuity. To this end, we define the following reconstruction operator R.

Given $\{\bar{w}_i\}$, cell averages of a piecewise smooth function $w(x)$, denote by $R(x; \bar{w})$ a piecewise polynomial function in x of uniform degree $r-1$ that satisfies :

(i) At all points x for which there is a neighborhood where w is smooth,

$$R(x; \bar{w}) = w(x) + e(x)\, h^r\,. \tag{4.8a}$$

(ii) R is conservative in the sense that A_h is its left-hand inverse.

$$\left(A_h\, R(x\,;\, \bar{w})\right)_i = \bar{w}_i\,. \tag{4.8b}$$

(iii) R is essentially non-oscillatory.

$$TV\left(R(\cdot\,;\bar{w})\right) \le TV\left(w\right) + O(h^r)\,. \tag{4.8c}$$

37

Property (4.8c) represents the manner in which we define the statement that $R(x; w)$ is an essentially non-oscillatory approximation of $w(x)$. Furthermore, this implies that R does not generate $O(1)$ spurious oscillations at points of discontinuity, and any oscillations which do occur must be of the order of the truncation error made by the approximation (4.8a). We have expressed the truncation error in (4.8a) as $e(x)h^r$ rather than simply $O(h^r)$ because the error made by polynomial approximation can always be expressed in this form. For example, if the $(r-1)$-st degree polynomial approximating $w(x)$ is a Taylor expansion about the point x_0, then

$$e(x) = \frac{(x - x_0)^r}{r!\, h^r} \frac{\partial^r w}{\partial x^r}(\xi) \,,$$

for some $\xi = \xi(x)$ between x and x_0.

Using the reconstruction (4.8) and the averaging operator (4.7), we can express the higher-order Godunov-type scheme (3.18) by

$$\bar{v}^{n+1} = E_h \, \bar{v}^n \equiv A_h \, E(\Delta t) \, R(\cdot \, ; v^n) \,, \tag{4.9}$$

where $E(t)$ is the exact evolution operator. These schemes (4.9) are a generalization of Godunov's scheme in the sense that when R represents the piecewise constant reconstruction

$$R(x; v^n) = v_i^n \,, \qquad x_{i-1/2} < x < x_{i+1/2} \,,$$

then (4.9) is identical to Godunov's scheme. If R represents the piecewise linear reconstruction

$$R(x; v^n) = v_i^n + s_i(x - x_i) \,, \qquad x_{i-1/2} < x < x_{i+1/2} \,,$$

such that

$$s_i = \frac{\partial v}{\partial x}(x_i) + O(h) \,,$$

then (4.9) is the abstract form of the second-order accurate extension to Godunov's scheme introduced by Van Leer [36].

38

In the next section, we will review a particular reconstruction algorithm which satisfies (4.8). It is shown in [26] that the incorporation of any such R into the abstract Godunov–type scheme (4.9) will result in a scheme that is consistent with the conservation law, r^{th}–order accurate, and essentially non–oscillatory, the second of which we now demonstrate.

For this purpose, we examine the local truncation error of the scheme, and show that it satisfies (4.5). For this purpose, we consider a single application of the process (3.18), beginning with exact initial data by setting $\{\bar{v}_i^n\} \equiv \{\bar{u}_i^n\}$. Starting with the reconstruction step (3.18a), it follows from (4.8a) that

$$v_h(x, t^n) = u(x, t^n) + e(x) h^r . \qquad (4.10a)$$

For the solution-in-the-small step (4.13b), we use the exact solution operator $E(t)$. This fact and our assumption of the well-posedness of the IVP (4.1) imply that

$$v_h(x, t) = u(x, t) + b(x) h^r , \qquad t^n < t < t^{n+1} , \qquad (4.10b)$$

where $b(x) = O(e(x))$. Since $f(u)$ is a differentiable function of u, it is therefore Lipschitz-continuous in u, and we have

$$f(v_h(x, t)) = f(u(x, t)) + d(x) h^r , \qquad t^n < t < t^{n+1} , \qquad (4.10c)$$

where $d(x) = O(b(x)) = O(e(x))$. Substituting (4.10c) for the integrand of the numerical flux (4.10b), we have, for $t^n < t < t^{n+1}$,

$$\bar{f}_{i+1/2} = \frac{1}{\Delta t} \int_{t^n}^{t^{n+1}} [f(u(x_{i+1/2}, t)) + d(x_{i+1/2}) h^r] dt$$

$$= \hat{f}_{i+1/2} + d(x_{i+1/2}) h^r , \qquad (4.10d)$$

i.e. we have approximated the exact flux (4.2) to $O(h^r)$. If we now put the result in (4.10d) into the conservative form of our scheme (4.3a) along with the exact initial

39

data, we get

$$v_i^{n+1} = \bar{u}_i^n - \frac{\Delta t}{\Delta x}\left\{ \hat{f}_{i+1/2} - \hat{f}_{i-1/2} + \left[d(x_{i+1/2}) - d(x_{i-1/2}) \right] h^r \right\},$$

and thus, we see that our local truncation error, in the sense of cell averages is

$$v_i^{n+1} - \bar{u}_i^{n+1} = \frac{\Delta t}{\Delta x}\left[d(x_{i+1/2}) - d(x_{i-1/2}) \right] h^r. \tag{4.10e}$$

Recalling that we always assume that $\Delta t = O(\Delta x)$, the truncation error in (4.10e) is $O(h^{r+1})$ whenever $d(x)$ is Lipschitz continuous, in which case the scheme (4.9) satisfies (4.5) and is therefore r^{th}–order accurate, as desired.

Since $d(x) = O(e(x))$, it is clear that non-smoothness in $d(x)$ in (4.10e) can only result from non-smoothness in the error coefficient $e(x)$ in (4.10a). We emphasize here that we have defined an rth–order accurate scheme in terms of one application, using exact initial data. Within this context, let $u(x,t)$ be a smooth solution of (4.1) and let us suppose that as $h \to 0$, $\Delta t = O(h)$, the numerical approximation converges pointwise to $u(x,t)$. If $e(x)$ is globally Lipschitz continuous, then the local truncation error in the sense of cell averages is $O(h^{r+1})$. When $e(x)$ fails to be Lipschitz continuous at a point, the local truncation error (4.18f) is only $O(h^r)$. But, assuming that $e(x)$ is globally Lipschitz continuous, then the local truncation error is given by (4.10e). At the end of N time steps, where $N = t/\Delta t = O(1/h)$, we assume the *cumulative error* to be $O(h^r)$, *i.e.*

$$v_i^N = \bar{u}_i^N + O(h^r), \tag{4.11a}$$

in which case we see from (4.10a) that

$$v_h(x,t^N) = R(x;v^N) = u(x,t^N) + O(h^r). \tag{4.11b}$$

Therefore, at the end of a given computation, we have two sets of output data at our disposal : discrete values $\{v_i^N\}$ that approximate $\{\bar{u}_i^N\}$ to $O(h^r)$, and a piecewise polynomial function of x, $R(x;v^N)$, that approximates $u(x,t^N)$ to $O(h^r)$.

40

4.3 ENO Reconstruction

From the work of the previous section, we see that, in the abstract form (4.9), the design of high–order accurate Godunov–type schemes boils down to a problem on the level of approximation of functions. In this section, we review the scalar reconstruction ideas presented in [26]. We then describe a particular reconstruction operator R which we will presently employ and then extend to two dimensions in Chapter 5. For this purpose, we introduce $Q_n(x;w)$, an nth–degree piecewise polynomial function of x that interpolates w at the points $\{x_i\}$, *i.e.*

$$Q_n(x_i;w) = w(x_i), \tag{4.12a}$$

$$Q_n(x;w) \equiv q_{n,i}(x;w), \qquad x_i \leq x \leq x_{i+1}, \tag{4.12b}$$

where $q_{n,i}$ is a polynomial in x of degree n.

We take each $q_{n,i}$ to be the (unique) n^{th}–degree polynomial that interpolates $w(x)$ at $n+1$ successive points $\{x_j\}$ which include x_i and x_{i+1}. Denoting by $j(i)$ the left–most index in this "stencil" of $n+1$ points, we write

$$q_{n,i}(x_j;w) = w(x_j), \qquad j = j(i),\ldots,j(i)+n; \tag{4.13a}$$

$$1-n \leq j(i) - i \leq 0. \tag{4.13b}$$

Clearly, there are exactly n such polynomials corresponding to the n possible choices of $j(i)$ subject to the condition (4.13b). This freedom will be used to assign to (x_i, x_{i+1}) a stencil of $n+1$ points satisfying (4.13). We will make this choice subject to the condition that $w(x)$ be "smoothest" on the chosen stencil in some asymptotic sense.

Though the smoothness of a given function $w(x)$ is ordinarily determined analytically in the sense of derivatives of w, this information is, in general, not available for a numerical solution, where only discrete values are known. However, information relevant to smoothness can be extracted from a table of divided differences of

41

w. Employing the standard notation, the kth divided difference of w can be defined recursively by

$$w[x_j] = w(x_j) ; \tag{4.14a}$$

$$w[x_j,\ldots,x_{j+k}] = \frac{w[x_{j+1},\ldots,x_{j+k}] - w[x_j,\ldots,x_{j+k-1}]}{x_{j+k} - x_j} . \tag{4.14b}$$

It is well known that if w has k continuous derivatives on $x_j \leq x \leq x_{j+k}$ then

$$w[x_j,\ldots,x_{j+k}] = \frac{1}{k!}\frac{d^k}{dx^k}w(\xi) , \quad x_j \leq \xi \leq x_{j+k} . \tag{4.14c}$$

Furthermore, if w has a jump discontinuity in the pth derivative in this interval, where $0 \leq p \leq k$, then

$$w[x_j,\ldots,x_{j+k}] = O(h^{p-k}[w^{(p)}]) , \tag{4.14d}$$

where $[w^{(p)}]$ denotes the jump in the pth derivative. Relationships (4.14c–d) suggest that

$$\big|\, w[x_j,\ldots,x_{j+k}]\,\big| ,$$

the magnitude of a kth divided difference, provides an asymptotic measure of the smoothness of w in (x_j, x_{j+k}), in the following sense. Suppose that w is smooth in the interval (x_0, x_k) but is discontinuous in (x_1, x_{k+1}). Then for h sufficiently small, we expect

$$\big|\, w[x_0,\ldots,x_k]\,\big| < \big|\, w[x_1,\ldots,x_{k+1}]\,\big| ,$$

and hence these divided differences can serve as a tool to compare the relative smoothness of w in various stencils. Therefore, we assign a stencil of $n+1$ points for each $q_{n,i}$ in (4.13) by determining that interval in which w has the "smallest divided differences."

Since we always assume any stencil we choose to be contiguous, we assign a particular stencil by determining its left–most index $j(i)$. The simplest algorithm

42

for determining the smoothest stencil is by the smallest magnitude of the highest order divided difference on all stencils subject to (4.13b), *i.e.* choose $j(i)$ so that

$$| w[x_{j(i)}, \ldots, x_{j(i)+n}] | = \min_{i-n+1 \leq j \leq i} \{ | w[x_j, \ldots, x_{j+n}] | \} . \qquad (4.15)$$

Clearly, (4.15) selects the "smoothest" stencil, provided that h is sufficiently small. However, in order to make a sensible selection of stencil in the "pre–asymptotic" range, we prefer to use the following hierarchical algorithm.

Let $j_k(i)$ denote the left–most index of a chosen "smoothest" $(k+1)$–point stencil for the interval (x_i, x_{i+1}). Denote this stencil

$$\{ x_{j_k(i)}, \ldots, x_{j_k(i)+k} \}, \qquad k = 1, 2, \ldots, n . \qquad (4.16a)$$

Since any stencil must include $\{x_i, x_{i+1}\}$, our recursive algorithm begins $(k=1)$ by setting

$$j_1(i) = i . \qquad (4.16b)$$

In order to choose $j_{k+1}(i)$, $k = 1, \ldots, n-1$, we consider as candidates the two stencils

$$\{ x_{j_k(i)-1}, \ldots, x_{j_k(i)+k} \} \quad \text{or} \quad \{ x_{j_k(i)}, \ldots, x_{j_k(i)+k+1} \} , \qquad (4.16c)$$

which are obtained by adding a point to the left or right, respectively, of the previously determined stencil. We select the one in which w is relatively smoother, *i.e.* the one in which the $(k+1)$–th order divided difference is smaller in magnitude

$$j_{k+1}(i) = \begin{cases} j_k(i) - 1, & \text{if } | w[x_{j_k(i)-1}, \ldots, x_{j_k(i)+k}] | < | w[x_{j_k(i)}, \ldots, x_{j_k(i)+k+1}] | , \\ j_k(i), & \text{otherwise} . \end{cases}$$

$$(4.16d)$$

Finally, we set $j(i) = j_n(i)$.

Appealing to well–known results pertaining to Newton interpolation, we see that a kth–degree interpolating polynomial $q_{k,i}(x)$, $k = 1, \ldots, n$, corresponding to the stencil (4.16a) selected by the above recursive algorithm, satisfies the relation

$$q_{k,i}(x) = q_{k-1,i}(x) + w[x_{j_k(i)}, \ldots, x_{j_k(i)+k}] \prod_{j=j_k(i)}^{j_k(i)+k-1} (x - x_j) . \qquad (4.17)$$

43

This shows that the choice made in (4.16d) selects $q_{k,i}(x)$ to be the one that deviates least from $q_{k-1,i}(x)$. It is this property (4.17) that makes the hierarchical algorithm (4.16) meaningful in the "pre-asymptotic" range.

Now, if we assume that $w(x)$ has only a finite number of discontinuities, then for h sufficiently small, there are $n+1$ points of smoothness between any two discontinuities. Consequently, if (x_i, x_{i+1}) is an interval in which $w(x)$ is smooth, then there is at least one choice of $j(i)$ such that the values $\{w(x_j)\}$ will represent smooth interpolation data on $\{x_{j(i)}, \ldots, x_{j(i)+n}\}$, as in the example $(n = 3)$ in Figure 6a. Therefore, $q_{n,i}(x; w)$ in (4.12b) is a polynomial that interpolates data from an interval in which $w(x)$ is smooth, and from classical interpolation results each such polynomial satisfies

$$\frac{d^k}{dx^k} q_{n,i}(x; w) = \frac{d^k}{dx^k} w(x) + O(h^{n-k+1}), \qquad x_i < x < x_{i+1},$$

for $k = 0, \ldots, n$. It immediately follows that, for h sufficiently small and wherever $w(x)$ is smooth,

$$\frac{d^k}{dx^k} Q_n(x; w) = \frac{d^k}{dx^k} w(x) + O(h^{n-k+1}), \qquad k = 0, \ldots, n. \quad (4.18a)$$

Furthermore, it follows from (4.18a), with $k = 0$, that wherever $w(x)$ is smooth,

$$TV\left(Q_n(x; w)\right) \leq TV\left(w(x)\right) + O(h^{n+1}). \tag{4.18b}$$

As for an interval (x_i, x_{i+1}) in which $w(x)$ has a discontinuity, it is heuristically argued in [18,19], that for h sufficiently small, although the local interpolating polynomial may oscillate wildly outside the ith interval, it is nonetheless monotone within this interval, regardless of the choice of interpolation stencil (Figure 6b). Though we can only justify that $q_{n,i}(x; w)$ approximates $w(x)$ to $O(1)$ in this interval, the local approximations in the adjoining intervals retain high-order accuracy, due to the adaptive stencil algorithm. Using the discrete function $\{w(x_j)\}$ of the two previous illustrations, an ENO polynomial approximation to this function might look like Figure 6c.

44

ENO Interpolation

Example : $n = 3$, $w(x)$ smooth in (x_i, x_{i+1}).
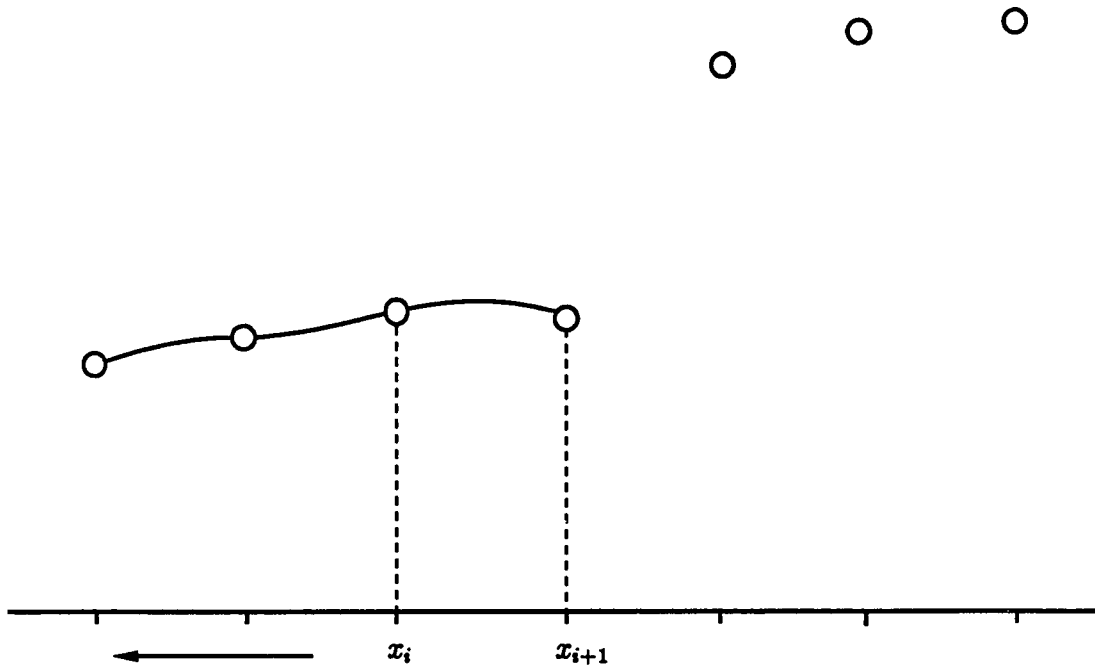
Figure 6a. Algorithm (4.16) determines a smooth interpolation stencil

to the left for $q_{3,i}(x; w)$ in (x_i, x_{i+1}).

45

ENO Interpolation

Example : $n = 4$, $w(x)$ discontinuous in (x_i, x_{i+1}).
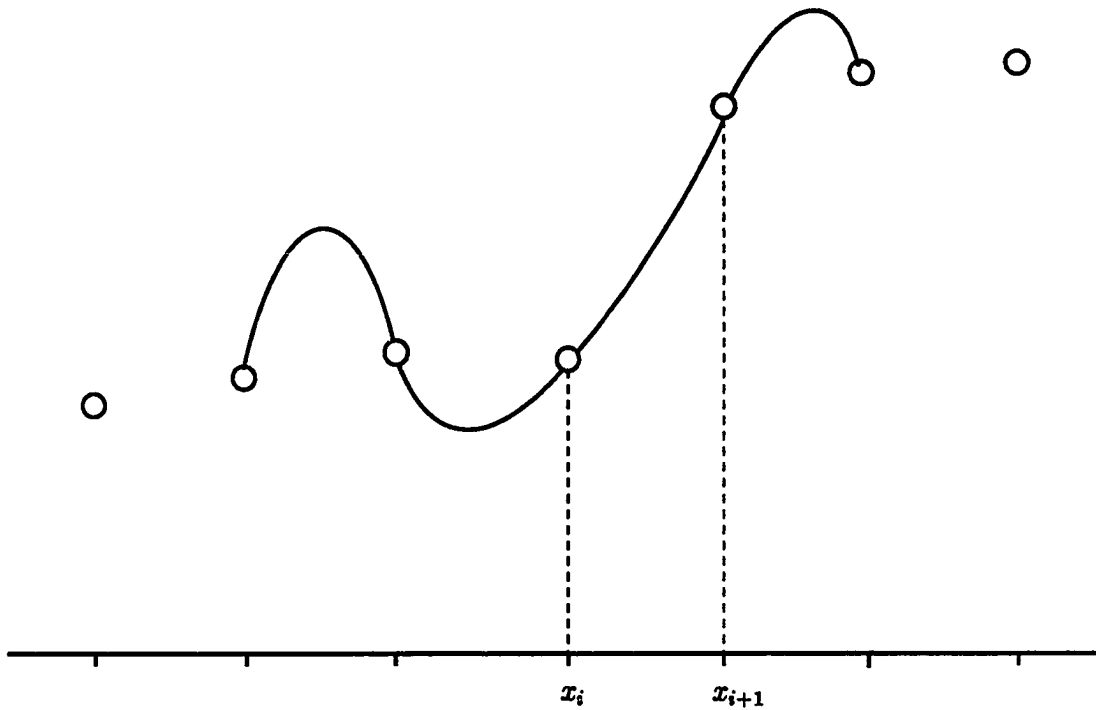
Figure 6b. $q_{4,i}(x; w)$ is oscillatory, but is monotone in (x_i, x_{i+1}).

46
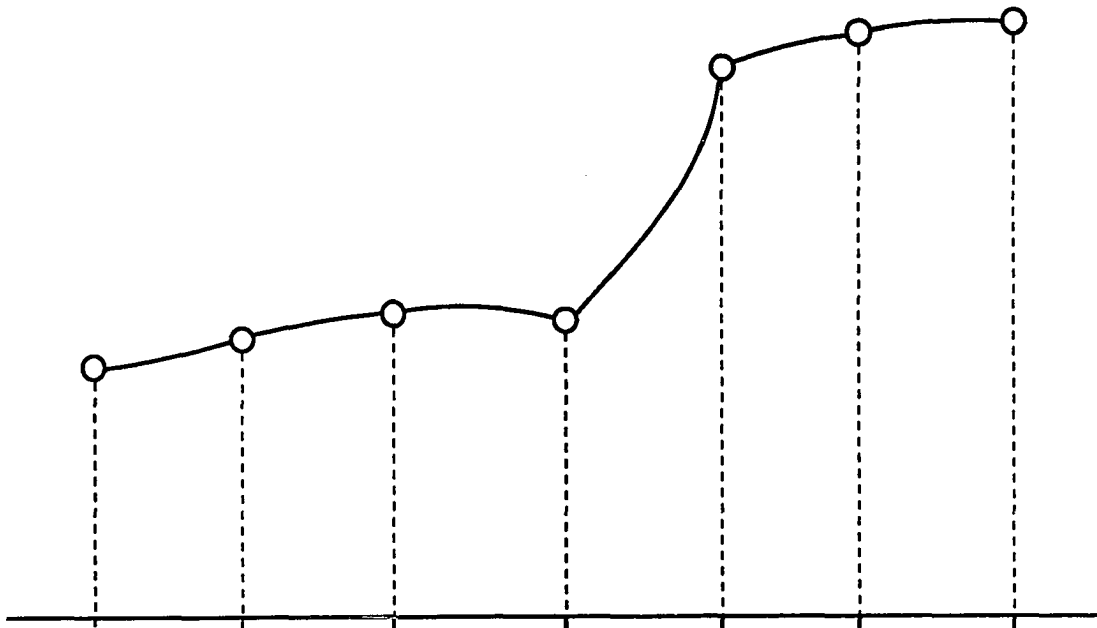
Figure 6c. $Q_n(x;w)$ is a piecewise polynomial which interpolates w

in an essentially non–oscillatory fashion

47

We will now employ the above interpolation ideas to develop a particular scalar reconstruction operator which satisfies the desired properties (4.8). We are initially given a discrete set of cell averages $\{\bar{w}_i\}$ of a piecewise smooth function w

$$\bar{w}_i = \frac{1}{h_i} \int_{x_{i-1/2}}^{x_{i+1/2}} w(x)\, dx \,, \tag{4.19}$$

where $h_i = x_{i+1/2} - x_{i-1/2}$. We define the *primitive function* $W(x)$ associated with w by

$$W(x) = \int_{x_0}^{x} w(\xi)\, d\xi \,. \tag{4.20}$$

Seeking a relationship between pointwise values of $W(x)$ and the discrete values $\{\bar{w}_i\}$ we see immediately from the definitions (4.19) and (4.20) that

$$h_i\, \bar{w}_i = W(x_{i+1/2}) - W(x_{i-1/2}) \,, \tag{4.21}$$

and we can therefore establish such a relationship at the cell interfaces:

$$W(x_{i+1/2}) = \sum_{k=k_0}^{i} h_k\, \bar{w}_k \,. \tag{4.22}$$

Now, since the definition (4.20) clearly implies

$$w(x) = \frac{d}{dx} W(x) \,, \tag{4.23}$$

if we approximate $W(x)$, in the afore–mentioned manner, by an rth–degree piecewise polynomial $Q_r(x;W)$ which interpolates W at the values given by (4.22), we can then obtain an approximation of $w(x)$ by defining our reconstruction operator

$$R(x;\bar{w}) = \frac{d}{dx} Q_r(x;W) \,. \tag{4.24}$$

Using the interpolation results (4.18) and the heuristic argument referenced in relation with Figure 6b, it follows from its definition (4.24) that R satisfies (4.8) (See [26] for details.).

48

It is ultimately the adaptive–stencil algorithm that enables this reconstruction (for sufficiently small h) to be high–order accurate in any interval where $w(x)$ is smooth, even if that interval is near one in which $w(x)$ is discontinuous. We also note that this procedure does not require uniformity of the mesh and that we shall extend this scalar reconstruction to vectors in Section 4.5. This procedure will be henceforth referred to as "reconstruction by primitive." As an example, if the function $\{w(x_j)\}$ of the previous examples was available to us in the form of cell averages, then the reconstruction-by–primitive procedure would yield a piecewise polynomial approximation resembling that of Figure 6d. The discontinuities at interval interfaces are the result of the differentiation (4.24) and the adaptive interpolation which produces the piecewise polynomial $Q_r(x\,;\,W)$. The use of these interface values in the solution–in–the–small step is described in the following section.

Finally, we recall the remark in the previous section, concerning the possible discontinuity of the error coefficient $e(x)$ in (4.8a). We now see that this can occur at points where there is a change in the stencil of the associated interpolation, due to the algorithm (4.16). Thus, when the stencil becomes a discontinuous function in x, the local truncation error of our scheme becomes $O(h^r)$ at such points and therefore $O(h^{r-1})$ cumulatively. However, due to the ENO property of the scheme, we expect the number of such points to remain bounded as $h \to 0$. If indeed this is the case, then we expect the cumulative error of our scheme to be $O(h^{r-1})$ in the L_∞ norm but to remain $O(h^r)$ in the L_1 norm. The interested reader is referred to [18,19] for results of numerical experiments which test this reconstruction operator for both its accuracy as well as its non–oscillatory properties.
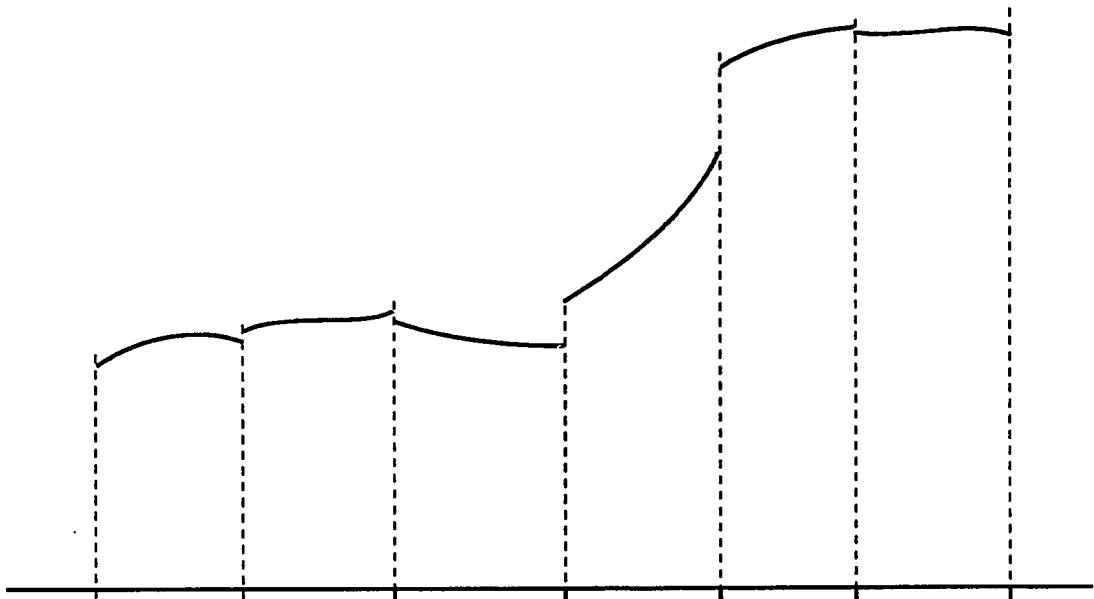
Figure 6d. $R(x;\bar{w})$ is a piecewise polynomial approximation to $w(x)$

as produced by the reconstruction (4.24).

50

4.4 Time Accuracy: Scalar Conservation Laws

To this point, we have considered the details of the design of high–order Godunov–type schemes in terms of spatial accuracy alone, due in large part to the fact that, in the abstract form (4.9), the scheme relies upon the exact evolution operator $E(t)$ of the IVP (4.1). Before extending the scalar reconstruction procedure of the previous section to vector functions, we find it prudent at this time to consider the issue of time accuracy, and therefore do so on the level of solutions of scalar conservation laws.

After reconstructing the discrete set $\{\bar{v}_i^n\}$ of cell averages of the numerical solution at time t^n to a desired order of pointwise accuracy, we are then required to solve exactly

$$v_t + f(v)_x = 0 , \qquad (4.25a)$$

with the piecewise polynomial initial data

$$v(x, t^n) = R(x; \bar{v}^n) , \qquad (4.25b)$$

in the time strip $(-\infty, \infty) \times (t^n, t^n + \Delta t)$, for Δt small. (We have dropped the "h" subscript for convenience.)

Due to the adaptive selection of stencil and subsequent differentiation that are required in the reconstruction procedure, the piecewise polynomial $R(x; \bar{v}^n)$ can be discontinuous at the cell interfaces $\{x_{i+1/2}\}$. In a smooth region of the solution, the relative size of these jump discontinuities will be of the order of the local interpolation error, whereas in the vicinity of a discontinuity, the size of the jump will be $O(1)$. A global description of the solution $v(x, t)$ of (4.25) can be quite complicated. However, in the conservation form (4.3a), our scheme calls for the exact evaluation of the "numerical flux" (4.3c). Therefore all we need is $v(x_{i+1/2}, t)$, the solution–in–the–small at each interface, in a small time interval.

Now, in the case $r = 1$, where $R(x; \bar{v}^n)$ is the piecewise–constant distribution

determined by the cell averages $\{\bar{v}_i^n\}$ themselves, the abstract scheme (4.9) is identical to Godunov's scheme. Hence, $v(x_{i+1/2}, t)$ is determined by the the solution of the local Riemann problem at $x = x_{i+1/2}$, *i.e.* we solve

$$v_t + f(v)_x = 0, \qquad v(x, t^n) = \begin{cases} v_i^n, & x < x_{i+1/2} \\ v_{i+1}^n, & x > x_{i+1/2}, \end{cases} \tag{4.26}$$

on the rectangle $(x_i, x_{i+1}) \times (t^n, t^n + \Delta t)$, for Δt small. As the Riemann problem is well-understood in the scalar case (*e.g.* [53]), the self-similar solution $v(x, t)$ of (4.26) can be determined exactly.

However, when $R(x; \bar{v}^n)$ is a piecewise polynomial function of higher degree we cannot, in general, express the solution of (4.25) in a simple closed form, though an inspection of its structure may prove insightful. We turn now to the *"generalized Riemann problem,"* which we define for our purposes as

$$v_t + f(v)_x = 0, \qquad v(x, t^n) = \begin{cases} v_i(x, t^n), & x < x_{i+1/2} \\ v_{i+1}(x, t^n), & x > x_{i+1/2}, \end{cases} \tag{4.27}$$

where $v_i(x, t^n)$ is the polynomial of degree $r - 1$ determined by $R(x; \bar{v}^n)$ in the interval $x_{i-1/2} < x < x_{i+1/2}$, with $r \geq 2$. Though a closed-form solution $v(x, t)$ is generally not available for (4.27), its structure can be described, for Δt sufficiently small, as two sections of smoothness separated by a "wave" emerging from the discontinuity at $x_{i+1/2}$. The term "wave" is used loosely here to describe a shock, a rarefaction fan, or a contact discontinuity, depending on the nature of f. Let $v_i(x, t)$, $t^n < t < t^{n+1}$, denote the section of smoothness of the solution of (4.27) that is determined by the initial polynomial data in $(x_{i-1/2}, x_{i+1/2})$. The solution $v(x, t)$ at $x = x_{i+1/2}$ of (4.27) can then be described in terms of $v_i(x, t)$, $v_{i+1}(x, t)$, and the "wave" emanating from $x = x_{i+1/2}$ at $t = t^n$, depending on whether the "wave" remains to the left or right of $x = x_{i+1/2}$, or whether the "wave" covers $x = x_{i+1/2}$. (See [26] for details.) Due to the non-uniformity of the initial data, the solution of (4.27) will generally not possess the self-similarity features of the

52

Riemann problem (4.26). The sections of smoothness are not necessarily constant, and the trajectory of the "wave" may not be a straight line. The solution $v(x_{i+1/2}, t)$ is therefore variable for $t^n < t < t^n + \Delta t$, and without a method to determine exactly what this arbitrary variation may be, the "numerical flux" (4.3c), in general, cannot be determined in an exact closed form. Therefore, the actual implementation of our numerical scheme for second- and higher-order accurate results will require a discretization in time to a correspondingly high order of accuracy.

There are two main approaches to the issue of time discretization. One is the *fully-discrete* or Taylor-series approach, which relies on the knowledge that, although an exact solution is not available for the generalized Riemann problem (4.27), we can obtain a local Taylor expansion of the solution to any desired order of accuracy. (See [1,14].) However, this approach requires a local Taylor expansion of $v_i(x, t)$ in $(x_{i-1/2}, x_{i+1/2}) \times (t^n, t^{n+1})$. The purely spatial derivatives required for this expansion clearly come from the reconstruction $R(x; \bar{v}^n)$. Any derivative involving time must come from a successively ordered differentiation of the PDE (4.25a), *i.e.*

$$
\begin{aligned}
v_t &= -f' v_x \\
v_{xt} &= -[f'' (v_x)^2 + f' v_{xx}] \\
v_{tt} &= -[f'' v_t v_x + f' v_{xt}]
\end{aligned}
\tag{4.28}
$$

etc.

This approach to the time integration has the advantage that only one spatial reconstruction step is required per update from time t^n to to t^{n+1}. However, application of this time discretization to systems of conservation laws can be quite complicated. One needs only to look at the algorithm (4.28) to see that if v and $f(v)$ in (4.25a) are vectors, then f' is a matrix, f'' is a tensor, and so on. This process becomes particularly excessive as multi-dimensional systems are then considered.

53

Therefore, we choose another approach to the issue of time integration.

We now consider the *semi-discrete* or "method-of-lines" approach to our time discretization. For our purposes, this approach amounts to a technique for replacing (4.25a), a partial differential equation in the two variables x and t, with an ordinary differential equation in the variable t. The region of integration $(-\infty, \infty) \times (t^n, t^{n+1})$ is divided into strips by lines $t = \text{constant}$, and the operation of the discrete operator in x is retained along each of these lines.

We note here that the numerical update scheme (4.3a) which we have studied to this point is a one-step forward Euler formula, in which the solution is updated from time t^n directly to t^{n+1}. However, due to our semi-discrete approach, this will no longer be the case. In fact, we will require a modification of the abstract form (4.9) of our Godunov-type scheme. Therefore, at this time, we develop the details which are pertinent to our particular method-of-lines application.

Again, we begin with the requirement that we solve exactly the IVP (4.25). Integrating the partial differential equation (4.25a) over the interval $(x_{i-1/2}, x_{i+1/2})$, we see that the exact solution $v(x, t)$ satisfies

$$\frac{\partial}{\partial t} \bar{v}_i(t) = -\frac{1}{\Delta x_i} \left[f(v(x_{i+1/2}, t)) - f(v(x_{i-1/2}, t)) \right] . \qquad (4.29a)$$

The notation $\bar{v}_i(t)$ is used here to stress the fact that a cell average is now considered a continuous function in time. We now treat (4.29a) as an ordinary differential equation for the purpose of time discretization. Along any $t = \text{constant}$ line, the right-hand side of (4.29a) is strictly a spatial operation in the unknown v, and we rewrite this equation, for fixed t, in the abstract operator-product form

$$\frac{\partial}{\partial t} \bar{v}_i = (\mathcal{L} v)_i . \qquad (4.29b)$$

Now, in order to evaluate $\mathcal{L} v$ for a fixed t, we must know how the exact solution $v(x, t)$ varies in time. As previously discussed, this information is only available when the initial distribution (4.25b) is piecewise constant. Therefore, for the general

54

higher–order case, let L denote the ENO spatial operator which approximates $\mathcal{L}$ to r^{th}–order, *i.e.*

$$L\,\bar{v} \;=\; \mathcal{L}\,v \;+\; O(h^r)\,,\qquad\qquad (4.29c)$$

wherever v is sufficiently smooth. The use of $\bar{v}$ as the operand of L signifies the dependence of L on the reconstruction operator R which acts upon given cell averages of the approximate solution. We can then define the approximate forward Euler method for (4.29b) in abstract form by

$$w \;=\; T\,\bar{v} \;=\; \bar{v} \;+\; \Delta t\, L\,\bar{v}\,.\qquad\qquad (4.29d)$$

Clearly, T is a nonlinear spatial operator.

Ultimately, we desire a numerical update scheme

$$\bar{v}_i^{n+1} \;=\; (\,S\,\bar{v}^n\,)_i\,,\qquad\qquad (4.30a)$$

where the operator S depends on T. We want this scheme to be an rth–order approximation of (4.29b), *i.e.* if $v(x,t)$ is an exact solution of (4.29b), then

$$\big(\,A_h\,v(x,t^{n+1})\,\big)_i \;-\; (\,S\,\bar{v}^n\,)_i \;=\; O(h^{r+1})\,,\qquad\qquad (4.30b)$$

thereby requiring a correspondingly high order time discretization of (4.29d). Furthermore, because we have effectively "separated" the spatial and temporal operators, it is also desirable that the action of the discrete time operator itself not induce any oscillatory behavior into the solution. In [51,50], two such classes of high–order time discretizations were developed, one a class of multi–level methods, the other of Runge–Kutta type. These time discretizations are required to be TVD, in the sense that

$$TV(\,S\,\bar{v}\,) \;\le\; TV(\,T\,\bar{v}\,)\,,\qquad\qquad (4.30c)$$

under suitable restrictions on Δt.

Because single–step methods are self–starting and less storage intensive, we choose to employ the class of Runge–Kutta type time discretizations developed

55

in [50]. To this end, we replace the abstract forward Euler operation (4.29d) with a general explicit Runge–Kutta method, which can be written

$$v_i^{(k)} = v^{(0)} + \Delta t \sum_{l=0}^{k-1} c_{kl} \, (L \, v^{(l)})_i \,, \qquad k = 1, 2, \ldots, p \,, \qquad (4.31a)$$

where

$$v_i^{(0)} = v_i^n \,, \qquad v_i^{(p)} = v_i^{n+1} \,.$$

It is well-known that $(p+1)^{\text{th}}$-order accurate methods of the form (4.31a) exist for $p \leq 3$, p^{th}-order methods for $p=4,5,6$, or $(p-1)^{\text{th}}$-order methods for $p=7,8$. For details, see e.g. [8].

For the purpose of obtaining a condition which fulfills the requirement (4.30c), it is shown in [50], that (4.31a) can be equivalently written

$$v_i^{(k)} = \sum_{l=0}^{k-1} [\, \alpha_{kl} \, v_i^{(l)} + \beta_{kl} \, \Delta t \, (L \, v^{(l)})_i \,] \,, \qquad k = 1, 2, \ldots, p \,. \qquad (4.31b)$$

For the classical 4th-order Runge–Kutta methods, the constants c_{kl} in (4.31a) are all positive. However, in the form (4.31b), some of the β_{kl} may well be negative, which would hinder the TVD analysis. Therefore, in order to obtain a TVD condition, whenever β_{kl} is negative, we replace L in (4.31b) with $\tilde{L}$, which approximates $\mathcal{L}$ in the "adjoint" equation

$$\frac{\partial}{\partial t} \bar{v}_i = -(\, \mathcal{L} \, v)_i \,, \qquad (4.32a)$$

and satisfies

$$\tilde{w} = \tilde{T} \, \bar{v} = \bar{v} - \Delta t \, \tilde{L} \, \bar{v} \,, \qquad (4.32b)$$

and

$$\tilde{L} \, \bar{v} = \mathcal{L} \, v + O(h^r) \,, \qquad (4.32c)$$

wherever v is sufficiently smooth.

Now, if the forward Euler version (4.29d) is total–variation stable under the CFL–like restriction

$$\frac{\Delta t}{\Delta x} \max_v |f'(v)| \leq K \,,$$

56

then the scheme (4.31b) is TVD if

$$\frac{\Delta t}{\Delta x} \max_v |f'(v)| \leq C_r K , \tag{4.33a}$$

where

$$C_r = \min_{k,l} \frac{\alpha_{kl}}{|\beta_{kl}|} , \tag{4.33b}$$

and if L in (4.31b) is replaced by $\tilde{L}$ in (4.32) whenever β_{kl} is negative. The coefficients α_{kl} and β_{kl} are chosen such that (4.31b) is of the highest possible order and such that the stability criterion (4.33) is optimal. The quantity C_r will be referred to as the "CFL coefficient." The interested reader is referred to [51,50] for details.

We repeat here the "optimal" Runge–Kutta schemes presented in [50], with their respective CFL coefficients, which we intend to employ throughout the remainder of this dissertation :

i) Second-order case : $C_r = 1.0$

$$v_i^{(1)} = v_i^{(0)} + \Delta t \, (L \, v^{(0)})_i \tag{4.34a}$$

$$v_i^{(2)} = \tfrac{1}{2} v_i^{(0)} + \tfrac{1}{2} v_i^{(1)} + \tfrac{1}{2} \Delta t (L \, v^{(1)})_i$$

ii) Third-order case : $C_r = 1.0$

$$v_i^{(1)} = v_i^{(0)} + \Delta t \, (L \, v^{(0)})_i$$

$$v_i^{(2)} = \tfrac{3}{4} v_i^{(0)} + \tfrac{1}{4} v_i^{(1)} + \tfrac{1}{4} \Delta t (L \, v^{(1)})_i \tag{4.34b}$$

$$v_i^{(3)} = \tfrac{1}{3} v_i^{(0)} + \tfrac{2}{3} v_i^{(2)} + \tfrac{2}{3} \Delta t (L \, v^{(2)})_i$$

iii) Fourth-order case : $C_r = 0.87$

$$v_i^{(1)} = v_i^{(0)} + \tfrac{1}{2} \Delta t (L \, v^{(0)})_i$$

$$v_i^{(2)} = \tfrac{2}{5} v_i^{(0)} - \tfrac{2}{5} \Delta t (\tilde{L} \, v^{(0)})_i + \tfrac{3}{5} v_i^{(1)} + \tfrac{3}{5} \Delta t (L \, v^{(1)})_i \tag{4.34c}$$

$$v_i^{(3)} = \tfrac{831}{20000} v_i^{(0)} - \tfrac{1769}{40000} \Delta t (\tilde{L} \, v^{(0)})_i + \tfrac{4669}{20000} v_i^{(1)} - \tfrac{161}{600} \Delta t (\tilde{L} \, v^{(1)})_i +$$

$$\tfrac{29}{40} v_i^{(2)} + \tfrac{5}{6} \Delta t (L \, v^{(2)})_i$$

$$v_i^{(4)} = \tfrac{1}{3} v_i^{(0)} + \tfrac{1}{3} v_i^{(1)} + \tfrac{1}{3} \Delta t (L \, v^{(1)})_i + \tfrac{1}{3} v_i^{(3)} + \tfrac{1}{6} \Delta t (L \, v^{(3)})_i$$

We note that (4.34a) is equivalent to the classical Heun's method or modified Euler method (See *e.g.* [9].). For remarks on (4.34b), (4.34c), as well as other third, fourth, and fifth order methods, see [50]. Unlike the Taylor–series method, the ideas in the method–of–lines approach described by (4.29)-(4.32) can be extended in a straight–forward fashion to systems of conservation laws, as well as systems in more than one spatial dimension. (See Chapter 5.)

We now describe the actual form of our numerical scheme. The most natural high–order approximation to (4.29a) would be to replace the flux $f(v(x_{i+1/2}, t))$ with $f(R(x_{i+1/2}; \bar{v}(t)))$, where the argument of f involves the ENO reconstruction of the cell averages $\{\bar{v}_i(t)\}$ at time t. However, since the piecewise polynomial determined by R can be discontinuous at the $\{x_{i+1/2}\}$, we cannot rigorously define $L\bar{v}$ in terms of such a multiple–valued quantity. Instead, we a choose a form which takes a local discontinuity into account, *i.e.* we make the approximation

$$f(v(x_{i+1/2}, t)) \approx f^R[R(x_{i+1/2} - 0; \bar{v}(t)), R(x_{i+1/2} + 0; \bar{v}(t))],$$

where $f^R[u_L, u_R]$ is the flux across $x=0$ associated with the solution of the Riemann problem whose initial states are u_L and u_R. The notation $q(x-0), q(x+0)$ denotes the limiting values of q at x from the left and right, respectively. Thus, along each $t = $ constant line, our first step is to apply the reconstruction operator R to the $\{\bar{v}(t)\}$. We then "partially update" the solution to the next time line, using a numerical flux computation which involves the solution of the local Riemann problem whose initial values are determined by the "endpoint" values of $R(x; \bar{v}(t))$ in each interval $(x_{i-1/2}, x_{i+1/2})$.

In fully–discrete form, we write our scheme as the Runge–Kutta method

$$\bar{v}_i^{(k)} = \sum_{l=0}^{k-1} [\alpha_{kl} \bar{v}_i^{(l)} + \beta_{kl} \Delta t (L\bar{v}^{(l)})_i], \qquad k = 1, 2, \ldots, p, \qquad (4.35a)$$

$$\bar{v}_i^{(0)} = \bar{v}_i^n, \qquad \bar{v}_i^{(p)} = \bar{v}_i^{n+1}, \qquad (4.35b)$$

58

where

$$(L\, v^{(k)})_i = -\frac{1}{\Delta x_i} \left[\bar{f}^{(k)}_{i+1/2} - \bar{f}^{(k)}_{i-1/2} \right] ,$$ (4.35c)

and

$$\bar{f}^{(k)}_{i+1/2} = f^R(\, R(x_{i+1/2} - 0\,;\, v^{(k)}) \,,\, R(x_{i+1/2} + 0\,;\, v^{(k)}) \,) \,.$$ (4.35d)

Both of the time–stepping approaches discussed in this section have been tested in numerical experiments by other authors [2,25,26,50] to compute solutions of one–dimensional, scalar conservation laws in both the linear and non–linear cases. Promising results were obtained for cases of convex and non–convex fluxes, giving numerical validation of the order of accuracy in time–accurate updates of smooth solutions as well as confirming monotonic behavior in discontinuous solutions.

Furthermore, in addition to computing numerical fluxes via the exact Riemann flux $f^R(v_L, v_R)$, experiments were also conducted in [26] using the approximate "Roe flux" which, in the scalar case, is

$$f^{ROE}(v_L, v_R) = \frac{1}{2} \left[f(v_L) + f(v_R) - |\bar{a}| \, (v_R - v_L) \right] ,$$ (4.36a)

where

$$\bar{a} = \begin{cases} (f(v_R) - f(v_L)) / (v_R - v_L), & \text{if } v_L \neq v_R \\ a(v_L), & \text{if } v_L = v_R , \end{cases}$$ (4.36b)

employing an appropriate "entropy fix" where necessary. These tests also met with encouraging results. Due to this previous thorough numerical investigation, we omit any one–dimensional scalar results from our discussion.

4.5 Systems of Conservation Laws

In this section, we extend the scalar reconstruction procedure of Section 4.3 and the method–of–lines time discretization of Section 4.4 to solutions of systems of hyperbolic conservation laws. To this end, we now reconsider the IVP (4.1), which we write as

$$U_t + F(U)_x = 0 ,$$ (4.37a)

59

$$U(x,0) = U^0(x) , \qquad\qquad (4.37b)$$

using upper case notation to emphasize the fact that (4.47a) represents a system of m equations. We denote the vector–valued solution $U = (u^1, u^2, \ldots, u^m)^T$, and therefore $F(U)$ is also a vector of m components. As always, we assume that the initial data $U^0(x)$ are such that the state vector $U(x,t)$ is a piecewise smooth function of x with at most a finite number of discontinuities.

Clearly, a cell average of a vector U is simply a vector whose components are the cell averages of the scalar components of U,

$$\bar{U}_i = (\mathbf{A}_h U)_i = \left((\mathbf{A}_h u^1)_i , (\mathbf{A}_h u^2)_i , \ldots, (\mathbf{A}_h u^m)_i \right)^T .$$

We now wish to develop a spatial operator, denoted by $\mathbf{R}(x; \bar{U})$, which will reconstruct a set $\{\bar{U}_i\}$ of vector–valued cell averages to high–order pointwise accuracy. It would seem natural, purely from an approximation theory viewpoint, to reconstruct the set $\{\bar{U}_i\}$ by applying the scalar reconstruction R to each of the component sets $\{ (\mathbf{A}_h u^j)_i \}$. However, this approach is valid only if we disregard the time–dependence of $U(x,t)$ which allows for discontinuities to collide with each other. In the scalar case, where only one characteristic family exists, a potential collision of discontinuities is not a problem. But when $U(x,t)$ is a solution of a coupled system of equations, such a solution can admit the collision of discontinuities of the same or of different families, as well as their collision with boundaries, e.g. solid walls. In the vicinity of such collisions, in the solution of more than one dependent variable, a component–wise reconstruction may develop spurious oscillations during this brief encounter which do not dissipate as the discontinuities then distance themselves from one another. Numerical experiments to demonstrate this potential problem can be found in [26]. In the following, we describe an algorithm to reconstruct the vector–valued solution $U(x,t)$ from its cell averages $\{\bar{U}_i^n\}$ which avoids this difficulty.

60

We begin by considering the constant coefficient case of (4.37), *i.e.* $F(U) = AU$, where A is a constant $m \times m$ matrix;

$$U_t + AU_x = 0 , \tag{4.38a}$$

$$U(x,0) = U^0(x) . \tag{4.38b}$$

We note that the eigenvalues $\{\lambda^k\}$ are constant as are the right eigenvectors $\{r^k\}$ and the left eigenvectors $\{l^k\}$. We assume that these eigenvectors are suitably normalized so that

$$l^i \cdot r^j = \delta_{ij} . \tag{4.39a}$$

If we define the kth characteristic variable w^k by the dot product

$$w^k = l^k \cdot U , \qquad k = 1,2,\ldots,m , \tag{4.39b}$$

then it follows from (4.39a) that

$$U = \sum_{k=1}^{m} w^k r^k . \tag{4.39c}$$

Multiplying (4.38) from the left by l^k, we see that $w^k(x,t)$ satisfies the scalar IVP

$$(w^k)_t + \lambda^k (w^k)_x = 0 , \tag{4.40a}$$

$$w^k(x,0) = l^k \cdot U^0(x) \equiv w_0^k(x) , \tag{4.40b}$$

the solution to which is

$$w^k(x,t) = w_0^k(x - \lambda^k t) . \tag{4.40c}$$

Performing the procedure (4.40) for $k = 1,2,\ldots,m$, and using (4.39c) and (4.40c), we can express the solution $U(x,t)$ of the IVP (4.38) as

$$U(x,t) = \sum_{k=1}^{m} w_0^k(x - \lambda^k t) r^k . \tag{4.40d}$$

It is easy to show that solving the IVP (4.38) "directly" in terms of U can lead to collisions of discontinuities in different characteristic fields. (An illustrative

61

example is given in [26].) However, if we work with the characteristic variables $\{w^k\}$, the coupled system of equations (4.38a) becomes a set of uncoupled equations of the form (4.40a). Therefore, any discontinuity in $w^k(x,t)$ cannot interact with a discontinuity in $w^j(x,t)$, when $j \neq k$. Furthermore, for a fixed k, all discontinuities in $w^k(x,t)$ propagate with same speed λ^k, and therefore cannot collide. Thus, in this constant–coefficient case (4.38), it makes sense to use these characteristic variables in the reconstruction procedure, rather than U itself. To this end, define the cell averages of the characteristic variables by

$$\bar{w}^k = l^k \cdot \bar{U} , \qquad (4.41a)$$

and then use the scalar reconstruction of these variables $R(x; \bar{w}^k)$ and the relation (4.39c) to define the vector reconstruction procedure

$$\mathbf{R}(x; \bar{U}) = \sum_{k=1}^m R(x; \bar{w}^k) r^k . \qquad (4.41b)$$

We now wish to generalize the reconstruction procedure (4.41) to the nonlinear system case. In the nonlinear case of (4.38a), the matrix $\mathbf{A}(U)$ is now a function of U, as are the eigenvalues $\{\lambda^k(U)\}$ and the eigenvectors $\{r^k(U)\}$, $\{l^k(U)\}$ associated with $\mathbf{A}(U)$. Our extension will require the use of locally defined characteristic variables, in the following manner. In order to reconstruct $U(x,t)$ in the interval $(x_{i-1/2}, x_{i+1/2})$, we first derive a set of local characteristic variables $\{\bar{w}_j^k(\bar{U}_i)\}$ by computing dot products of $l^k(\bar{U}_i)$ with the cell averages $\{\bar{U}_j\}$ associated with intervals in the immediate vicinity of $(x_{i-1/2}, x_{i+1/2})$, i.e.

$$\bar{w}_j^k(\bar{U}_i) = l^k(\bar{U}_i) \cdot \bar{U}_j , \qquad \text{for} \quad j = i - q, \ldots, i + q , \qquad (4.42a)$$

where q is the degree of the reconstruction polynomial. We then apply the scalar reconstruction operator R to this set $\{\bar{w}_j^k(\bar{U}_i)\}$ of $2q+1$ variables in which the left eigenvector has been locally "frozen" at the ith cell location. Our vector reconstruction procedure then becomes

$$\mathbf{R}(x;\bar{U}) = \sum_{k=1}^{m} R(x;\bar{w}^k(\bar{U}_i)) r^k(\bar{U}_i) . \tag{4.42b}$$

We turn now to our time discretization procedure. Having applied the vector reconstruction procedure (4.42) to the discrete set $\{\bar{V}_i^n\}$ of approximations of the cell averages $\{\bar{U}_i^n\}$ of the solution of (4.37) at time t^n, the abstract form of our scheme then requires that we solve exactly

$$V_t + F(V)_x = 0 , \tag{4.43a}$$

$$V(x,t^n) = \mathbf{R}(x;\bar{V}^n) , \tag{4.43b}$$

where (4.43a) represents a system of m equations.

The extension of our previous method–of–lines approach to systems of conservation laws is straight–forward. Analogous to the scalar case, we rewrite the system of partial differential equations (4.37a) in its semi–discrete formulation

$$\frac{\partial}{\partial t} \bar{V}_i(t) = - \frac{1}{\Delta x_i} \left[F(V(x_{i+1/2},t)) - F(V(x_{i-1/2},t)) \right] , \tag{4.44}$$

and then treat (4.44) as a system of ordinary differential equations for the purpose of time discretization. This is accomplished by applying the Runge–Kutta scheme (4.35) to each of the m equations in (4.44). In vector form, we write our scheme

$$\bar{V}_i^{(k)} = \sum_{l=0}^{k-1} [\alpha_{kl} \bar{V}_i^{(l)} + \beta_{kl} \Delta t (\mathbf{L} \bar{V}^{(l)})_i] , \qquad k = 1,2,\ldots,p , \tag{4.45a}$$

$$\bar{V}_i^{(0)} = \bar{V}_i^n , \qquad \bar{V}_i^{(p)} = \bar{V}_i^{n+1} , \tag{4.45b}$$

where

$$(\mathbf{L} \bar{V}^{(k)})_i = - \frac{1}{\Delta x_i} \left[\bar{F}_{i+1/2}^{(k)} - \bar{F}_{i-1/2}^{(k)} \right] , \tag{4.45c}$$

and

$$\bar{F}_{i+1/2}^{(k)} = F^R[\mathbf{R}(x_{i+1/2}-0;\bar{V}^{(k)}), \mathbf{R}(x_{i+1/2}+0;\bar{V}^{(k)})] , \tag{4.45d}$$

63

where $F^R[V_L, V_R]$ denotes the vector version of the "Riemann flux" defined in the previous section. Thus, the Runge–Kutta method (4.45a) is simply an application of the scalar method (4.35a) to each of the m components of the $\{V_i^n\}$. This simple extension is accomplished due to the previously mentioned "separation" of spatial and temporal operations.

We note, however, that the "Riemann flux" (4.45d) is *not* simply a component-wise extension of (4.35d). In fact, it is the solution at $x = x_{i+1/2}$ of the Riemann problem determining (4.45d) which ultimately takes into account the coupling of the system (4.44). We recall from Section 2.4 that, in the case of a hyperbolic system, the solution of the Riemann problem

$$V_t + F(V)_x = 0, \qquad V(x,0) = \begin{cases} V_L, & x < 0 \\ V_R, & x > 0, \end{cases} \qquad (4.46)$$

consists of $m+1$ constant states separated by m "waves." (The term "wave" is used here in the same loose sense as in the previous section.) As in the scalar case, the Riemann problem for systems is well understood. However, in light of the fact that a local solution is required only at a cell interface $x_{i+1/2}$, we see that much of the fine detail of this solution, which can be costly to compute, is ignored in the ultimate numerical flux computation (4.45d). Therefore, it makes sense to approximate the local solution of (4.46) through the use of an approximate "solver." We see that the "locally–linearized" method of vector reconstruction (4.42) presented in this section is ideally suited to the Roe's approximate Riemann solver derived in Section 3.4. Our local approximation is achieved by a local linearization with respect to a particular average

$$\hat{V} = \hat{V}(V_L, V_R), \qquad (4.47a)$$

i.e. we replace (4.46) by

$$V_t + \hat{A} V_x = 0, \qquad V(x,0) = \begin{cases} V_L, & x < 0 \\ V_R, & x > 0, \end{cases} \qquad (4.47b)$$

64

where $\hat{A} = A(\hat{V})$, the Jacobian matrix associated with (4.46) evaluated at $\hat{V}$. Following the design of Roe [45], we take this matrix $\hat{A}(V_L, V_R)$ to satisfy (3.26). As shown in Section 3.4, the flux at $x=0$ of the solution of (4.47b) can be expressed

$$F^{ROE}(V_L, V_R) = \frac{1}{2}\left[F(V_L) + F(V_R) \right] - \frac{1}{2}\sum_{k=1}^{m} \hat{\delta}^k |\hat{\lambda}^k| \hat{r}^k, \qquad (4.48a)$$

where

$$\hat{\delta}^k = \hat{l}^k \cdot (V_R - V_L), \qquad (4.48b)$$

and the eigenvalues $\{\hat{\lambda}^k\}$ and eigenvectors $\{\hat{l}^k\}$, $\{\hat{r}^k\}$ are evaluated with respect to $\hat{A}$. We recall from its derivation in Section 3.4 that the use of (4.48) can make it possible for the numerical scheme (4.45) to perfectly resolve a stationary discontinuity. However, because it does not allow for the finite spread of a rarefaction fan, F^{ROE} can admit a stationary "rarefaction shock" as a weak solution. Therefore, strictly speaking, (4.48) requires a modification in order to ensure that computed solutions are indeed entropy solutions. Most of these modifications , or "entropy fixes," to be found in the literature are based on the scalar problem. For some examples and their derivations, the interested reader is referred to [16,17,22].

Now that we have developed the high–order accurate ENO schemes for one–dimensional nonlinear systems, a numerical application would be in order. We take this route because ENO schemes are highly nonlinear and consequently do not easily lend themselves to rigorous analysis. Under such circumstances, computer experiments have become our main tool of "analysis." With these experiments we hope to address some of the open questions which as yet are not fully answered by purely mathematical methods. These topics include the accumulation of error, the adequacy of the "solution–in–the–small" procedure, consistency with entropy inequalities, and the characteristic reconstruction method (4.42). In the following section, we perform two numerical experiments with solutions of the one–dimensional Euler equations of gas dynamics, in order to make at least some qualitative judgement as

to the effectiveness of the scheme (4.45).

4.6 Euler Equations of Gas Dynamics

In this section, we describe the application of the scheme (4.45) to the Euler equations of gas dynamics, which we write in conservation form

$$U_t + F(U)_x = 0, \tag{4.49a}$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho E \end{bmatrix}, \qquad F(U) = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ (\rho E + P)u \end{bmatrix}. \tag{4.49b}$$

Here ρ, u, P, and E are the density, velocity, pressure, and total specific energy, respectively. We close the system (4.49a–b) of three equations with the polytropic equation of state :

$$P = (\gamma - 1)\rho(E - \tfrac{1}{2}u^2), \tag{4.49c}$$

where γ is the ratio of specific heats.

The eigenvalues of the Jacobian matrix $A(U) = \partial F/\partial U$ are

$$\lambda^1(U) = u - c, \quad \lambda^2(U) = u, \quad \lambda^3(U) = u + c, \tag{4.50a}$$

where c is the local speed of sound and, in this case, is given by

$$c = \sqrt{\frac{\gamma P}{\rho}}. \tag{4.50b}$$

The corresponding right eigenvectors associated with A are

$$r^1(U) = \begin{bmatrix} 1 \\ u - c \\ H - uc \end{bmatrix}, \quad r^2(U) = \begin{bmatrix} 1 \\ u \\ \tfrac{1}{2}u^2 \end{bmatrix}, \quad r^3(U) = \begin{bmatrix} 1 \\ u + c \\ H + uc \end{bmatrix}, \tag{4.50c}$$

where H is the total enthalpy and satisfies

$$H = \frac{E + P}{\rho} = \frac{c^2}{\gamma - 1} + \frac{u^2}{2}. \tag{4.50d}$$

66

The left eigenvectors $\{l^k(U)\}$, corresponding to the $\{r^k(U)\}$ in (4.50b), are

$$l^1(U) = \frac{1}{2}\left[\frac{(\gamma-1)u^2}{2c^2} + \frac{u}{c}, \quad \frac{(1-\gamma)u}{c^2} - \frac{1}{c}, \quad \frac{(\gamma-1)}{c^2}\right]$$

$$l^2(U) = \left[1 - \frac{(\gamma-1)u^2}{2c^2}, \quad \frac{(\gamma-1)u}{c^2}, \quad \frac{(1-\gamma)}{c^2}\right] \qquad (4.50e)$$

$$l^3(U) = \frac{1}{2}\left[\frac{(\gamma-1)u^2}{2c^2} - \frac{u}{c}, \quad \frac{(1-\gamma)u}{c^2} + \frac{1}{c}, \quad \frac{(\gamma-1)}{c^2}\right]$$

and satisfy the orthonormal relationship

$$l^j(U) \cdot r^k(U) = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}.$$

We now have all the information necessary to apply the scheme (4.45). Given a set $\{V_i^n\}$ of approximate cell averages of the solution to (4.49) at time t^n, we use (4.50e) to evaluate the local subsets of characteristic variables, to which we apply the vector reconstruction procedure (4.42). Then, in order to apply the kth stage of the Runge–Kutta method (4.45a), our scheme ultimately boils down to the calculation of the numerical flux

$$\bar{F}_{i+1/2}(\bar{V}^{(k)}) = F^R[\, \mathbf{R}(x_{i+1/2} - 0; \bar{V}^{(k)}), \, \mathbf{R}(x_{i+1/2} + 0; \bar{V}^{(k)}) \,], \qquad (4.51)$$

the flux across $x = x_{i+1/2}$ associated with the solution of the local Riemann problem, whose initial data are the limiting values at an interface from the left and right of the piecewise polynomial $\mathbf{R}(x; \bar{V}^{(k)})$. This reconstruction and subsequent flux computation is performed for $k = 1, 2, \ldots, p$, depending upon the desired order of accuracy of the Runge–Kutta scheme (4.45a).

Now, the flux computation (4.51) requires that each local Riemann problem be solved exactly. The solution to the Riemann problem for the one–dimensional Euler equations consists of at most four constant states separated by three centered waves,

67

these waves being of the three types discussed in Section 2.3. If all three types of waves are present in the solution, then the inner wave must be a contact discontinuity (See *e.g.* [37].). This solution is well understood and can be computed by an iterative algorithm. We note, however, that this algorithm is rather complicated and we refer the interested reader to [3,13,55] for its details. We choose, instead, to approximate (4.51) with the "Roe flux" (4.48). In order to accomplish this approximation with initial states V_L, V_R for a local Riemann problem, we require a particular average $\hat{V}(V_L, V_R)$ for the Euler equations of gas dynamics (See [47,45].). Specifically, the quantities (4.50) must be evaluated with respect to this average. Therefore, the only "Roe–average" values we need are $\hat{u}$, $\hat{c}$, and $\hat{H}$, and are given by

$$\hat{u} = \frac{\sqrt{\rho_L}\, u_L + \sqrt{\rho_R}\, u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}}$$

$$\hat{H} = \frac{\sqrt{\rho_L}\, H_L + \sqrt{\rho_R}\, H_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \qquad (4.52)$$

$$\hat{c} = \sqrt{(\gamma - 1)\left(\hat{H} - \tfrac{1}{2}\hat{u}^2\right)}.$$

It is shown in [48] that (4.52) represents the only average quantities for the Euler equations that satisfy (3.26).

We now describe two numerical experiments, both of which are themselves Riemann problems. Solutions "in the large" of particular Riemann problems are of physical significance in that they model the flow of a gas in a *shock tube*, pictorially described in Figure 7. A shock tube experiment is initialized in a cylindrical tube in which a diaphragm, perpendicular to the cylinder's axis, separates the gas into two different thermodynamic states (Figure 7a). Neglecting viscous effects near the cylinder walls, the ensuing motion of the gas, upon removal of the diaphragm, is one–dimensional. Therefore, the self–similar solution of the Riemann IVP (Figure 7b) determined by the two initial states is a good model of this phenomenon. The
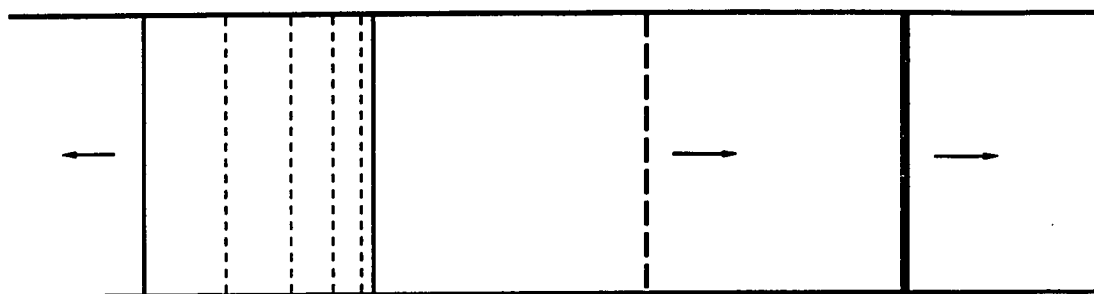
68

Figure 7c.   Ensuing wave motion for $t > 0$.
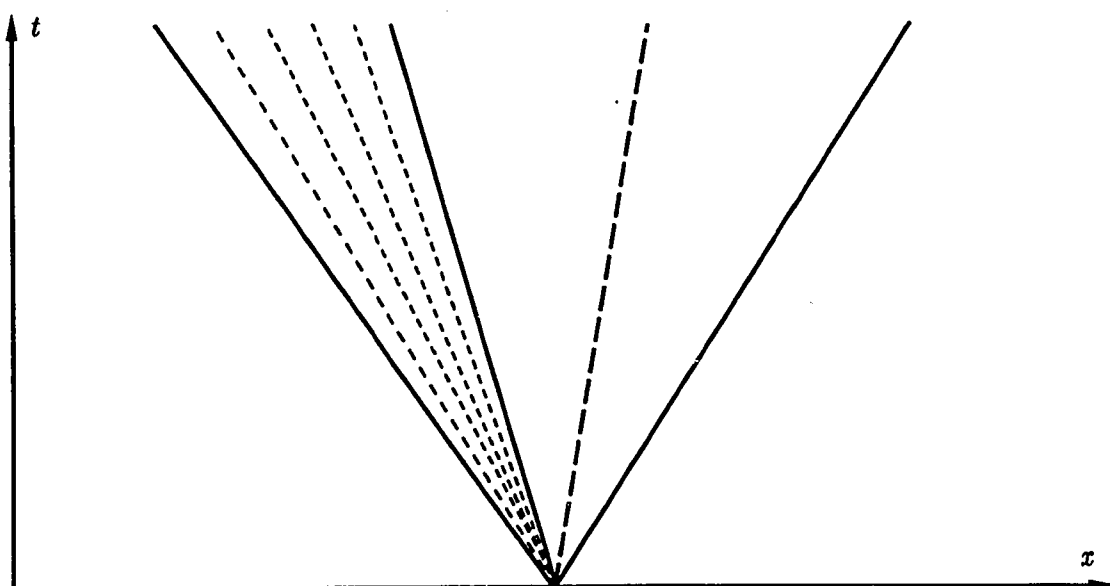

Figure 7b.   Solution of the Riemann IVP for the Euler equations
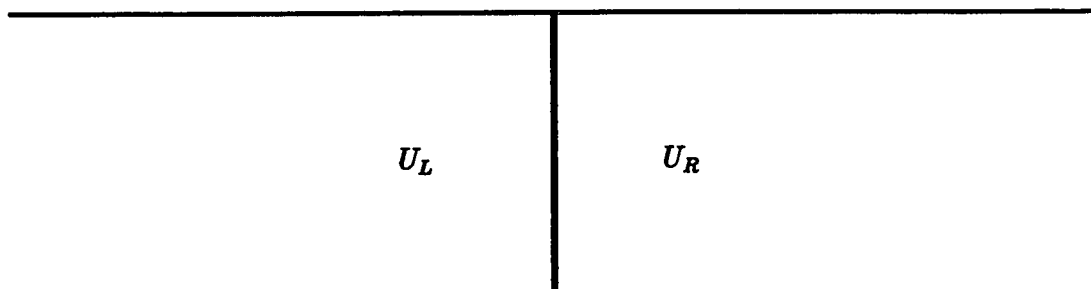

Figure 7a.   Initial conditions

69

physical orientation of both of our test problems will be as pictured in Figure 7c, with a shock moving to the right, followed by a contact discontinuity (dashed line), and a left–traveling expansion wave. The gas we model in both experiments is air, using the polytropic equation of state (4.49c) and $\gamma = 1.4$.

We will apply the scheme (4.45), modified by the "Roe flux" (4.48), using the averages (4.52). No entropy correction is used for (4.48) due to the fact that the sonic point within the expansion waves in either of the test cases is not stationary. The solution's spatial order of accuracy is achieved by the reconstruction (4.42), and for time accuracy we use the Runge–Kutta methods in (4.34).

Our first example, attributed to Sod [55], has become a standard test case in the literature. The initial conditions U_L, U_R can be stated in terms of the density, velocity, and pressure,

$$\begin{bmatrix} \rho_L \\ u_L \\ P_L \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} , \qquad \begin{bmatrix} \rho_R \\ u_R \\ P_R \end{bmatrix} = \begin{bmatrix} 0.125 \\ 0 \\ 0.10 \end{bmatrix} . \qquad (4.53)$$

We solve this IVP discretely on $0 \le x \le 1$, with 100 cells of uniform width. At $t = 0$, the "diaphragm" is positioned at $x = 1/2$, and the Euler equations (4.49) are solved subject to the initial data (4.53) for 50 time steps with a CFL number of 0.8 .

Test runs were performed for orders of accuracy $r = 1, 2, 3$ and 4, with satisfactory results. However, because viewing becomes impractical for four sets of data for each solution, we instead compare results for $r = 2$ and $r = 4$. In Figures 8a–f, these comparisons are presented with solution plots for density, pressure, and velocity. The numerical solution, depicted by circles, represents cell–centered output, while the continuous line represents the exact solution. Overall crisper resolution of discontinuities is observed in the fourth–order solution, with no visible sign of oscillatory behavior. In addition, "tighter" resolution of the smooth interior of the expansion fan is noticed near the wave's head and foot, where the first derivative is
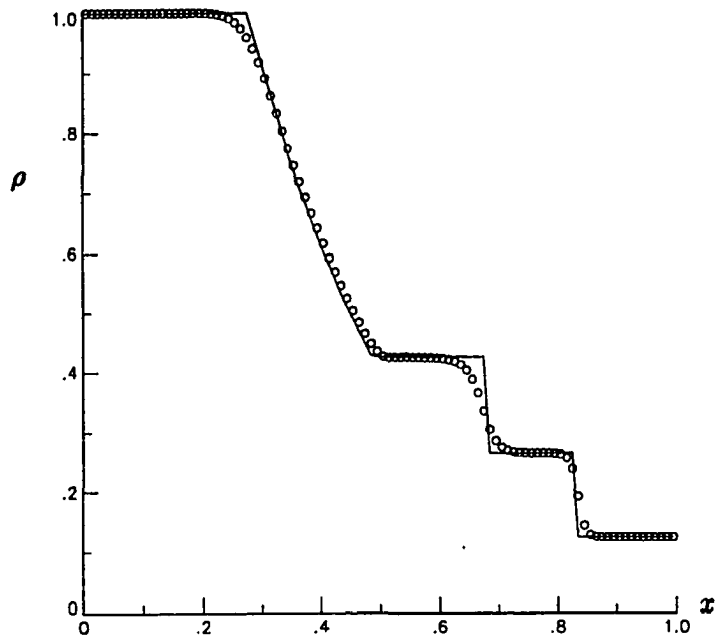
70

Sod's Problem

Figure 8a.　Density (2nd-Order ENO)

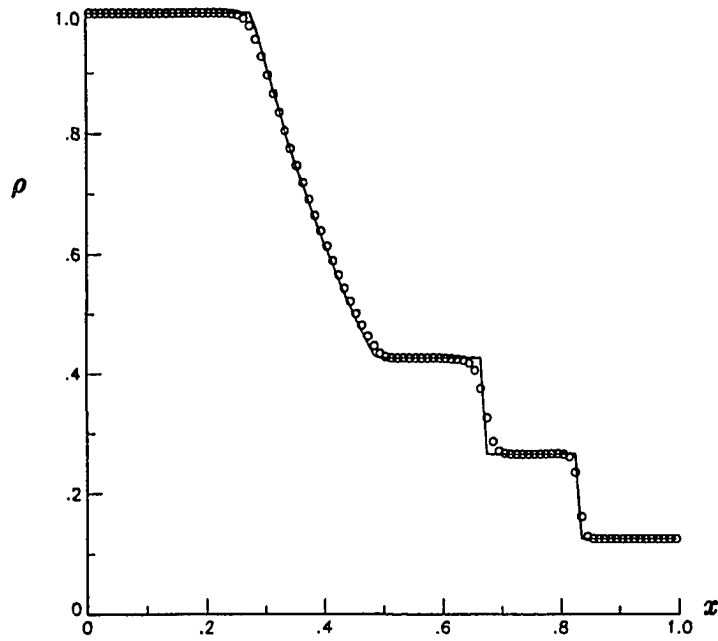Figure 8b.　Density (4th-Order ENO)

71
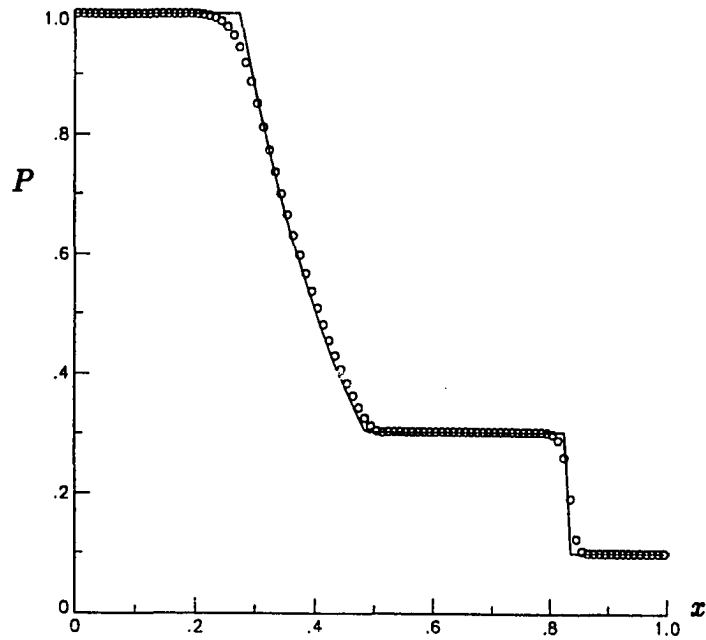
Sod's Problem

Figure 8c. Pressure (2nd–Order ENO)
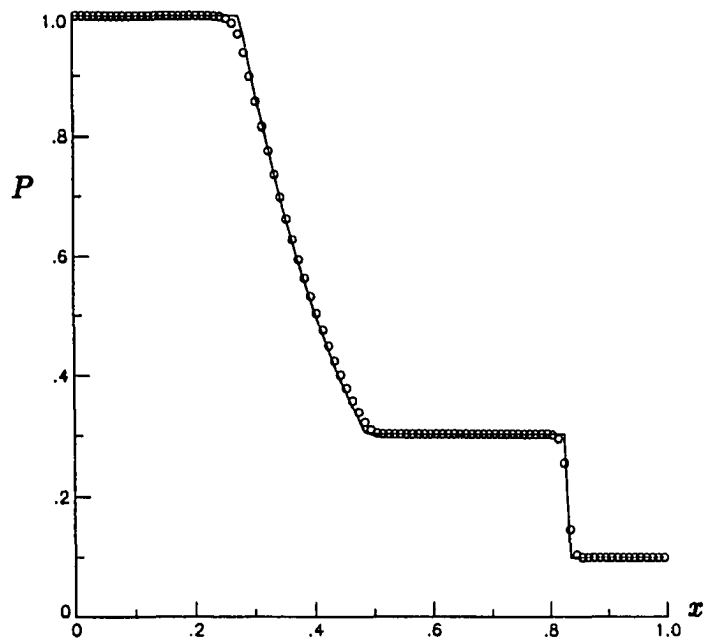
Figure 8d. Pressure (4th–Order ENO)

72

Sod's Problem
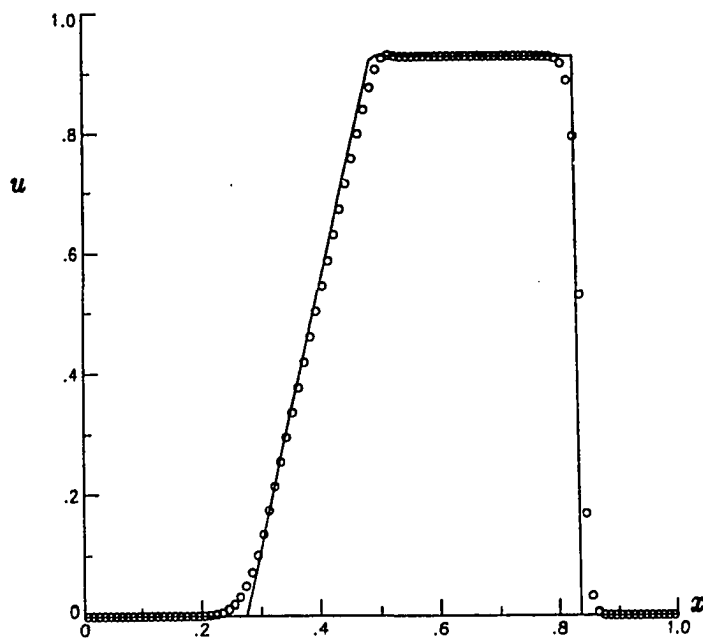
Figure 8e.   Velocity (2nd–Order ENO)
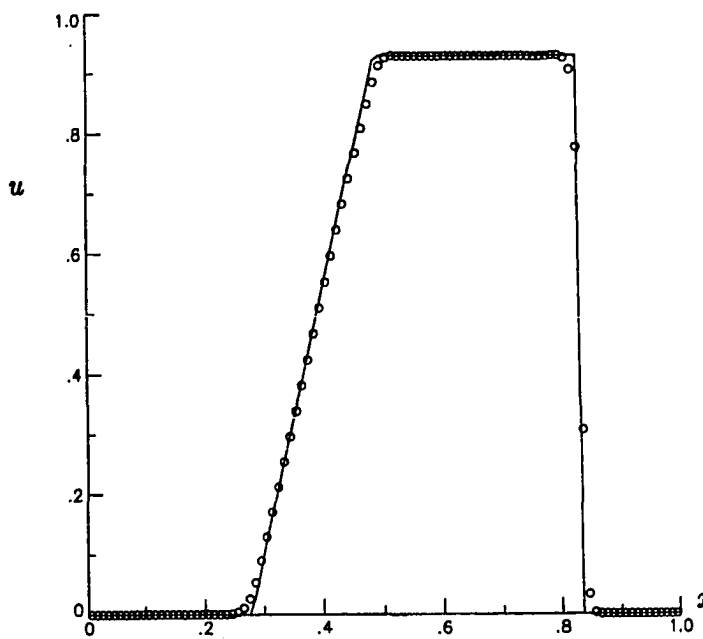
Figure 8f.   Velocity (4th–Order ENO)

73

discontinuous.

Our second Riemann problem, first presented by Lax [29], is given by the initial data

$$
\begin{bmatrix} \rho_L \\ u_L \\ P_L \end{bmatrix} = \begin{bmatrix} 0.445 \\ 0.698 \\ 3.528 \end{bmatrix} , \qquad \begin{bmatrix} \rho_R \\ u_R \\ P_R \end{bmatrix} = \begin{bmatrix} 0.5 \\ 0 \\ 0.571 \end{bmatrix} . \qquad (4.54)
$$

Once again, the spatial domain is $0 \le x \le 1$, with 100 cells of uniform width, and the "diaphragm" initially positioned at $x = 1/2$. We apply our numerical scheme for 75 time steps with a CFL number of 0.8. Second- and fourth-order results are displayed in Figures 9a–f. Because of the initial velocity in this problem, this case differs from Sod's problem, in that the density profile is no longer monotonic, and therefore has an intermediate state which has to be "built up," causing a very steep contact discontinuity to form. Once again, there is a noticeable difference between the second- and fourth-order cases, analogous to the previous example.

74
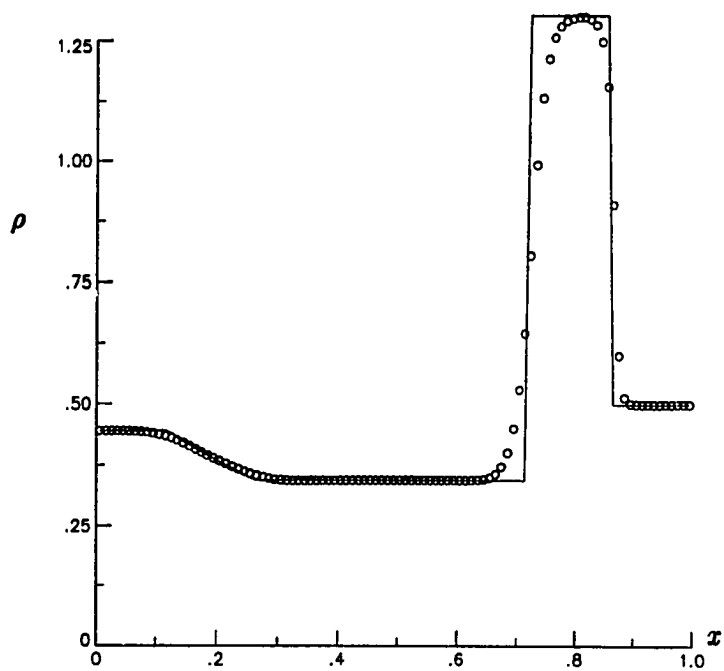
Lax's Problem

Figure 9a. Density (2nd-Order ENO)
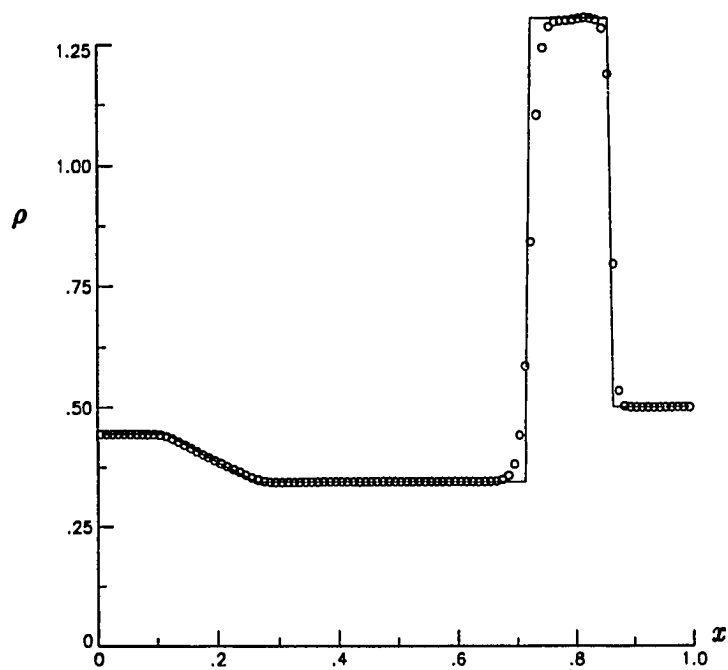
Figure 9b. Density (4th-Order ENO)

75
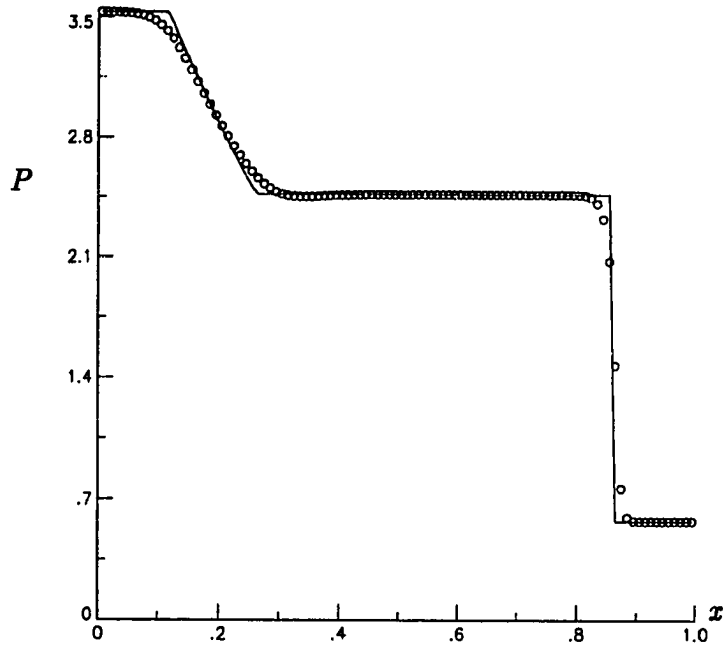
Lax's Problem

Figure 9c. **Pressure (2nd–Order ENO)**
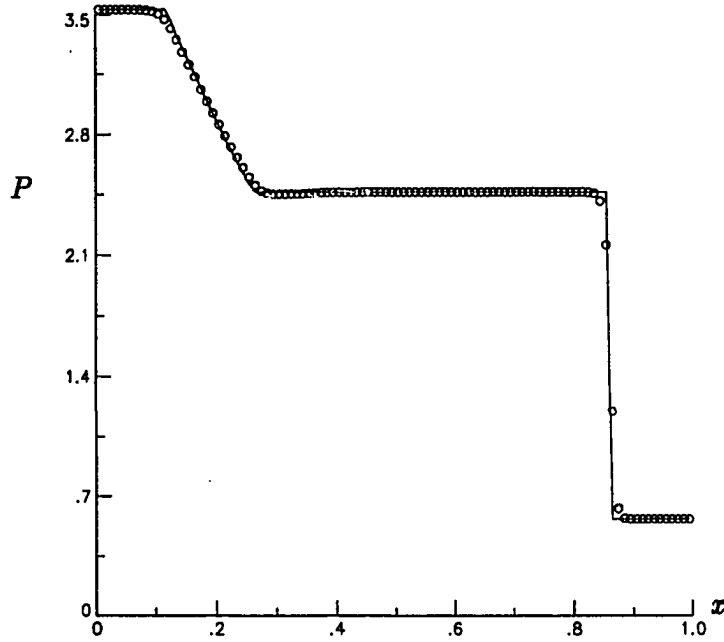
Figure 9d. **Pressure (4th–Order ENO)**

76

Lax's Problem
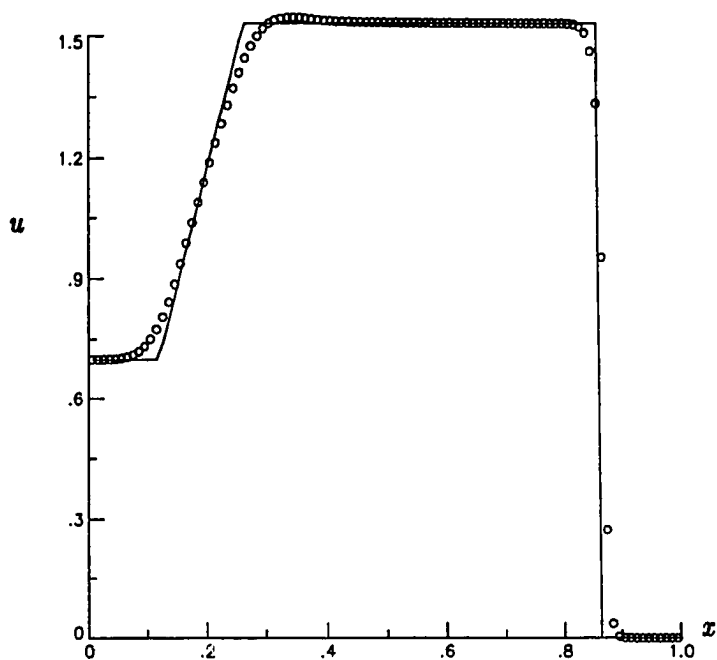
Figure 9e. Velocity (2nd-Order ENO)
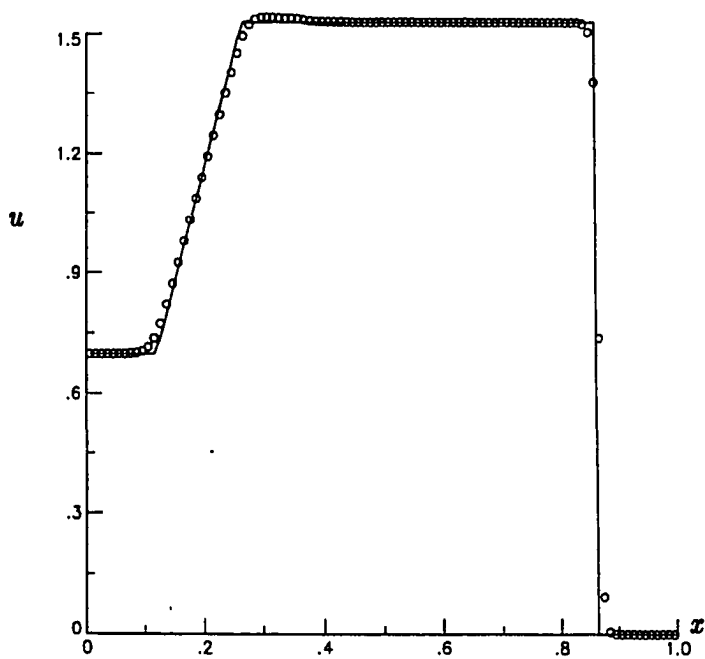
Figure 9f. Velocity (4th-Order ENO)

77

Chapter 5

Two–Dimensional ENO Schemes

Having thoroughly reviewed ENO schemes as applied to conservation laws in one spatial dimension, we are now prepared to discuss the central topic of this dissertation, namely the extension of these schemes to multiple dimensions. In particular, we will extend the ideas presented in Chapter 4 to two–dimensional conservation laws. Within the context of a finite–volume formulation, this extension represents the present author's contribution, for which purpose this dissertation is intended.

5.1   Formulation in Two Dimensions

We wish to extend the ideas presented in Chapter 4 to the design of high-order accurate ENO schemes for the numerical approximation of weak solutions of a two–dimensional system of conservation laws

$$u_t + f(u)_x + g(u)_y = 0 , \qquad (5.1a)$$

subject to given initial conditions

$$u(x, y, 0) = u^0(x, y) . \qquad (5.1b)$$

We will first formulate our extension in terms of the initial–value problem (5.1) and consider the boundary–value problem later.

The function $u = (u^1, u^2, \ldots, u^m)^T$ is a state vector and the fluxes $f(u)$ and $g(u)$ are vector-valued differentiable functions of m components. We assume that the

system (5.1a) is *hyperbolic* in the sense that the $m \times m$ Jacobian matrices

$$A(u) = \frac{\partial f}{\partial u}, \qquad B(u) = \frac{\partial g}{\partial u}$$

are such that any linear combination of A and B has m real eigenvalues $\{\lambda^k(u)\}$ and a complete set of m right eigenvectors $\{r^k(u)\}$ and left eigenvectors $\{l^k(u)\}$, which we assume to satisfy the orthonormal relation $l^i \cdot r^j = \delta_{ij}$.

We assume that the initial-value problem (5.1) is well-posed and that its solution is piecewise smooth, with at most a finite number of discontinuities. As in the one-dimensional case, we seek weak solutions of (5.1). Again appealing to the theory of weak solutions (Chapter 2), we require that u obey the integral form of (5.1a), where the limits of integration can reflect any smoothly bounded domain in the $x-y$ plane and any time interval (t^1, t^2). To this end, let

$$x_{i-1/2} < x < x_{i+1/2}, \qquad y_{j-1/2} < y < y_{j+1/2}, \qquad -\infty < i, j < \infty,$$

denote a rectangular partition of the $x-y$ plane, with (x_i, y_j) denoting the centroid of each rectangle. With a semi–discrete formulation in mind, we note that, for every rectangle $(x_{i-1/2}, x_{i+1/2}) \times (y_{j-1/2}, y_{j+1/2})$, a weak solution of (5.1a) must satisfy

$$\frac{\partial}{\partial t} \bar{u}_{ij}(t) = -\frac{1}{a_{ij}} \left[\hat{f}_{i+1/2,j}(t) - \hat{f}_{i-1/2,j}(t) + \hat{g}_{i,j+1/2}(t) - \hat{g}_{i,j-1/2}(t) \right], \quad (5.2a)$$

where a_{ij} is the area of the rectangle and

$$\bar{u}_{ij}(t) = \frac{1}{a_{ij}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} u(x, y, t) \, dx \, dy \qquad (5.2b)$$

is the cell average of u over the control volume at time t. The fluxes $\hat{f}$ and $\hat{g}$ are given by

$$\hat{f}_{i+1/2,j}(t) = \int_{y_{j-1/2}}^{y_{j+1/2}} f\big(u(x_{i+1/2}, y, t)\big) \, dy \; ; \qquad (5.2c)$$

$$\hat{g}_{i,j+1/2}(t) = \int_{x_{i-1/2}}^{x_{i+1/2}} g\big(u(x, y_{j+1/2}, t)\big) \, dx \; . \qquad (5.2d)$$

79

We now treat (5.2a) as a system of ordinary differential equations for the purpose of time discretization, using a "method–of–lines" approach. Along any $t = $ constant line, the right–hand side of (5.2a) is strictly a spatial operation in the unknown u, and we rewrite this equation, for fixed t, in the abstract operator–product form

$$\frac{\partial}{\partial t}\bar{\bar{u}}_{ij}(t) = (\mathcal{L}\, u(t))_{ij}, \tag{5.3}$$

thus effectively "separating" the spatial and temporal operations for computing solutions of (5.1).

Clearly, our two–dimensional extension of a finite–volume, semi–discrete formulation has been straight forward. So, too, is the temporal discretization of (5.3), as we will employ the high–order Runge–Kutta methods discussed in Section 4.4 in order to achieve our desired accuracy in time. However, the high–order discretization of the right–hand side of (5.3) requires some special consideration when using a finite–volume approach, as will be explained in the following two sections. The use of the formulation (5.2a), rather than the finite–difference approach of [50,52], makes for a more immediate application of its numerical approximation to the solutions of boundary–value problems involving solid walls or non–trivial geometries.

5.2 Two–Dimensional Design

We now proceed to abstractly design a two–dimensional, finite–volume ENO scheme which retains the basic properties of the one–dimensional scheme in Chapter 4. Given $\{\bar{\bar{u}}^n\}$, cell averages of a piecewise smooth solution $u(x, y, t^n)$ of (5.1), we desire a high–order accurate numerical solution operator E_h which will update these averages to time $t^{n+1} = t^n + \Delta t$. Specifically, we require that E_h be r^{th}–order accurate in the sense of local truncation error, *i.e.*

$$E_h\, \bar{\bar{u}}^n - \bar{\bar{u}}^{n+1} = O(h^{r+1}) \tag{5.4}$$

80

wherever u is sufficiently smooth, with $\Delta x, \Delta y$, and Δt all assumed to be $O(h)$. Furthermore, we desire that our numerical update scheme avoid the development of spurious $O(1)$ oscillations near discontinuities in u. In order to achieve this property we require our operator to be essentially non–oscillatory, whose definition is relegated to the one–dimensional scalar case,

$$TV(E_h\,\bar{u}) = TV(u) + O(h^r)\,,\tag{5.5}$$

where TV represents total variation in x.

Employing the formulation (5.3) for the numerical update of the $\{\bar{u}^n\}$, we discretize the temporal operation by using a Runge–Kutta method of the type described in Section 4.4, the extension to two dimensions being straight forward :

$$\bar{u}_{ij}^{(l)} = \sum_{m=0}^{l-1} [\,\alpha_{lm}\,\bar{u}_{ij}^{(m)} + \beta_{lm}\,\Delta t\,(\,\mathcal{L}\,u^{(m)}\,)_{ij}\,]\,,\qquad l = 1, 2, \ldots, p\,,\tag{5.6}$$

$$\bar{u}_{ij}^{(0)} = \bar{u}_{ij}^n\,,\qquad \bar{u}_{ij}^{(p)} = \bar{u}_{ij}^{n+1}\,.$$

As in the one–dimensional case, the order of accuracy achieved by this time discretization, as well as its TVD property, is determined by the values of the integer p and the coefficients α and β, and are given by (4.34) for $r = 1, 2, 3$, and 4.

Now, if we assume that the scheme (5.6) achieves our desired r^{th}–order accuracy in time, then clearly, this scheme satisfies (5.4) if we can evaluate $(\,\mathcal{L}\,u(t)\,)_{ij}$, the exact spatial operation on the right–hand side of (5.2a). However, the calculation of the fluxes (5.2c–d) needed for this evaluation requires that we know the solution $u(x, y, t)$, pointwise, at a given time t, whereas the information we have at any time t is that of the cell averages (5.2b). Analogous to the one–dimensional case, since

$$\bar{u}_{ij}(t) = u(x_i, y_j, t) + O(h^2)\,,$$

wherever u is smooth, there is an inherent limit on the order of accuracy if we use the cell averages themselves in the flux calculation. Therefore, we replace the operator

81

$\mathcal{L}$ with a discrete spatial operator L which acts upon the $\{\bar{u}(t)\}$ and approximates the pointwise operation of $\mathcal{L}$ to high order. To this end, we see that if

$$L\,\bar{u}(t) \;=\; \mathcal{L}\,u(t) \;+\; O(h^r)\,, \tag{5.7a}$$

wherever u is smooth, then, upon replacement of $\mathcal{L}$ in (5.6), the local truncation error of our fully discrete scheme will satisfy (5.4), as required.

We define L explicitly by

$$(L\,\bar{u}(t))_{ij} \;=\; -\,\frac{1}{c_{ij}}\left[\, \bar{f}_{i+1/2,j}(t) \,-\, \bar{f}_{i-1/2,j}(t) \,+\, \bar{g}_{i,j+1/2}(t) \,-\, \bar{g}_{i,j-1/2}(t)\,\right]\,, \tag{5.7b}$$

where $\bar{f}$ and $\bar{g}$ are to be designed as high-order numerical approximations to $\hat{f}$ and $\hat{g}$ in (5.2c–d). We require that the numerical flux functions $\bar{f}$, $\bar{g}$

$$\bar{f}_{i+1/2,j}(t) \;=\; \bar{f}(\bar{u}_{i-k+1,j-m+1}(t),\dots,\bar{u}_{i+k,j+m}(t))\,, \quad 1\le k,m\le r\,, \tag{5.7c}$$

$$\bar{g}_{i,j+1/2}(t) \;=\; \bar{g}(\bar{u}_{i-l+1,j-n+1}(t),\dots,\bar{u}_{i+l,j+n}(t))\,, \quad 1\le l,n\le r\,,$$

be Lipschitz continuous functions of their arguments and consistent with the true fluxes $f(u)$, $g(u)$ in the sense

$$\bar{f}(w,\dots,w) \;=\; f(w)\,, \qquad \bar{g}(w,\dots,w) \;=\; g(w)\,. \tag{5.7d}$$

As in the previous one–dimensional design, the first and most important step in the high–order approximations of $\hat{f}$, $\hat{g}$ is the method by which we obtain high–order accurate pointwise information of the solution $u(x,y,t)$ from the given set of cell averages $\{\bar{u}(t)\}$. For this purpose, let R^2 be a spatial operator which reconstructs this set of cell averages and yields a two–dimensional, piecewise polynomial $R^2(x,y\,;\bar{u}(t))$ of degree $r-1$ and thereby approximates $u(x,y,t)$ with a truncation error of $O(h^r)$, wherever u is sufficiently smooth. We write this approximate relationship in the form

$$R^2(x,y\,;\bar{u}(t)) \;=\; u(x,y,t) \;+\; e(x,y)\,h^r\,. \tag{5.8}$$

82

In general, the reconstruction operator R^2 which we have developed is a "natural" two–dimensional extension of the one–dimensional "reconstruction by primitive" presented in Section 4.3 . This operator is detailed in the following section.

In order to include the more general case, where $f(u)$ cannot be integrated in closed form in (5.2c), we will approximate this integration by Gaussian quadrature. In order to express the error made by this approximation, let $q(x)$ be a $C^{(2K)}$ function whose integration on $[a, b]$ we approximate by the "classical" Gaussian quadrature, *i.e.* relative to the unit weight function on the interval $[-1, 1]$. It can be shown (*e.g.* [54]) that the error made by this approximation with a K–point quadrature is given by

$$\int_a^b q(x)\, dx \;-\; \frac{b-a}{2} \sum_{k=1}^K c_k\, q(x_k) \;=\; \frac{q^{(2K)}(\xi)}{(2K)!} \int_a^b P_K^2(x)\, dx \; ,$$

for some ξ in (a, b), with P_K being the polynomial of degree K in the orthogonal basis that spans the space of polynomials of degree not exceeding K. This quadrature is exact when $q(x)$ is a polynomial of degree less than or equal to $2K - 1$. The roots of $P_K(x) = (x - x_1)(x - x_2) \ldots (x - x_K)$ are real and distinct, making it clear that the above truncation error is $O(h^{2K+1})$. Relating this error to the $(r-1)$–st degree polynomial reconstruction (5.8), we see that for $r \leq 2K$, this truncation error is at worst $O(h^{r+1})$ when $r-1$ is odd, and $O(h^{r+2})$ when $r-1$ is even. Therefore, using the "larger" error, for fixed x and t, and sufficiently smooth f, the approximation of the flux integral (5.2c) by Gaussian quadrature satisfies

$$\hat{f}_{i+1/2, j}(t) \;=\; \int_{y_{j-1/2}}^{y_{j+1/2}} f\big(u(x_{i+1/2}, y, t) \big)\, dy$$

$$=\; \frac{\Delta y_j}{2} \sum_{k=1}^K c_k\, f\big(u(x_{i+1/2}, y_k, t) \big) \;+\; s(x_{i+1/2}, \eta)\, h^{r+1} \; , \quad (5.9a)$$

for some η in $(y_{j-1/2}, y_{j+1/2})$, and $\Delta y_j = y_{j+1/2} - y_{j-1/2}$.

Let $v_h(x, y, t)$ denote the piecewise polynomial approximation to u which is de-

83

termined by the reconstruction operator R^2 in (5.8) and therefore satisfies

$$v_h(x,y,t) = u(x,y,t) + e(x,y)h^r , \tag{5.9b}$$

for fixed t, wherever u is sufficiently smooth. Since f is assumed differentiable in u, it is therefore Lipschitz continuous in u, and thus, for fixed t,

$$f(u(x,y,t)) = f(v_h(x,y,t)) + d(x,y)h^r , \tag{5.9c}$$

where $d(x,y) = O(e(x,y))$. Finally, we substitute (5.9c) into the quadrature in (5.9a), and we see that

$$\hat{f}_{i+1/2,j}(t) = \frac{\Delta y_j}{2} \sum_{k=1}^{K} c_k \left[f(v_h(x_{i+1/2}, y_k, t)) + d(x_{i+1/2}, y_k)h^r \right] \tag{5.9d}$$

$$+ s(x_{i+1/2}, \eta) h^{r+1} .$$

Therefore, if we define our "abstract" numerical flux $\bar{f}_{i+1/2,j}(t)$ in (5.7b) by

$$\bar{f}_{i+1/2,j}(t) = \frac{\Delta y_j}{2} \sum_{k=1}^{K} c_k f(v_h(x_{i+1/2}, y_k, t)) , \tag{5.10a}$$

then the error made by the approximate flux difference $\bar{f}_{i+1/2,j}(t) - \bar{f}_{i-1/2,j}(t)$ in the definition (5.7b) is given by

$$\hat{f}_{i+1/2,j}(t) - \hat{f}_{i-1/2,j}(t) = \bar{f}_{i+1/2,j}(t) - \bar{f}_{i-1/2,j}(t)$$

$$+ \frac{\Delta y_j}{2} \sum_{k=1}^{K} c_k \left[d(x_{i+1/2}, y_k) - d(x_{i-1/2}, y_k) \right] h^r$$

$$+ \left[s(x_{i+1/2}, \eta) - s(x_{i-1/2}, \eta) \right] h^{r+1} . \tag{5.10b}$$

Clearly, if d and s are Lipschitz continuous on $[x_{i-1/2}, x_{i+1/2}]$ for each y, then the error relation in (5.10b) satisfies

$$\hat{f}_{i+1/2,j}(t) - \hat{f}_{i-1/2,j}(t) = \bar{f}_{i+1/2,j}(t) - \bar{f}_{i-1/2,j}(t) + O(h^{r+2}) . \tag{5.10c}$$

Moreover, a symmetrical argument can be used to show

$$\hat{g}_{i,j+1/2}(t) - \hat{g}_{i,j-1/2}(t) = \bar{g}_{i,j+1/2}(t) - \bar{g}_{i,j-1/2}(t) + O(h^{r+2}) , \tag{5.10d}$$

84

where

$$g_{i,j+1/2}(t) \;=\; \frac{\Delta x_i}{2} \sum_{k=1}^{K} c_k\, g\big(v_h(x_k,\, y_{j+1/2},\, t)\big)\,. \tag{5.10e}$$

Noting that the area a_{ij} is $O(h^2)$, we see that upon substitution of the numerical fluxes (5.10a) and (5.10e) which satisfy the error relations (5.10c) and (5.10d) into (5.7b), we have thus designed the spatial operator L that satisfies (5.7a), and therefore, when substituted for $\mathcal{L}$ in (5.6), yields a numerical solution operator E_h which is r^{th}- order accurate in the sense of local truncation error. We note here that the desired truncation errors given by (5.10c–d) are achieved only if the functions representing the errors due to the quadrature (5.9a) and the solution approximation (5.8) are Lipschitz continuous on $[x_{i-1/2}, x_{i+1/2}] \times [y_{j-1/2}, y_{j+1/2}]$.

We now wish to modify the numerical fluxes (5.10a) and (5.10e) such that (5.4) still holds in regions where the solution is smooth and, in addition, these fluxes will account for possible discontinuities in u. This modification is largely due to the nature of the reconstruction step, at which the pointwise behavior of $u(x,y,t)$ is approximated in a piecewise polynomial fashion within each cell (Section 5.3.). As a result, the polynomial generated by the reconstruction operator can be discontinuous at cell interfaces, analogous to the one–dimensional reconstruction in Section 4.3 . Therefore, in order to resolve these discontinuities, the flux integrands in (5.2c–d) are approximated by

$$f(u(x,y,t)) \;\approx\; f^{R}\big[\,R^2(x-0,y;\bar{u}(t)),\; R^2(x+0,y;\bar{u}(t))\,\big]\,, \tag{5.11a}$$

$$g(u(x,y,t)) \;\approx\; g^{R}\big[\,R^2(x,y-0;\bar{u}(t)),\; R^2(x,y+0;\bar{u}(t))\,\big]\,, \tag{5.11b}$$

where $f^{R}[u_1, u_2]$ denotes the flux across $x = 0$ associated with the solution to the Riemann problem whose initial states are u_1 and u_2. As before, the notation $q(x+0)$, $q(x-0)$ denotes the limiting values of q at x from the right and left, respectively. When the solution u is sufficiently smooth, the "jumps" in the approximate

85

solution at cell interfaces will be $O(h^r)$ in which case our previous conclusions concerning high–order truncation error are not altered by the modification (5.11). The manner in which we interpret the Riemann problem in this two–dimensional setting is discussed in Sections 5.6 and 5.7.

In fully–discrete form, we write our two–dimensional scheme as the Runge–Kutta method

$$\bar{u}_{ij}^{(l)} = \sum_{m=0}^{l-1} [\, \alpha_{lm}\, \bar{u}_{ij}^{(m)} + \beta_{lm}\, \Delta t\, (\, L\bar{u}^{(m)}\,)_{ij} \,]\,, \qquad l = 1, 2, \ldots, p\,, \quad (5.12a)$$

$$\bar{u}_{ij}^{(0)} = \bar{u}_{ij}^n\,, \qquad \bar{u}_{ij}^{(p)} = \bar{u}_{ij}^{n+1}\,,$$

where

$$(\, L\,\bar{u}^{(m)}\,)_{ij} = -\frac{1}{a_{ij}} \left[\, \bar{f}_{i+1/2,j}^{(m)} - \bar{f}_{i-1/2,j}^{(m)} + \bar{g}_{i,j+1/2}^{(m)} - \bar{g}_{i,j-1/2}^{(m)} \,\right]\,, \qquad (5.12b)$$

and

$$\bar{f}_{i+1/2,j}^{(m)} = \frac{\Delta y_j}{2} \sum_{k=1}^{K} c_k\, f^R[\, R^2(x_{i+1/2} - 0\,,\, y_k\,;\, \bar{u}^{(m)})\,,\, R^2(x_{i+1/2} + 0\,,\, y_k\,;\, \bar{u}^{(m)}) \,]\,,$$

$$(5.12c)$$

$$\bar{g}_{i,j+1/2}^{(m)} = \frac{\Delta x_i}{2} \sum_{k=1}^{K} c_k\, g^R[\, R^2(x_k\,,\, y_{j+1/2} - 0\,;\, \bar{u}^{(m)})\,,\, R^2(x_k\,,\, y_{j+1/2} + 0\,;\, \bar{u}^{(m)}) \,]\,.$$

$$(5.12d)$$

Assuming the error functions $d(x, y)$ and $s(x, y)$ in (5.10b) to be globally Lipschitz continuous, the numerical solution operator E_h defined by (5.12) is formally r^{th}–order accurate in the sense of local truncation error as given by (5.4). Furthermore, if these error functions remain Lipschitz continuous for N time steps, where $N = t/\Delta t = O(1/h)$, we assume the cumulative error to be $O(h^r)$. Thus, at the end of such a computation, we have a set $\{\bar{v}_{ij}^N\}$, approximations to the cell averages of u at time t^N which satisfy

$$\bar{v}^N - \bar{u}^N = O(h^r)\,. \qquad (5.13a)$$

If we desire our high-order accurate output in pointwise form, we simply perform one final reconstruction which, by (5.8) and (5.13a), will yield

$$R^2(x, y; \bar{v}^N) = u(x, y, t^N) + O(h^r) \,. \qquad (5.13b)$$

In addition to the accuracy properties (5.13), we desire that if u should develop discontinuities, then the scheme (5.12) will avoid $O(1)$ spurious oscillations, and we will design the reconstruction operator R^2 to do so.

5.3   Two–Dimensional Reconstruction

We turn now to describe more fully the high–order spatial reconstruction operator R^2 in (5.8) which is crucial to the scheme's accuracy. For the purpose of clarity, we discuss the finer points of this procedure within the framework a scalar function defined on a rectangular computational mesh.

At first glance, it might seem plausible to extend the reconstruction operator R (Chapter 4) to two dimensions by simply "overlapping" two one–dimensional stencils, as depicted in Figure 10. As will become apparent in this section, this "dimension–by–dimension" approach is inherently limited to second–order spatial accuracy. Nonetheless, if we restrict ourselves to the notion of a "structured" grid, we can describe the implementation of R^2 as a composition of two applications of a one–dimensional operator R, where the latter is the "reconstruction-by-primitive" operator in Chapter 4.

We are initially given a discrete set of cell averages $\{\bar{w}_{ij}\}$ of a piecewise smooth function $w(x, y)$,

$$\bar{w}_{ij} = \frac{1}{a_{ij}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} w(x, y) \, dy \, dx \,, \qquad (5.14a)$$

where $a_{ij} = \Delta x_i \Delta y_j = (x_{i+1/2} - x_{i-1/2})(y_{j+1/2} - y_{j-1/2})$. For $y_{j+1/2} < y < y_{j-1/2}$,
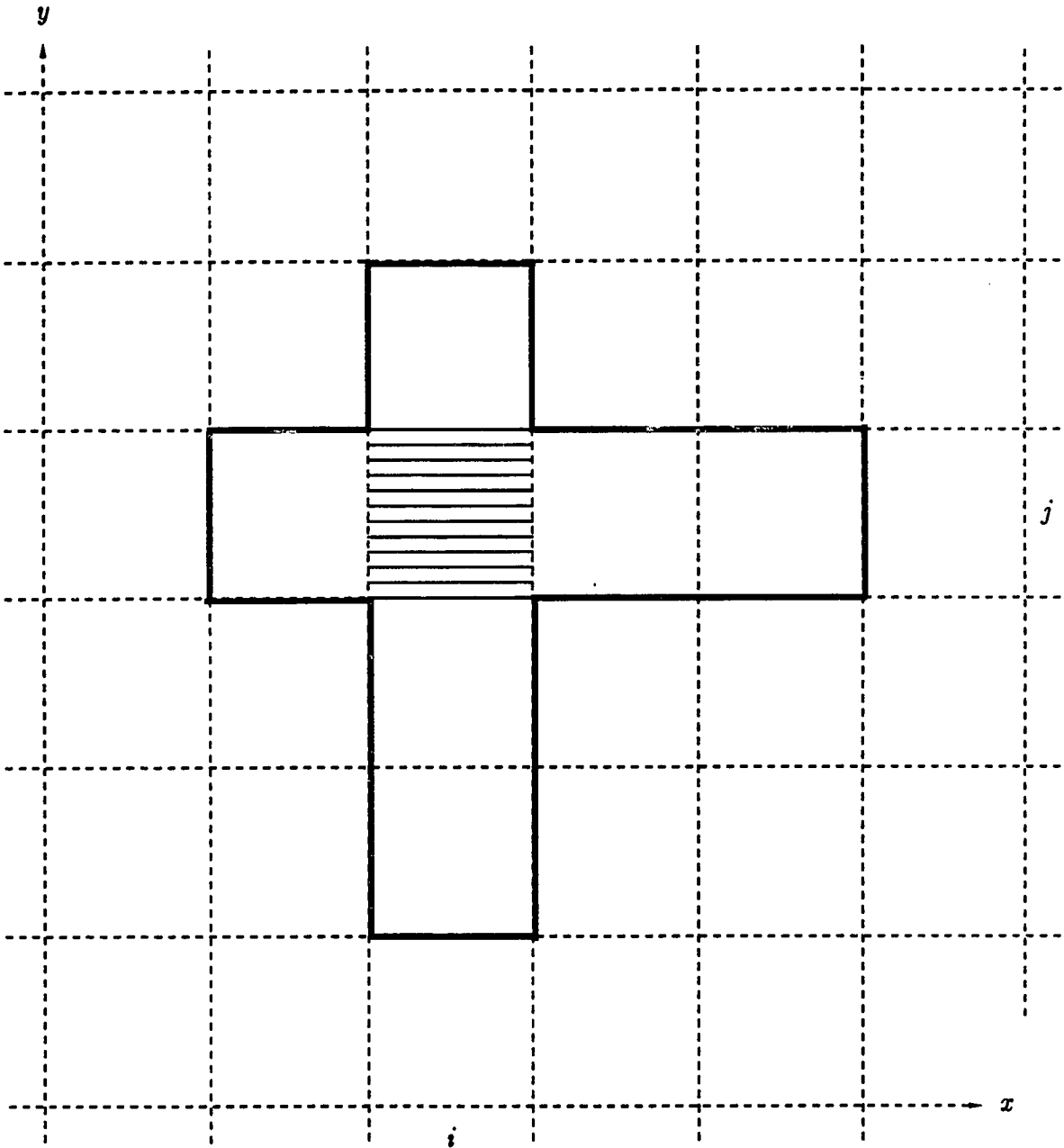
87

Figure 10.   A "dimension-by-dimension" stencil (bold outline) uses cell-average
information from the x-stencil for point values on the vertical faces of
the shaded cell (i,j), and the y-stencil accounts for those values on the
horizontal faces. This approach is inherently limited to second-order
accuracy.

88

define the *primitive function* $\bar{W}_j(x)$ associated with w by

$$\bar{W}_j(x) = \int_{x_0}^{x} \frac{1}{\Delta y_j} \int_{y_{j-1/2}}^{y_{j+1/2}} w(\xi, y) \, dy \, d\xi \ . \tag{5.14b}$$

Seeking a relationship between pointwise values of $\bar{W}_j(x)$ and the discrete values $\{\bar{\bar{w}}_{ij}\}$, we see immediately from the definitions (5.14a-b) that

$$\Delta x_i \, \bar{\bar{w}}_{ij} = \bar{W}_j(x_{i+1/2}) - \bar{W}_j(x_{i-1/2}) \ ,$$

and we can therefore establish such a relationship at the cell interfaces:

$$\bar{W}_j(x_{i+1/2}) = \sum_{k=i_0}^{i} \Delta x_k \, \bar{\bar{w}}_{kj} \ . \tag{5.14c}$$

Now, the definition (5.14b) clearly implies

$$\frac{d}{dx} \bar{W}_j(x) = \frac{1}{\Delta y_j} \int_{y_{j-1/2}}^{y_{j+1/2}} w(x, y) \, dy \equiv \bar{w}_j(x) \ , \tag{5.14d}$$

thus denoting by $\bar{w}_j(x)$ the line average in y of $w(x, y)$ for a fixed x. This suggests that if we approximate $\bar{W}_j(x)$ by $Q_r(x; \bar{W}_j)$, the ENO piecewise polynomial of degree r which interpolates $\bar{W}_j$ at the values given by (5.14c), we can then obtain an approximation of $\bar{w}_j(x)$ by defining the first step in our reconstruction procedure as

$$R(x; \bar{w}) = \frac{d}{dx} Q_r(x; \bar{W}_j) \equiv \bar{v}_j(x) \ . \tag{5.14e}$$

Then $\bar{v}_j(x)$ is a polynomial in x of degree $r-1$ which satisfies

$$\bar{v}_j(x) = \bar{w}_j(x) + O(\Delta x^r) \ , \tag{5.14f}$$

wherever $\bar{w}_j(x)$ is sufficiently smooth in x.

If the procedure (5.14) is performed for all j, then we have a set of piecewise polynomials $\{\bar{v}_j(x)\}$, each of which is a high-order approximation in x to each $\bar{w}_j(x)$. Clearly, from the definition (5.14d), the value of $\bar{w}_j(x)$ in x is equivalent in form to a one-dimensional cell average on the interval $[y_{j+1/2}, y_{j-1/2}]$. Therefore, for a fixed x, the remainder of our reconstruction procedure becomes equivalent to the one-dimensional method in Chapter 4, applied to the set $\{\bar{v}_j(x)\}$.

89

For fixed x, we now treat the set $\{\bar{v}_j(x)\}$ as one–dimensional cell averages in y of a piecewise smooth function $v(x,y)$, which we wish to reconstruct to high–order pointwise accuracy in y. Analogous to the method (5.14), we define another primitive function $W(x,y)$ associated with $v(x,y)$ by

$$W(x,y) \;=\; \int_{y_0}^{y} v(x,\eta)\,d\eta \; , \tag{5.15a}$$

whose pointwise values we know at cell interfaces

$$W(x, y_{j+1/2}) \;=\; \sum_{k=j_0}^{j} \Delta y_k\, \bar{v}_k(x) \; . \tag{5.15b}$$

Fitting the point values (5.15b) of $W(y)$ with a piecewise polynomial $Q_r(y\,;W)$ of degree r by ENO interpolation, we can obtain a high–order pointwise approximation to $v(x,y)$ in y by defining the second reconstruction step

$$R(y\,;\bar{v}(x)) \;=\; \frac{d}{dy}\,Q_r(y\,;W) \;\equiv\; p(x,y) \; , \tag{5.15c}$$

where, for fixed x, $p(x,y)$ is a polynomial in y of degree $r-1$ that satisfies

$$p(x,y) \;=\; v(x,y) \,+\, O(\Delta y^r) \; , \tag{5.15d}$$

wherever $v(x,y)$ is sufficiently smooth.

Noting the reconstruction definitions and the error relationships above, in addition to the standard polynomial approximation results referred to in Section 4.3, we can see that the values obtained in (5.15c) are the high–order pointwise approximations to $w(x,y)$ which we desired from the initial cell averages $\{\bar{\bar{w}}_{ij}\}$, i.e.

$$R^2(x,y\,;\,\bar{\bar{w}}) \;\equiv\; R(y\,;R(x\,;\bar{\bar{w}})) \;=\; w(x,y) \,+\, O(\Delta x^r, \Delta y^r) \; . \tag{5.16}$$

In addition to the high–order truncation error in regions of smoothness, we also note that each of the one–dimensional reconstruction operators (5.14e) and (5.15c) is essentially non–oscillatory, due to the nature of the interpolating polynomial Q_r as described in Chapter 4. Furthermore, we note that R^2 is "conservative", in the

90

sense that the cell–averaging operator A_h defined by the right–hand side of (5.14a) is its left–hand inverse, $i.e.$

$$(A_h \, R^2(x,y; \bar{w}))_{ij} \;=\; \frac{1}{a_{ij}} \int_{x_{i-1/2}}^{x_{i+1/2}} \int_{y_{j-1/2}}^{y_{j+1/2}} R(\, y; \, R(x; \bar{w})\,) \, dy \, dx$$

$$=\; \frac{1}{a_{ij}} \int_{x_{i-1/2}}^{x_{i+1/2}} \left[\; Q_r(y_{j+1/2};W) - Q_r(y_{j-1/2};W) \; \right] \, dx$$

$$=\; \frac{1}{a_{ij}} \int_{x_{i-1/2}}^{x_{i+1/2}} \left[\; W(x,y_{j+1/2}) - W(x,y_{j-1/2}) \; \right] \, dx$$

$$=\; \frac{1}{\Delta x_i} \int_{x_{i-1/2}}^{x_{i+1/2}} \bar{v}_j(x) \, dx$$

$$=\; \frac{1}{\Delta x_i} \left[\; Q_r(x_{i+1/2};\bar{W}_j) - Q_r(x_{i-1/2};\bar{W}_j) \; \right]$$

$$=\; \frac{1}{\Delta x_i} \left[\; \bar{W}_j(x_{i+1/2}) - \bar{W}_j(x_{i-1/2}) \; \right]$$

$$=\; \bar{w}_{ij} \;.$$

This property is necessary in order that our numerical scheme (5.12) remain conservative, as explained in the one–dimensional case in Chapter 4, and results directly from the various definitions in the reconstruction (5.14–15), and the fact that the Q_r's are interpolating polynomials. It is the adaptive–stencil algorithm (4.16) that enables this reconstruction (for sufficiently small h) to be high–order accurate on any domain where $w(x,y)$ is smooth, even if that region is near one in which $w(x,y)$ is discontinuous. Furthermore, algorithm (4.16) is ultimately responsible for the adequate resolution of a discontinuity, near which the "jumps" in $R^2(x,y; \bar{w})$ at cell boundaries become large relative to the mesh spacing.

We further note that the error coefficient $e(x,y)$ in (5.8), due to this reconstruction, becomes discontinuous at points where there is a change of orientation in the stencil of the associated interpolation. This discontinuity may occur at critical points of the function and/or its derivatives. It is clear that when $e(x,y)$ fails to be Lipschitz continuous at a point, the truncation error of the approximate flux differ-

91

ence in (5.10c) is only $O(h^{r+1})$. We therefore expect the cumulative pointwise error due to N applications of the operator E_h to be only $O(h^{r-1})$ at such points, but to remain $O(h^r)$ away from these points. Owing to the essentially non–oscillatory nature of E_h, it is reasonable to expect the number of points at which $e(x,y)$ fails to be Lipschitz continuous to remain bounded as $h \rightarrow 0$. In this case, we see that the cumulative error of our numerical scheme is $O(h^{r-1})$ in the L_∞ norm and $O(h^r)$ in the L_1 norm.

It should now be clear that this reconstruction procedure is not a "dimension–by–dimension" operation which simply employs two one–dimensional interpolation stencils, each based on the cell averages $\{\bar{\bar{w}}\}$. Such an approach would not take into account the necessary high–order cross derivatives required for a two–dimensional polynomial expansion. In fact, it is clear upon reviewing equation (5.14e), which represents a purely one–dimensional reconstruction, from which we recover not point values but $\{\bar{v}_j\}$, line averages in y. Though as a function in x, these line averages are high–order approximations to $\{\bar{w}_j\}$ as given by (5.14f), as a pointwise function in y, we have from the definition (5.14d),

$$\bar{w}_j(x) = w(x,y_j) + O(\Delta x^r, \Delta y^2) \ .$$

Thus, for a general nonlinear flux, regardless of the number of cell averages we use in a one–dimensional stencil, any finite–volume scheme based on approximation by the overlapping of two such stencils is inherently limited to second–order accuracy. It is the second stage of the reconstruction which acts upon these line averages that then accounts for the high–order approximation to gradients in y. Figure 11 depicts a "truly two–dimensional" fourth–order stencil as it might be chosen by the algorithm (5.14–15).

Lastly, we remark that this procedure does not require uniformity of the mesh and that we shall extend this scalar reconstruction to vector–valued functions and to curvilinear co–ordinates in the following sections.

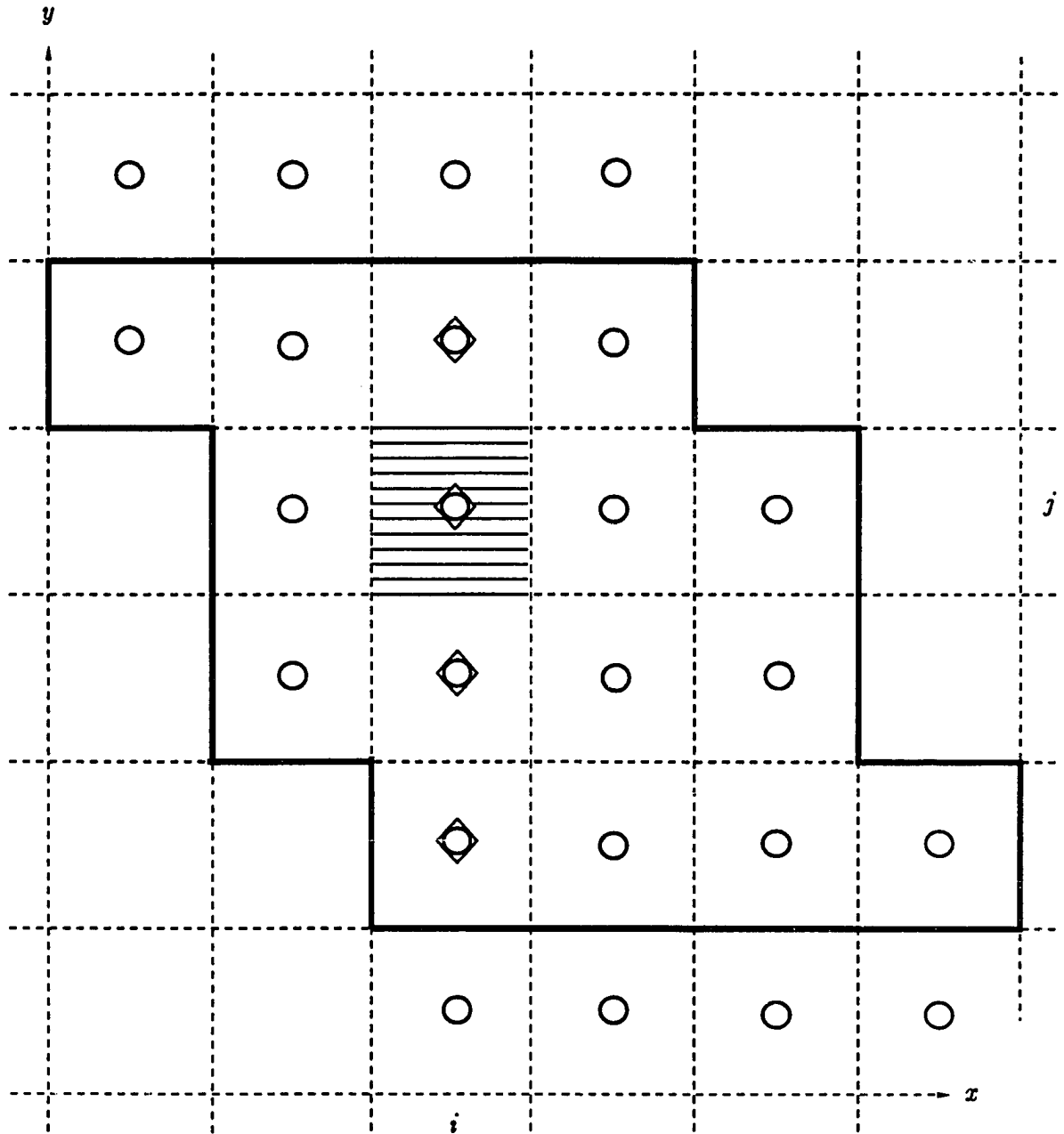Figure 11. The circles represent the stencils chosen in the x-direction for each j, for i fixed, using the cell-averages. The diamonds then represent the stencil chosen in the y-direction, using the line-averages $\bar{v}_j(x_i)$, generated by the first step.Thus, the information entering into the polynomial approximation within the shaded cell (i,j) comes from the two-dimensional stencil within the bold outline.

93

5.4 Systems of Conservation Laws

In this section, we extend the scalar reconstruction procedure of Section 5.3 to solutions of hyperbolic systems of conservation laws. To this end, we now reconsider the IVP (5.1), whose solution u is a vector of m components, as are the fluxes $f(u)$ and $g(u)$. We now wish to develop a vector reconstruction operator, denoted by $\mathbf{R}^2$, which will reconstruct a set $\{\bar{u}_{ij}\}$ of vector-valued cell averages to high-order pointwise accuracy. Recall from Section 4.5, due to the potential collision of discontinuities in time, it was reasoned that a component-wise reconstruction should be avoided. In the two-dimensional case, where discontinuities can intersect in space as well as collide in time, it becomes even more incumbent upon us to utilize the notion of characteristic reconstruction. We turn now to the discussion of this procedure as it applies to a function of two spatial variables.

For the purpose of clarity, we begin the discussion of a two-dimensional characteristic reconstruction by considering the constant coefficient case of (5.1), $i.e.$ $f(u) = A u$, $g(u) = B u$, where A and B are constant $m \times m$ matrices:

$$u_t + A u_x + B u_y = 0, \qquad (5.17a)$$

$$u(x,y,0) = u^0(x,y). \qquad (5.17b)$$

We also assume, for now, that our reconstruction takes place within the context of a Cartesian mesh. We note that the eigenvalues $\{\lambda_A^k\}$ of A and $\{\lambda_B^k\}$ of B are constant as are the right eigenvectors $\{r_A^k\}$, $\{r_B^k\}$ and left eigenvectors $\{l_A^k\}$, $\{l_B^k\}$. We assume that these eigenvectors are suitably normalized so that

$$l_A^i \cdot r_A^j = l_B^i \cdot r_B^j = \delta_{ij}. \qquad (5.18a)$$

If we define the kth characteristic variables w_A^k and w_B^k by the dot products

$$w_A^k = l_A^k \cdot u, \quad w_B^k = l_B^k \cdot u, \qquad k = 1, 2, \ldots, m, \quad (5.18b)$$

94

then it follows from (5.18a) that

$$u = \sum_{k=1}^{m} w_A^k r_A^k = \sum_{k=1}^{m} w_B^k r_B^k . \qquad (5.18c)$$

We argued in Section 4.5 that, in the constant–coefficient case, it is advantageous to use these characteristic variables (5.18b) in the reconstruction procedure, rather than u itself. This is due to the fact that, under the transformation (5.18b), the coupled system (5.17a) becomes an uncoupled set of equations, thereby rendering any discontinuity in a particular characteristic variable "undetectable" by another.

Therefore, in the case of a linear system (5.17a), we can describe our vector reconstruction as follows. Given $\{\bar{u}_{ij}\}$, cell averages of a vector u, we begin by defining the cell averages of the characteristic variables in the x–direction by

$$\bar{w}_{A,ij}^k = l_A^k \cdot \bar{u}_{ij} , \qquad \text{for } j \text{ fixed,} \quad \text{all } i , \qquad (5.19a)$$

and then perform the scalar reconstruction given by (5.14e) on these averages. Using the result (5.18c), we can define the first step of our linear vector reconstruction procedure by

$$\mathbf{R}(x;\bar{u}) = \sum_{k=1}^{m} R(x;\bar{w}_A^k) r_A^k \equiv \bar{v}_j(x) , \qquad (5.19b)$$

the right–hand side of which is the vector–valued analogy of $\bar{v}_j(x)$ in (5.14f). In analogy to the two–step procedure in Section 5.3, the reconstruction (5.19b) is performed for all j. For x fixed, we then proceed by approximating the line–average characteristic variables in the y–direction by the dot product

$$\bar{w}_{B,j}^k = l_B^k \cdot \bar{v}_j . \qquad (5.19c)$$

The scalar reconstruction (5.15c) is applied to the values (5.19c) and, for a fixed x, we have a polynomial in y

$$\mathbf{R}(y;\bar{v}(x)) = \sum_{k=1}^{m} R(y;\bar{w}_B^k) r_B^k , \qquad (5.19d)$$

95

which completes our reconstruction for the linear case and we write

$$\mathbf{R}^2(x, y; \bar{u}) \equiv \mathbf{R}(y; \mathbf{R}(x; \bar{u}))$$

$$= \sum_{k=1}^{m} R\left(y; l_B^k \cdot \sum_{p=1}^{m} R(x; l_A^p \cdot \bar{u}) r_A^p\right) r_B^k . \qquad (5.19e)$$

We now wish to generalize the reconstruction procedure (5.19) to the case of a nonlinear system. In the nonlinear case of (5.17a), the matrices $A(u)$, $B(u)$ are now functions of u, as are the eigenvalues $\{\lambda_A^k(u)\}, \{\lambda_B^k(u)\}$, and the eigenvectors $\{r_A^k(u)\}$, $\{r_B^k(u)\}$, $\{l_A^k(u)\}$, and $\{l_B^k(u)\}$. Our extension will require the use of locally defined characteristic variables, in the following manner. In order to reconstruct $u(x, y)$ on the region $(x_{i-1/2}, x_{i+1/2}) \times (y_{j-1/2}, y_{j+1/2})$, we first derive a set of local average characteristic variables $\{\bar{w}_{nj}^k(\bar{u}_{ij})\}$, where n varies in the x-direction. We do this, for a fixed j, by computing dot products of $l_A^k(\bar{u}_{ij})$ with the cell averages $\{\bar{u}_{nj}\}$ associated with intervals in the immediate vicinity of $(x_{i-1/2}, x_{i+1/2})$, i.e.

$$\bar{w}_{nj}^k(\bar{u}_{ij}) = l_A^k(\bar{u}_{ij}) \cdot \bar{u}_{nj} , \qquad \text{for} \quad n = i - q, \ldots, i + q , \qquad (5.20a)$$

where q is the degree of the reconstruction polynomial. We then apply the scalar reconstruction operator R to this set $\{\bar{w}_{nj}^k(\bar{u}_{ij})\}$ of $2q + 1$ variables in which the left eigenvector has been locally "frozen" at the ith location of the jth row of cells. This "local linearization" allows us to apply locally the linear vector reconstruction described in (5.19). The first step in our nonlinear vector reconstruction procedure then becomes

$$\mathbf{R}(x; \bar{u}) = \sum_{k=1}^{m} R(x; \bar{w}^k(\bar{u}_{ij})) r_A^k(\bar{u}_{ij}) \equiv \bar{v}_j(x) . \qquad (5.20b)$$

Upon performing (5.20b) for all j, then, for x fixed, we define a set of local "line-average" characteristic variables $\{\bar{w}_n^k(\bar{v}_j)\}$ in the y-direction :

$$\bar{w}_n^k(\bar{v}_j) = l_B^k(\bar{v}_j) \cdot \bar{v}_n , \qquad \text{for} \quad n = j - q, \ldots, j + q . \qquad (5.20c)$$

96

We complete our "nonlinear" reconstruction by applying the scalar operator R to the set $\{\bar{w}_n^k(\bar{v}_j)\}$ of local variables in (5.19c) which results in

$$\mathbf{R}(y\,;v(x)) \;=\; \sum_{k=1}^{m} R(\,y\,;\bar{w}^k(\bar{v}_j)\,)\, r_B^k(\bar{v}_j)\,. \qquad\qquad (5.20\text{d})$$

Thus, our vector reconstruction operator $\mathbf{R}^2$ is a composition of (5.20b) and (5.20d) and, for a polynomial within a cell (i,j), we write *abstractly*

$$\mathbf{R}^2(x,y\,;\bar{u})_{ij} \;=\; \sum_{k=1}^{m} R\left(\, y\,;l_B^k(\bar{v}_j(x))\cdot\sum_{p=1}^{m} R(\,x\,;l_A^p(\bar{u}_{ij})\cdot\bar{u}\,)\,r_A^p(\bar{u}_{ij})\,\right)\,r_B^k(\bar{v}_j(x))\,, $$
$$(5.20\text{e})$$

which satisfies

$$\mathbf{R}^2(\,x,y\,;\bar{u}(t)) \;=\; u(x,y,t) \,+\, O(h^r)\,, \qquad\qquad (5.20\text{f})$$

wherever u is sufficiently smooth, and which is designed to avoid the oscillatory behavior associated with colliding discontinuities.

5.5   Curvilinear Co–ordinates

In order to apply the scheme (5.12) to realistic two–dimensional problems, it must be suitable to other than rectangular geometries. We therefore wish to generalize the spatial domain of solutions of the IVP (5.1). To this end let

$$x \;=\; x(\xi,\eta)\,, \qquad\qquad y \;=\; y(\xi,\eta)\,, \qquad\qquad (5.21\text{a})$$

denote a smooth transformation from the physical x–y plane to the computational ξ–η plane. The differential relationship between (x,y) and (ξ,η) can be written in the vector form

$$\begin{bmatrix} dx \\ dy \end{bmatrix} \;=\; \begin{bmatrix} x_\xi & x_\eta \\ y_\xi & y_\eta \end{bmatrix} \begin{bmatrix} d\xi \\ d\eta \end{bmatrix}\,, \qquad\qquad (5.21\text{b})$$

where the 2×2 matrix above is the Jacobian of the transformation, and its determinant J is given by

$$J \;=\; x_\xi\, y_\eta \,-\, y_\xi\, x_\eta\,. \qquad\qquad (5.21\text{c})$$

97

Seeking a solution of (5.1) within this context, let $\mathcal{R}_{ij}$ denote a discrete region in physical space which is mapped into a rectangle $(\xi_{i-1/2}, \xi_{i+1/2}) \times (\eta_{j-1/2}, \eta_{j+1/2})$ by the transformation (5.21a), as in Figure 12. With a semi–discrete formulation in mind, appealing to the theory in Chapter 2, we integrate (5.1a) over the region $\mathcal{R}_{ij}$ and note that a weak solution must satisfy

$$\iint_{\mathcal{R}_{ij}} u_t \, dx \, dy = - \oint_C \left[f \, dy - g \, dx \right] \ ,$$

where the form of the right–hand side is due to the application of Green's theorem in the plane, and the contour C is oriented as in Figure 12a. Performing the prescribed integration with the aid of the transformation (5.21), our semi–discrete formulation can be written identically to (5.2a):

$$\frac{\partial}{\partial t} \bar{u}_{ij}(t) = - \frac{1}{a_{ij}} \left[\hat{f}_{i+1/2,j}(t) - \hat{f}_{i-1/2,j}(t) + \hat{g}_{i,j+1/2}(t) - \hat{g}_{i,j-1/2}(t) \right] \ . \quad (5.22a)$$

Under the transformation (5.21), we interpret the cell average $\bar{u}_{ij}(t)$ at time t as

$$\bar{u}_{ij}(t) = \frac{1}{a_{ij}} \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \int_{\eta_{j-1/2}}^{\eta_{j+1/2}} u(\xi,\eta,t) \, J(\xi,\eta) \, d\xi \, d\eta \ , \quad (5.22b)$$

where a_{ij} is the area of $\mathcal{R}_{ij}$. The fluxes $\hat{f}$ and $\hat{g}$ are given by

$$\hat{f}_{i+1/2,j}(t) = \int_{\eta_{j-1/2}}^{\eta_{j+1/2}} \tilde{F}(u(\xi_{i+1/2}, \eta, t)) \, d\eta \ ; \quad (5.22c)$$

$$\hat{g}_{i,j+1/2}(t) = \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \tilde{G}(u(\xi, \eta_{j+1/2}, t)) \, d\xi \ , \quad (5.22d)$$

where

$$\tilde{F}(u) = y_\eta f(u) - x_\eta g(u) \ , \qquad \tilde{G}(u) = x_\xi g(u) - y_\xi f(u) \ , \quad (5.22e)$$

and $f(u)$ and $g(u)$ are the Cartesian flux vectors in (5.1a).

Having defined the necessary terms of our finite–volume formulation in a curvilinear co–ordinate system, we now wish to discretize our spatial and temporal operations. Again, with our Runge–Kutta time discretization, the extension is straight
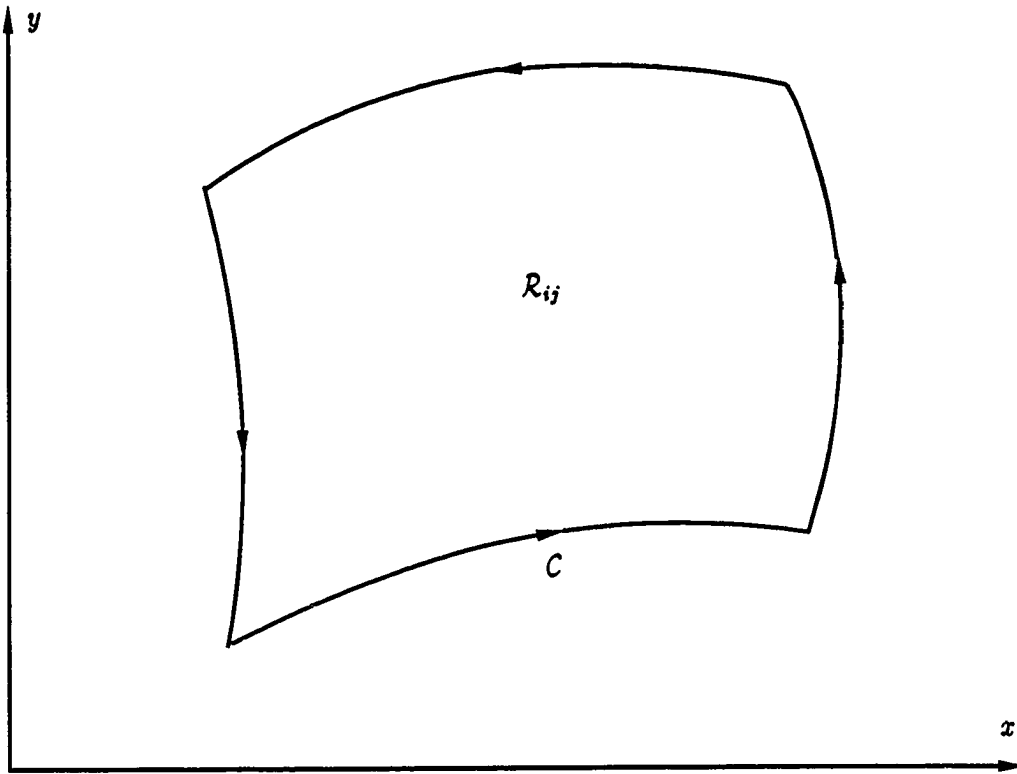
98

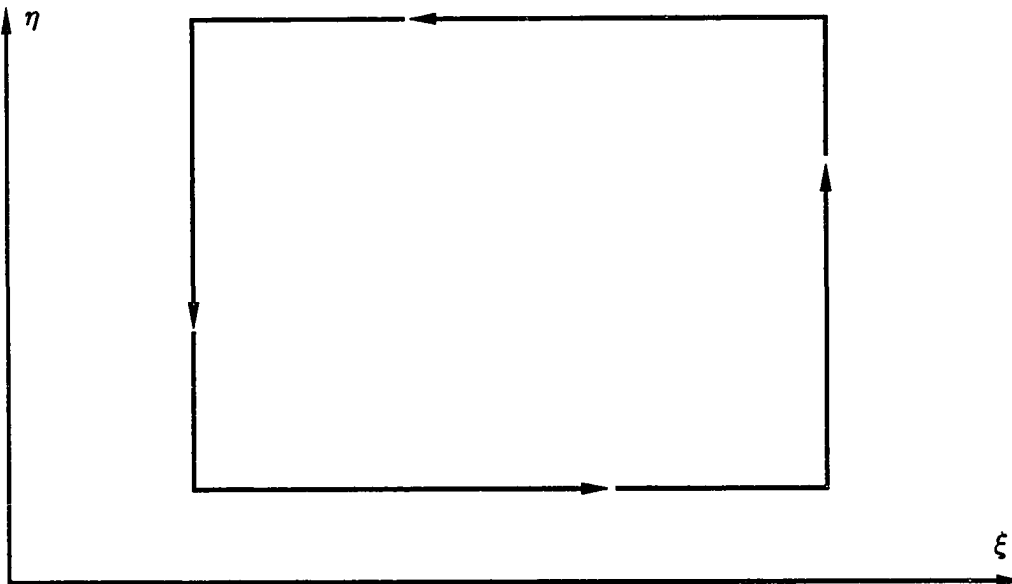Figure 12a.   Curvilinear control volume in physical space.

Figure 12b.   Control volume $\mathcal{R}_{ij}$ as mapped into computational space.

99

forward. As for the spatial operation, we note two basic modifications that must be made.

The first alteration involves the reconstruction operator. Having clearly and consistently defined this procedure in terms of a rectangular mesh, we therefore would like to be able to perform the reconstruction procedure in $\xi-\eta$ space. Given a cell average $\bar{u}_{ij}$, as defined by (5.22b), we see that the quantity $a_{ij}\,\bar{u}_{ij}$ can be interpreted as a "cell average" of the function $u(\xi,\eta)\,J(\xi,\eta)$ on the corresponding computational cell. Therefore if we choose, instead, to reconstruct the set $\{a_{ij}\,\bar{u}_{ij}\}$ of cell averages which are scaled by their respective areas, we transform the reconstruction procedure to rectangular co-ordinates, and in the scalar case, we simply use the procedures described in Section 5.3. The polynomial $p(\xi,\eta)$ we obtain by this approach satisfies

$$p(\xi,\eta) \;=\; R^2(\xi,\eta\,;\,a\,\bar{u}) \;=\; u(\xi,\eta)\,J(\xi,\eta) \;+\; O(h^r) \;,$$

approximating the function $u(\xi,\eta)\,J(\xi,\eta)$ to high order, and therefore must be re-scaled by $J(\xi,\eta)$ in order to yield the pointwise values of $u(\xi,\eta)$ we need to approximate the fluxes (5.22c–d). Thus, we define our curvilinear reconstruction operator $\tilde{R}^2$ by

$$\tilde{R}^2(\xi,\eta\,;\,\bar{u}) \;=\; \frac{1}{J(\xi,\eta)}\,R^2(\xi,\eta\,;\,a\,\bar{u})\,. \qquad (5.23)$$

We can define a curvilinear vector reconstruction operator $\tilde{\mathbf{R}}^2$ in an identical manner, with the understanding that the various eigenvalues and eigenvectors required for this purpose are the corresponding quantities of the Jacobian matrices of $\tilde{F}(u)$ and $\tilde{G}(u)$ in (5.22e), which are, respectively,

$$\tilde{A}(u) \;=\; y_\eta\,A(u) - x_\eta\,B(u)\,, \quad \text{and} \quad \tilde{B}(u) \;=\; x_\xi\,B(u) - y_\xi\,A(u)\,.$$

Our second modification relates to the curvilinear mesh itself. We first note that in the application of second-order schemes, the integration of a numerical flux is achieved by the midpoint rule, and therefore the grid lines which form the cell

100

boundaries can be approximated in a piecewise linear fashion. In this second–order accurate setting, all of the necessary grid quantities are easily approximated. For instance, the transformation "metrics" $\{x_\xi, x_\eta, y_\xi, y_\eta\}$ are approximated to second order by simple finite differences, and the values $\{a_{ij}\}$ are easily computed, all the cells being quadrilaterals.

However, when we desire accuracy that is higher than second–order, a piecewise linear interpretation of a two–dimensional grid will no longer suffice. For example, when we require more than one point in the quadrature which will approximate the flux integrals (5.22c–d), we can no longer assume that the quantities $\{x_\xi, x_\eta, y_\xi, y_\eta\}$ are constant along a given cell boundary. We must therefore assume that the mesh is "truly curvilinear" and account for any change in these grid metrics at each point required in the quadrature. It should also be noted that the curvature of cell boundaries also affects the lengths of cell interfaces, which become the curvilinear analogies of Δy_j, Δx_i in (5.12c–d). This curvature affects the values of the cell areas as well, which are given by

$$a_{ij} = \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \int_{\eta_{j-1/2}}^{\eta_{j+1/2}} J(\xi, \eta)\, d\xi\, d\eta\, . \tag{5.24}$$

Clearly, if the transformation (5.21) avails itself to differentiation and integration in closed form, all these necessary quantities can be obtained exactly. If not, numerical techniques for their high–order approximation would be in order. Our numerical fluxes can then be written

$$\bar{f}_{i+1/2,j}(t) = \frac{|\bar{\eta}|_{i+1/2,j}}{2} \sum_{k=1}^{K} c_k\, \tilde{F}^{R}[\, \tilde{R}^2(\xi_{i+1/2}-0,\, \eta_k;\, \bar{u}(t)),\, \tilde{R}^2(\xi_{i+1/2}+0,\, \eta_k;\, \bar{u}(t))\,] \, ,$$

$$\tag{5.25a}$$

$$\bar{g}_{i,j+1/2}(t) = \frac{|\tilde{\xi}|_{i,j+1/2}}{2} \sum_{k=1}^{K} c_k\, \tilde{G}^{R}[\, \tilde{R}^2(\xi_k,\, \eta_{j+1/2}-0;\, \bar{u}(t)),\, \tilde{R}^2(\xi_k,\, \eta_{j+1/2}+0;\, \bar{u}(t))\,] \, ,$$

$$\tag{5.25b}$$

where $\tilde{R}^2$ is given by (5.23), and the quantities $|\bar{\eta}|_{i+1/2,j}$ and $|\tilde{\xi}|_{i,j+1/2}$ represent the arclengths of the boundaries of $\mathcal{R}_{ij}$ along $\xi = \xi_{i+1/2}$ and $\eta = \eta_{j+1/2}$, respectively.

101

Now, assuming that the transformation (5.21a) is sufficiently smooth, the numerical scheme resulting from the curvilinear formulation (5.22) with numerical fluxes defined by (5.25) will be r^{th}-order accurate as defined by (5.4) in smooth regions and avoid oscillations near steep gradients.

We make one further generalization. It happens that there are a lot of applications of structured computational meshes for which a closed-form transformation is not available. For example, a set of grid points may be initially generated as a solution of a system of differential equations, after which they may then be subjected to some smoothing operator, $e.g.$ Laplacian. In such a case we do not have a set of equations (5.21a) from which to determine all the grid variables necessary for the flux computations (5.25).

However, given such a set of points, we might consider the equivalent of a locally defined set of transformation equations which are derived by polynomial approximation. By this we mean that each "grid line" through a set of points is approximated by piecewise polynomial interpolation of some pre-determined order, and that all the necessary mesh quantities are calculated from these polynomials. As the grid metrics $\{x_\xi, x_\eta, y_\xi, y_\eta\}$ represent the components of outward normals at cell boundaries, a simple calculus argument involving the polynomial approximation to a cell boundary will enable us to compute these quantities at the desired quadrature points. Approximate cell areas are also straight forward, once the four polynomials defining each control volume are determined. Arclengths of cell faces can be approximated by an appropriate integration technique. An approximate Jacobian determinant, which we will denote by J', required for the curvilinear reconstruction procedure (5.23) is also obtainable by the following reasoning. The relationship (5.24) allows us to interpret a cell area a_{ij} as a rectangular "cell average" of $J(\xi, \eta)$. Thus, we first determine a set of high-order approximations to the cell areas, which we denote by $\{a'_{ij}\}$. We then apply our scalar reconstruction

102

algorithm from Section 5.3 to this set and define the approximation J' by

$$J'(\xi,\eta) = R^2(\xi,\eta\,;\,a')\,.$$

We note here that, without a known transformation (5.21a) from which to mathematically determine "sufficient smoothness," we can no longer make the claim concerning formal order of accuracy when this polynomial grid approximation is employed.

5.6 Implementation

In this section, we make a few general remarks concerning the implementation of some of the ideas in the preceding sections. Certainly, as with any numerical scheme, the fine details of its application will depend upon the equations one is solving as well as the given problem. We will make some specific remarks concerning the application of our two–dimensional ENO schemes to the Euler equations of gas dynamics in the following section.

Our first topic concerns the reconstruction procedure. As the adaptive stencil algorithm (4.16) ultimately relies on the entries in a divided–difference table, the obvious choice of polynomial approximation is Newton interpolation. Though it is possible to use the reconstruction operator R^2 to compute the coefficients of a two–dimensional polynomial and then evaluate it at all of the points required for the flux calculations (5.12c–d), instead we choose to implement R^2 in the manner in which it is presented in Section 5.3, *i.e.* in one–dimensional "sweeps." Furthermore, though it is not necessary from an approximation theory viewpoint, the order in which we apply these directional sweeps will depend upon the particular cell face along which we desire a flux computation. For the evaluation of the flux $\hat{f}_{i+1/2,j}(t)$ in (5.12c), the pointwise approximation of u at the Gauss points $\{y_k\}$ along $x = x_{i+1/2}$

103

is determined by

$$u(x_{i+1/2} \pm 0, y_k, t) \approx R(y_k; R(x_{i+1/2} \pm 0; \bar{u}(t))) , \qquad (5.26a)$$

and for the evaluation of $\hat{g}_{i+1/2,j}(t)$ in (5.12d), we employ the composite operation (5.26a) in "reversed" form and achieve the required pointwise values by

$$u(x_k, y_{j+1/2} \pm 0, t) \approx R(x_k; R(y_{j+1/2} \pm 0; \bar{u}(t))) . \qquad (5.26b)$$

Also pertaining to the reconstruction is the issue of its implementation near a computational boundary. We have taken the position that a high–order approximation must reflect available information. For instance, in the case of an assumption of periodicity of the solution, we can allow interpolation stencils to move outside the "boundary" by simply installing the known periodic behavior of the solution into "ghost cells." However, when the solution is not known beyond a boundary, as in the case of a solid wall, we restrict our interpolation stencil to remain within the computational domain.

It was found during numerical experimentation that problems could result from a high–order, one–sided interpolation procedure. In particular, in the case of the Euler equations, if a solution developed a shock which reflected from a wall, oscillatory behavior was noticed in the smooth region near the reflection point between the shock and the wall. Attempting to eliminate such numerical noise, one might suggest some sort of test for a "desirable reconstruction" which, when failed, will result in a local reduction in the order of interpolation. Owing to the recursive nature of the Newton interpolation procedure, such a test can be readily applied during the actual "building" of the polynomial. One such test ([42]) simply checks for "over–shoots" or "under–shoots." In a one–dimensional setting, this is equivalent to requiring the endpoint values of the polynomial $p_i(x)$ which approximates $u(x)$ on $[x_{i-1/2}, x_{i+1/2}]$ not to over–shoot a larger adjacent cell average, or to under–shoot a smaller one,

104

as in Figure 13a. We can so restrict $p_i(x)$ by requiring that it satisfy

$$(\bar{u}_{i+1} - \bar{u}_i) (\bar{u}_{i+1} - p_i(x_{i+1/2})) \geq 0 , \qquad (5.27a)$$

$$(\bar{u}_i - \bar{u}_{i-1}) (p_i(x_{i-1/2}) - \bar{u}_{i-1}) \geq 0 . \qquad (5.27b)$$

We define any $p_i(x)$ which satisfies (5.27) to be a "desirable" reconstruction. If either of the inequalities (5.27) is not satisfied (Figure 13b), then the degree of $p_i(x)$ is reduced. Clearly, this restriction will never require any less than a locally linear reconstruction, since a piecewise linear polynomial is always locally monotone, thereby making the scheme "locally TVD."

As for the application of the boundary conditions themselves, such conditions as inflow or outflow are handled in a standard manner, depending upon the direction of the characteristic wave speeds at these types of boundaries. On the other hand, a wall condition is relatively simple to handle. Because we actually approximate the solution on the boundary of each cell, those point values $\{u_w\}$ along a solid wall are treated with an appropriate boundary condition and the numerical flux $\hat{f}_w$ on the wall is simply

$$\hat{f}_w = f (B(u_w)) , \qquad (5.28)$$

with $B(\cdot)$ denoting that a wall condition has been applied. A more detailed account of boundary conditions for the Euler equations is deferred to Section 5.7.

We make a final comment here concerning the interpretation of the Riemann problem in this two–dimensional formulation. Recently, other authors, e.g. Roe [49], have attempted to develop genuinely multi–dimensional characteristic based schemes, but these new ideas have not yet sufficiently matured. We therefore take the "conventional" approach which extends the solution of the Riemann problem into two dimensions by simply solving the one–dimensional problem in a manner which resolves waves which propagate normal to a cell boundary. In the following section, we detail this approach as it pertains to the Euler equations of gas dynamics.
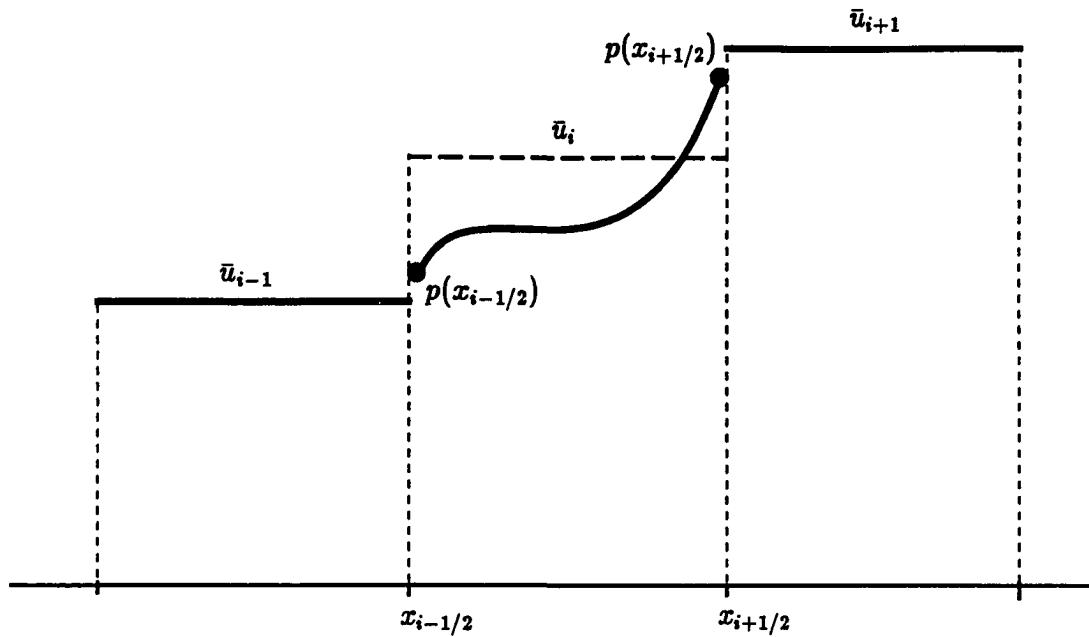
105

Figure 13a.   A polynomial reconstruction which satisfies inequalities (5.27).
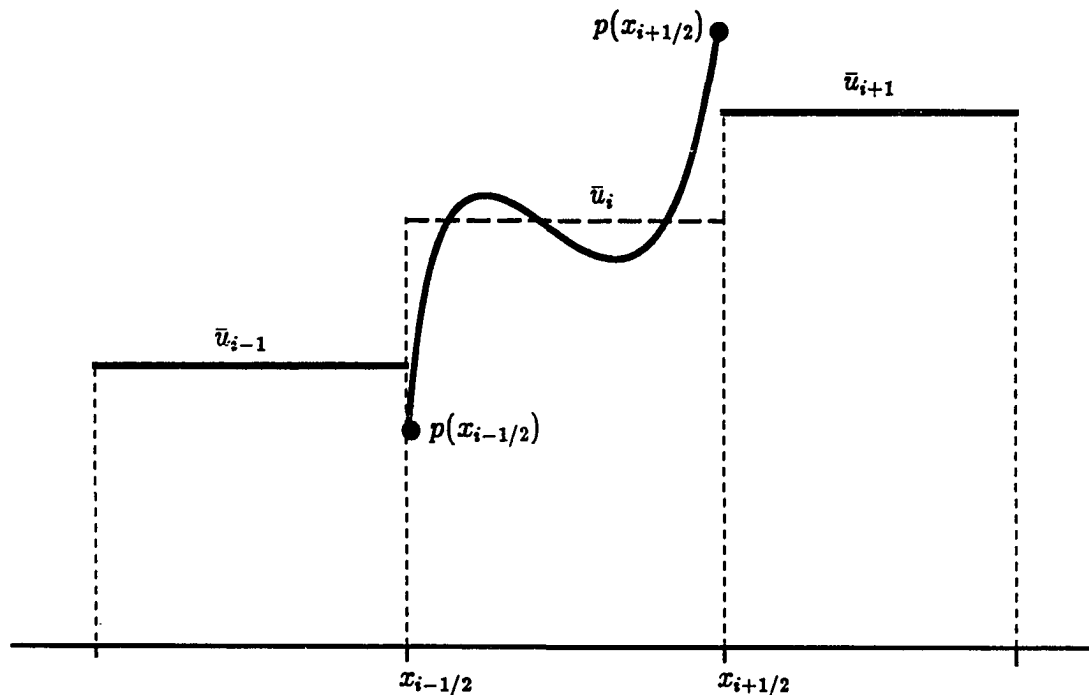

Figure 13b.   A polynomial reconstruction which violates both inequalities (5.27).

106

5.7 Euler Equations of Gas Dynamics

We now discuss the application of our two–dimensional high–order ENO scheme (5.12) to the Euler equations of gas dynamics. In a Cartesian frame of reference, we write them in conservation form

$$U_t + F(U)_x + G(U)_y = 0, \qquad (5.29a)$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho E \end{bmatrix}, \quad F(U) = \begin{bmatrix} \rho u \\ \rho u^2 + P \\ \rho u v \\ (\rho E + P)u \end{bmatrix}, \quad G(U) = \begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 + P \\ (\rho E + P)v \end{bmatrix}. \qquad (5.29b)$$

The quantities ρ, P, and E are the density, pressure, and total specific energy, respectively, and u and v are the Cartesian components of the velocity vector $\vec{V}$. We close the system (5.29a) of four equations with the polytropic equation of state

$$P = (\gamma - 1)\rho(E - \tfrac{1}{2}V^2), \qquad (5.29c)$$

where γ is the ratio of specific heats and $V^2 = u^2 + v^2$.

In quasi–linear form, we can write (5.29a) as

$$U_t + A U_x + B U_y = 0, \qquad (5.30a)$$

where $A(U)$ and $B(U)$ are the 4×4 Jacobian matrices

$$A(U) = \frac{\partial F}{\partial U}, \qquad B(U) = \frac{\partial G}{\partial U}. \qquad (5.30b)$$

Appealing to the co–ordinate transformation (5.21), we rewrite (5.29a) as

$$\tilde{U}_t + \tilde{F}(U)_\xi + \tilde{G}(U)_\eta = 0, \qquad (5.31a)$$

where

$$\tilde{F}(U) = y_\eta F(U) - x_\eta G(U), \qquad \tilde{G}(U) = x_\xi G(U) - y_\xi F(U), \qquad (5.31b)$$

107

and $\tilde{U} = U\, J(\xi, \eta)$, where J is the Jacobian determinant of the co-ordinate transformation, and $F(U)$ and $G(U)$ are the Cartesian flux vectors in (5.29b). In quasi-linear form, (5.31a) can be written

$$\tilde{U}_t + \tilde{A}\, U_x + \tilde{B}\, U_y = 0 \,, \tag{5.31c}$$

where the Jacobian matrices $\tilde{A}(U)$ and $\tilde{B}(U)$ are related to their Cartesian counterparts in (5.30b) by

$$\tilde{A}(U) = y_\eta\, A(U) - x_\eta\, B(U) \,, \qquad \tilde{B}(U) = x_\xi\, B(U) - y_\xi\, A(U) \,. \tag{5.31d}$$

We turn now to describe the eigenvalues and eigenvectors of $\tilde{A}(U)$ and $\tilde{B}(U)$. Now, since these matrices (5.31d) are both of the form of a generic matrix $M(U)$, defined by

$$M(U) \equiv \alpha\, A(U) + \beta\, B(U) \,, \tag{5.32}$$

with α and β independent of U, it is therefore convenient to describe the necessary quantities in terms of $M(U)$. We assume that the grid metrics are suitably normalized so that α, β will satisfy

$$\sqrt{\alpha^2 + \beta^2} = 1 \,.$$

The eigenvalues of $M(U)$ are

$$\lambda^1(U) = \omega - c \,, \qquad \lambda^2(U) = \omega = \lambda^3(U) \,, \qquad \lambda^4(U) = \omega + c \,, \tag{5.33a}$$

where

$$\omega = \alpha\, u + \beta\, v \,, \qquad c = \sqrt{\frac{\gamma P}{\rho}} \,, \tag{5.33b}$$

and c is the sound speed. We note that the eigenvalues are not distinct, but nonetheless give rise to linearly independent eigenvectors, as required in our definition of a hyperbolic system.

As the equations (5.29a-b) and eigenvalues (5.33a) are ordered, the corresponding right eigenvectors are

108

$$r^1(U) = \frac{1}{c\sqrt{2}} \begin{bmatrix} 1 \\ u - \alpha c \\ v - \beta c \\ H - \omega c \end{bmatrix}, \qquad r^2(U) = \frac{1}{c} \begin{bmatrix} 1 \\ u \\ v \\ \frac{1}{2}V^2 \end{bmatrix},$$

$$\text{(5.33c)}$$

$$r^3(U) = \begin{bmatrix} 0 \\ \beta \\ -\alpha \\ \beta u - \alpha v \end{bmatrix}, \qquad r^4(U) = \frac{1}{c\sqrt{2}} \begin{bmatrix} 1 \\ u + \alpha c \\ v + \beta c \\ H + \omega c \end{bmatrix},$$

where H is the total enthalpy, given by

$$H = \frac{E + P}{\rho} = \frac{c^2}{\gamma - 1} + \frac{V^2}{2}. \qquad \text{(5.33d)}$$

The left eigenvectors $\{l^k(U)\}$, corresponding to the $\{r^k(U)\}$ in (5.33c), are

$$l^1(U) = \frac{1}{c\sqrt{2}} \left[\frac{(\gamma-1)}{2} V^2 + \omega c, \quad (1-\gamma)u - \alpha c, \quad (1-\gamma)v - \beta c, \quad \gamma - 1 \right]$$

$$l^2(U) = \frac{1}{c} \left[c^2 - \frac{(\gamma-1)}{2} V^2, \quad (\gamma-1)u, \quad (\gamma-1)v, \quad 1 - \gamma \right]$$

$$\text{(5.33e)}$$

$$l^3(U) = \left[\alpha v - \beta u, \quad \beta, \quad -\alpha, \quad 0 \right]$$

$$l^4(U) = \frac{1}{c\sqrt{2}} \left[\frac{(\gamma-1)}{2} V^2 - \omega c, \quad (1-\gamma)u + \alpha c, \quad (1-\gamma)v + \beta c, \quad \gamma - 1 \right]$$

The particular form taken by these left and right eigenvectors is due in part to well-conditioning considerations during their derivation, as they are required to satisfy the orthonormal relationship

$$l^j(U) \cdot r^k(U) = \begin{cases} 0, & j \neq k \\ 1, & j = k \end{cases}.$$

We now have all of the necessary information for the application of our two-dimensional scheme in the formulation (5.22), under the co-ordinate transformation

109

(5.21). Along each $t = $ constant line, we use (5.33e) for the evaluation of the local subsets of characteristic variables, to which we apply the vector reconstruction

$$\tilde{\mathbf{R}}^2(\xi,\eta\,;\bar{U}) \;=\; \frac{1}{J(\xi,\eta)}\,\mathbf{R}^2(\xi,\eta\,;a\bar{U})\,, \qquad\qquad (5.34)$$

where $\mathbf{R}^2$ is the Cartesian reconstruction (5.20). The appropriate interpretation of (α,β) as $(y_\eta,-x_\eta)$ or $(-y_\xi,x_\xi)$ is determined by the direction of the interpolation at a particular stage of the reconstruction.

The high-order pointwise information in (5.34) is then used to evaluate the fluxes (5.25). At each Gauss point η_k along $\xi = \xi_{i+1/2}$ and each ξ_k along $\eta = \eta_{j+1/2}$, we are required to determine

$$\tilde{F}^{\mathrm{R}}\left[\,\tilde{\mathbf{R}}^2(\xi_{i+1/2}-0,\eta_k\,;\bar{U}(t)),\,\tilde{\mathbf{R}}^2(\xi_{i+1/2}+0,\eta_k\,;\bar{U}(t))\,\right], \qquad (5.35a)$$

$$\tilde{G}^{\mathrm{R}}\left[\,\tilde{\mathbf{R}}^2(\xi_k,\eta_{j+1/2}-0\,;\bar{U}(t)),\,\tilde{\mathbf{R}}^2(\xi_k,\eta_{j+1/2}+0\,;\bar{U}(t))\,\right], \qquad (5.35b)$$

respectively. Each of these Riemann fluxes then becomes a contribution to the quadratures which determine the numerical fluxes (5.25).

We now describe the Riemann problem as it is interpreted in our two-dimensional scheme for the Euler equations. As mentioned in the previous section, our implementation will be one-dimensional, but in a directional manner as to account for characteristic wave propagation normal to a cell interface. Clearly the value $\omega = \alpha u + \beta v$ is the component of the flow velocity vector $\vec{V}$ which is normal to a cell boundary at a given point, and has the same role in the two-dimensional problem as that of u in the one-dimensional problem (Section 4.6). The quantity that distinguishes the present case is $\nu = \beta u - \alpha v$, the tangential component of $\vec{V}$ in this reference frame.

We recall from Section 4.6 that, in one dimension, the Riemann IVP for the Euler equations gives rise to three waves, separating four constant states (U_L, U_1, U_2, U_R).

110

In our directional application, we must now account for four characteristic waves. As in the one–dimensional case, we have left and right acoustic waves, each of which is either a shock or a rarefaction. However, due to the repeated eigenvalues λ^2, λ^3, it is now possible to have two interior waves which propagate at the same speed. One of these interior waves is the contact discontinuity that we would ordinarily expect in the one–dimensional case. The other interior wave is due to the presence of the tangential velocity component ν. Any difference in ν_L and ν_R will give rise to a shear wave which is coincident with the contact discontinuity. Therefore, there are still effectively three waves separating four states as originally depicted in Figure 7b in Section 4.6. We can therefore solve this problem in the same manner as the one–dimensional problem, with the requirement that $\nu_1 = \nu_L$ and $\nu_2 = \nu_R$.

With the interpretation of the Riemann problem outlined above, the extension of any characteristic–based method to two–dimensional flows is straight forward. In particular, we outline here such an extension for Roe's method. For this purpose, let $\{\lambda^k(\xi), r^k(\xi), l^k(\xi)\}$ denote the eigenvalues and eigenvectors evaluated with respect to the Jacobian matrix $\tilde{A}$, and $\{\lambda^k(\eta), r^k(\eta), l^k(\eta)\}$ the corresponding quantities associated with $\tilde{B}$. Then, given two values (U_L, U_R) at η_k along $\xi = \xi_{i+1/2}$, or at ξ_k along $\eta = \eta_{j+1/2}$, we approximate the Riemann fluxes (5.35) by

$$\tilde{F}^{ROE}(U_L, U_R) = \frac{1}{2}\left[\tilde{F}(U_L) + \tilde{F}(U_R)\right] - \frac{1}{2}\sum_{k=1}^{4} \hat{\delta}^k(\xi)\,|\hat{\lambda}^k(\xi)|\,\hat{r}^k(\xi), \quad (5.36a)$$

$$\tilde{G}^{ROE}(U_L, U_R) = \frac{1}{2}\left[\tilde{G}(U_L) + \tilde{G}(U_R)\right] - \frac{1}{2}\sum_{k=1}^{4} \hat{\delta}^k(\eta)\,|\hat{\lambda}^k(\eta)|\,\hat{r}^k(\eta), \quad (5.36b)$$

where

$$\hat{\delta}^k(\xi) = \hat{l}^k(\xi)\cdot(U_R - U_L), \qquad \hat{\delta}^k(\eta) = \hat{l}^k(\eta)\cdot(U_R - U_L), \qquad (5.36c)$$

and the quantities $\{\hat{\lambda}^k(\xi), \hat{l}^k(\xi), \hat{r}^k(\xi)\}$ and $\{\hat{\lambda}^k(\eta), \hat{l}^k(\eta), \hat{r}^k(\eta)\}$ are evaluated with respect to $\tilde{A}(\hat{U})$ and $\tilde{B}(\hat{U})$, respectively. This quantity $\hat{U} = U(U_L, U_R)$ is the "Roe–average" value of U_L, U_R as previously discussed in Chapter 4. Now, in order

111

to evaluate (5.36a–b), all the quantities (5.33) must be evaluated with respect to this average. We see that the particular values we need are $\hat{u}, \hat{v}, \hat{c}$, and $\hat{H}$, which are given by

$$\hat{u} = \frac{\sqrt{\rho_L}\,u_L + \sqrt{\rho_R}\,u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} , \qquad\qquad \hat{v} = \frac{\sqrt{\rho_L}\,v_L + \sqrt{\rho_R}\,v_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} ,$$

$$\text{(5.36d)}$$

$$\hat{H} = \frac{\sqrt{\rho_L}\,H_L + \sqrt{\rho_R}\,H_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} , \qquad \hat{c} = \sqrt{(\gamma - 1)\,(\hat{H} - \tfrac{1}{2}(\hat{u}^2 + \hat{v}^2))} .$$

We recall here that the fluxes computed by the approximations (5.36) can result in the attainment of unphysical solutions (Section 3.4), which most commonly arise when one of the acoustic eigenvalues $\lambda^1(U)$ or $\lambda^4(U)$ is such that

$$\lambda(U_L) < 0 < \lambda(U_R) .$$

In such a case, the fluxes (5.36a–b) must be modified with an appropriate entropy correction. There are many ways of achieving this correction, and different considerations might govern the choice of particular one. We therefore defer the details of such an entropy fix for (5.36) until such time that it is needed (Section 6.3).

We now address the issue of boundary conditions. To this end, let D denote the domain of our solution, $Int(D)$ its interior, and ∂D its boundary. It happens that the extension of our two–dimensional ENO scheme to boundaries is quite natural. The pointwise approximation of the solution that is required on the boundaries of every cell is required as well on any cell boundary which interfaces with ∂D, and is achieved in a similar manner. The fluxes on ∂D are then calculated in a fashion which accounts for characteristic signal propagation, as is any flux in $Int(D)$. The only significant difference is in the application of the boundary conditions themselves. The two types of conditions we must address are those of inflow/outflow and solid walls, and we do so by example.

112

Suppose an inflow/outflow boundary exists at $\xi = \xi_b$ whose orientation with regard to $Int(D)$ is as illustrated in Figure 14a. Appealing to characteristic theory, we are required to specify those characteristic variables $q^k(U) = l^k \cdot U$ whose associated eigen–speeds dictate wave motion towards $Int(D)$, while those associated with outward traveling waves must remain unspecified.

In order to adequately account for the required inward wave propagation at such a boundary, we rely on the imposition of the boundary condition itself and the solution of the associated local Riemann problem (Figure 14b). Let U_I denote the limiting value of the solution on ∂D from $Int(D)$, and U_E the limiting value from the exterior. As we take the position that any value of the solution must reflect available information, U_I is therefore determined from the high–order reconstruction procedure, as is any other interior value, *i.e.*

$$U_I(\xi_b, \eta, t) \;=\; \tilde{\mathbf{R}}^2(\,\xi_b - 0, \eta\,;\, \bar{U}(t)\,) \;. \tag{5.37a}$$

The only modification we make to (5.37a) is to ensure that only the solution in $Int(D)$ is made available for this reconstruction on ∂D. This may occasionally require that an interpolation stencil pass through an unsmooth region, causing oscillations, in which case a local reduction in the degree of the interpolating polynomial would be in order (See Section 5.6). It is the boundary condition which must determine U_E and the manner in which we do so is problem dependent. For example, it might be appropriate to set U_E equal to a known free–stream value U_∞. It is ultimately up to the Riemann solver to detect those characteristic waves which will influence the solution near the boundary, and our flux $\hat{f}_k(t)$ through the point (ξ_b, η_k) in Figure 14a is

$$\hat{f}_k(t) \;=\; \tilde{F}^{\mathrm{R}}\,[\,U_I(\xi_b, \eta_k, t), U_E(\xi_b, \eta_k, t)\,] \;. \tag{5.37b}$$

Our second example involves a solid wall, which we assume to be aligned with the curve $\eta = \eta_b$, as in Figure 15. As in the previous example, the high–order
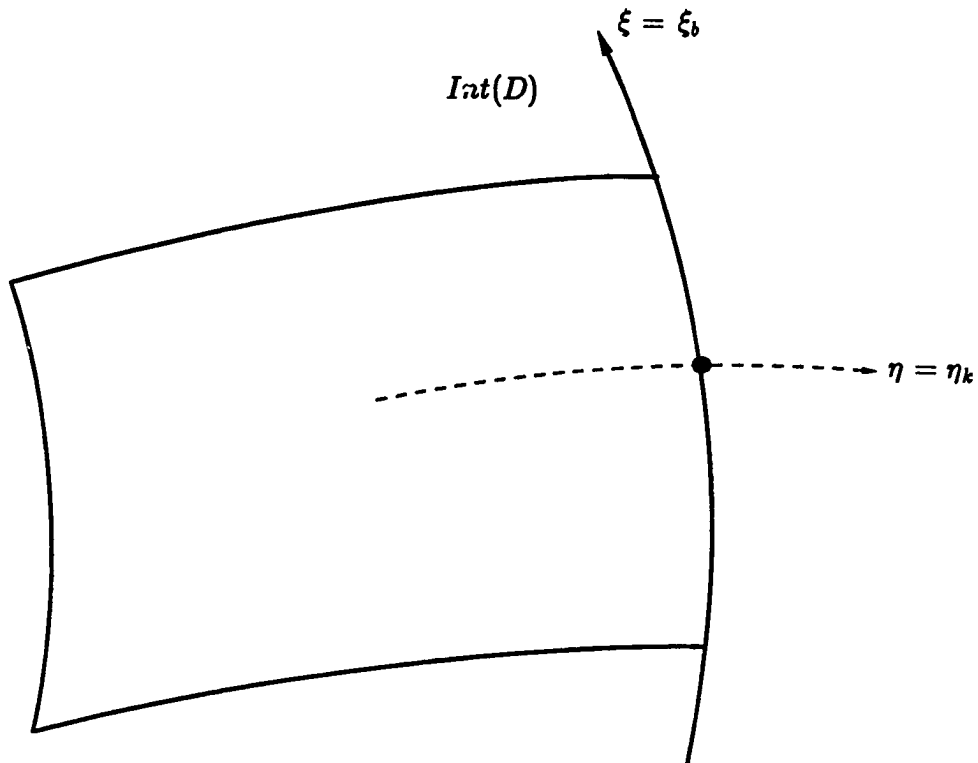
113

Figure 14a. The Euler equations are solved along $\eta = \eta_k$, where (ξ_b, η_k) is a point on an inflow/outflow boundary.
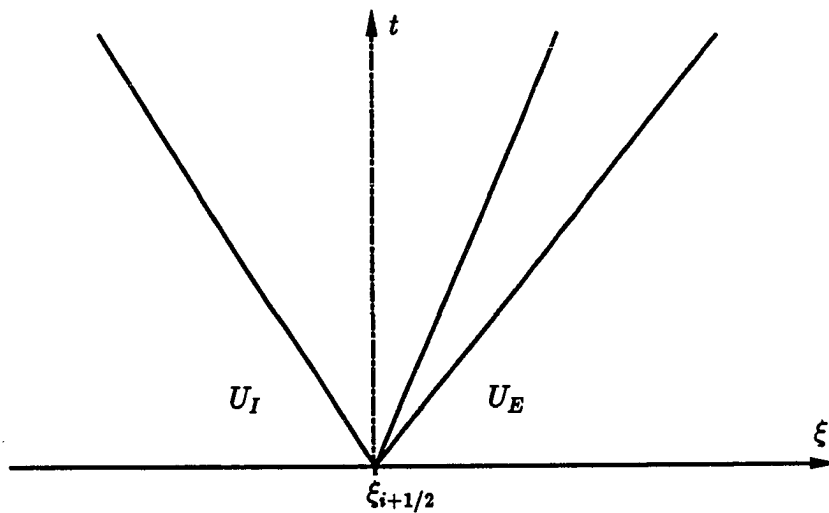

Figure 14b. Characteristic wave propagation at an inflow/outflow boundary is determined by the solution of a Riemann problem. The initial values U_I, U_E are determined from the interior and the boundary condition, respectively.
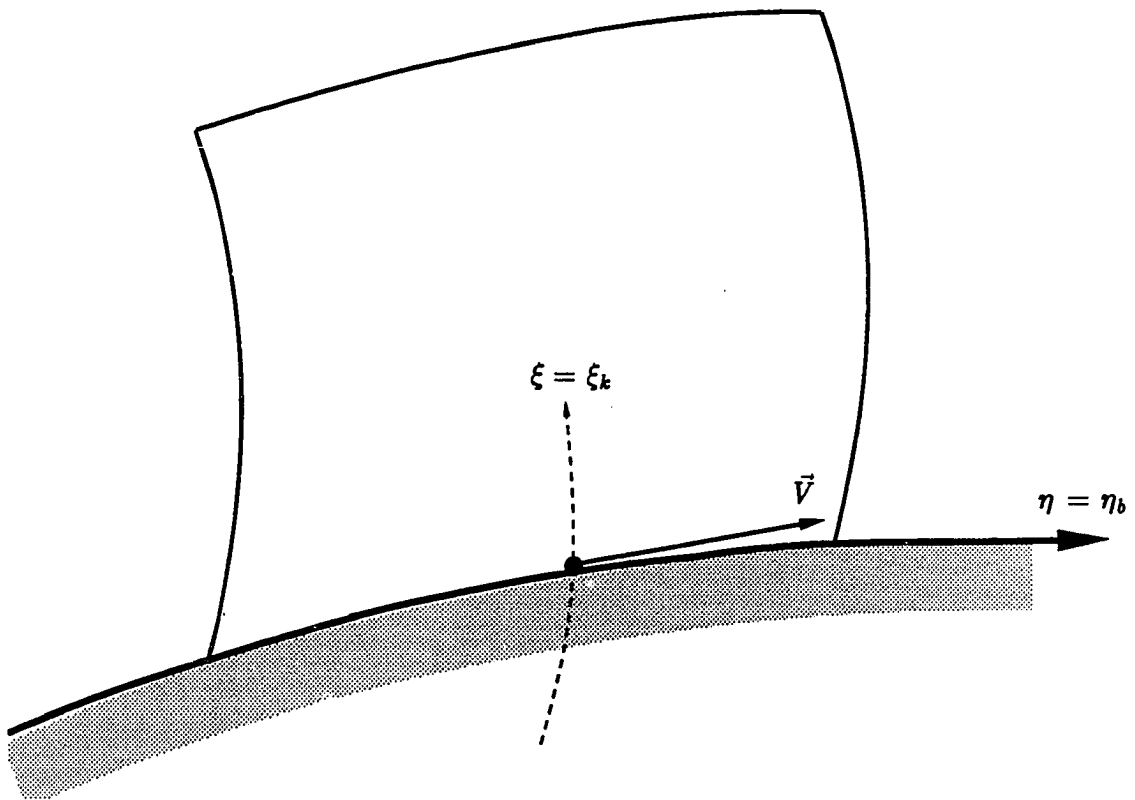
114

Figure 15. The Euler equations are solved along $\xi = \xi_k$, where (ξ_k, η_b) is a point on a solid wall. In this case, tangency is enforced and the flux is not determined by the solution of a Riemann problem.

pointwise approximation to U_I along the boundary is determined by

$$U_I(\xi, \eta_b, t) = \tilde{\mathbf{R}}^2(\xi, \eta_b + 0; \bar{\bar{U}}(t)) \ , \tag{5.37c}$$

where information for polynomial interpolation is restricted to $Int(D)$. At the wall we insist that there be no normal velocity, *i.e.* the wall is coincident with a streamline. Thus, along $\xi = \xi_k$, we solve

$$\tilde{U}_t + \tilde{G}(U)_\eta = 0 \ , \tag{5.38a}$$

subject to the boundary condition

$$-y_\xi u + x_\xi v = 0 \ . \tag{5.38b}$$

Let U_W denote the solution U_I in (5.37c), with the boundary condition (5.38b) having been applied. Unlike the inflow/outflow case, there is no exterior value U_E along $\eta = \eta_k$ which can influence the solution interior to the wall. Therefore we can write our flux along the lower boundary of the pictured cell by direct integration of (5.38a),

$$\hat{g}(t) = \int_{\xi_{i-1/2}}^{\xi_{i+1/2}} \tilde{G}(U_W(\xi, \eta_b, t)) \, d\xi \ .$$

In quadrature form then, the flux $\hat{g}_k(t)$ through the point (ξ_k, η_b) is simply

$$\hat{g}_k(t) = \tilde{G}(U_W(\xi_k, \eta_b, t)) \ . \tag{5.38c}$$

Now, since

$$\tilde{G}(U) = -y_\xi F(U) + x_\xi G(U) \ ,$$

using the Cartesian fluxes (5.29b) and the enthalpy relation (5.33d), we can rewrite $\tilde{G}(U)$ as

$$\tilde{G}(U) = \rho(-y_\xi u + x_\xi v) \begin{bmatrix} 1 \\ u \\ v \\ H \end{bmatrix} - y_\xi \begin{bmatrix} 0 \\ P \\ 0 \\ 0 \end{bmatrix} + x_\xi \begin{bmatrix} 0 \\ 0 \\ P \\ 0 \end{bmatrix} \ . \tag{5.38d}$$

116

Then upon application of the tangency condition (5.38b), we evaluate the flux (5.38c) at the point (ξ_k, η_b) on the wall by

$$
\hat{g}_k(t) \;=\; \begin{bmatrix} 0 \\ -y_\xi\, P_W \\ x_\xi\, P_W \\ 0 \end{bmatrix} \;, \tag{5.38e}
$$

where the wall pressure P_W is ultimately determined by the interior reconstruction (5.39c).

We make two final remarks concerning the scheme. First, we will *assume* that the scheme (5.12) is stable when the time step is required to satisfy

$$
\Delta t \;=\; \text{CFL}\, \frac{a_{ij}}{\max_U \left(|\lambda_A^k(U)|\, \Delta y \,+\, |\lambda_B^k(U)|\, \Delta x \right)} \;, \tag{5.39}
$$

where "CFL" is a positive number bounded above by the appropriate value "C_r" in (4.34).

Secondly, we must ensure that, during the course of a calculation, the values of density and pressure remain positive. This is a consideration which arises from the high–order reconstruction operator $\mathbf{R}^2$. For instance, when an interpolation stencil is forced to the interior near a boundary, it might cross a strong shock or near the center of a strong expansion fan. The steep gradients in density and pressure encountered in such an interpolation could easily cause a pointwise polynomial approximation of these variables to be negative within the interval of interest. In [26], a positivity check is suggested which requires consistency with the coefficients of the Taylor expansion of $\rho(x)$ and $P(x)$. We choose instead to set lower bounds $\{\rho_{\min}, P_{\min}\}$ on these variables, and if the desired interface values at $x = x_{i+1/2}$ do not satisfy

$$
\rho(x_{i+1/2}) \geq \rho_{\min} \;, \qquad P(x_{i+1/2}) \geq P_{\min} \;, \tag{5.40}
$$

then the degree of the reconstruction polynomial is locally reduced. If it is not possible to determine $\{\rho_{\min}, P_{\min}\}$ from *a priori* knowledge of a given problem,

117

then these values must be arbitrarily set to some values which are small relative to the scale of the problem. Experience suggests that the positivity condition itself is not as important as ensuring that the manner in which it is enforced enables the numerical scheme to remain conservative. This is accomplished by the local reduction in the degree of polynomial interpolation.

Chapter 6

Numerical Experiments

We present in this chapter several examples of numerically computed solutions using our two–dimensional ENO scheme as well as its various extensions. We have performed experiments on scalar equations in order to test for the computational order of accuracy, and for problems testing our extension to hyperbolic systems we solve the two–dimensional Euler equations of gas dynamics. The solutions we obtain for the Euler equations represent the first successful application of high–order ENO schemes to boundary–value problems with solid walls.

6.1   Linear Advection

In order to test the scheme (5.12) for its accuracy, we solve the two–dimensional linear advection equation

$$u_t + u_x + u_y = 0 , \quad t > 0 , \tag{6.1a}$$

with initial data

$$u(x,y,0) = \tfrac{1}{2} \cos \pi \left(x + y \right) + \tfrac{1}{2} . \tag{6.1b}$$

The solution of (6.1) is 2–periodic in x and y for all time. By restricting our computational domain to $-1 \leq x, y \leq 1$, we thereby make the boundaries 2–periodic also, effectively removing them from consideration. We note here that even though we are solving a linear equation, the scheme (5.12) applied to (6.1a) is

119

still non-linear, due to the adaptive stencil algorithm of the reconstruction operator R^2.

The exact solution of (6.1) can be easily calculated one-dimensionally in terms of the variable $\xi = x + y$, and can be written

$$u(x,y,t) = \tfrac{1}{2} \cos \pi \left(x + y - 2t \right) + \tfrac{1}{2}. \tag{6.2}$$

However, because we computationally solve (6.1) on a Cartesian grid, our application is truly two–dimensional. We further emphasize the two–dimensionality of the numerical solution by discretizing the computational domain so that $\Delta x \neq \Delta y$.

Since the solution (6.2) is smooth for all time, we apply our scheme for one period in time. We assume that our scheme is stable under the "conventional" explicit time–step restriction, which in the scalar case is given by

$$\Delta t = \text{CFL} \frac{\Delta x \, \Delta y}{\max_u \left(|f'(u)| \, \Delta y + |g'(u)| \, \Delta x \right)},$$

where, for our purpose, "CFL" is a positive number bounded above by the stability limit of the chosen time discretization in (4.34). In this particular test case, we choose $\text{CFL} = 2/3$. The number of iterations required to reach $t = 2.0$ on a given grid is high enough to expect a significant accumulation of error.

We have measured solution errors on five consecutively refined meshes for the scheme (5.12) for the orders of accuracy $r = 1, 2, 3, 4$. Though good results were obtained in all four cases, we are particularly interested in the higher–order cases $r = 3$ and $r = 4$. These errors are presented in Table A, and are calculated with respect to the L_∞ and L_1 norms. We use r_c to denote the "computational order of accuracy." This value is calculated by assuming a linear accumulation of error as in (5.13), and is computed by

$$r_c = \frac{\ln \left(e_{h_1} / e_{h_2} \right)}{\ln \left(h_1 / h_2 \right)},$$

120

TABLE A

Solution Error for IVP (6.1)

$$t = 2.0 \qquad \text{CFL} = \tfrac{2}{3}$$

L_∞ ERROR

Grid	$r = 3$	r_c	$r = 4$	r_c
8 × 12	1.559 E-1		8.764 E-2	
16 × 24	2.331 E-2	2.74	1.021 E-2	3.10
32 × 48	3.002 E-3	2.97	1.239 E-3	3.04
64 × 96	3.749 E-4	3.00	1.390 E-4	3.16
128 × 192	4.666 E-5	3.01	1.544 E-5	3.17

L_1 ERROR

Grid	$r = 3$	r_c	$r = 4$	r_c
8 × 12	3.890 E-1		2.078 E-1	
16 × 24	5.678 E-2	2.78	1.605 E-2	3.69
32 × 48	7.627 E-3	2.97	1.764 E-3	3.20
64 × 96	9.130 E-4	2.99	1.237 E-4	3.82
128 × 192	1.142 E-4	3.00	8.658 E-6	3.84

121

where e_h is the solution error measured on a grid of mesh–width h. The exact and computational solutions are compared by cell–centered pointwise output. We see "one less order" of accuracy in the L_∞ norm for $r = 4$. We therefore expect that there are points in the solution where the reconstruction stencil is discontinuous, as referred to in Section 5.3. However, the third–order error for this particular problem is uniform.

6.2 Burgers Equation

We now test a nonlinear equation, namely the two–dimensional Burgers equation

$$u_t + (\tfrac{1}{2}u^2)_x + (\tfrac{1}{2}u^2)_y = 0 , \quad t > 0 , \tag{6.3a}$$

again with the initial data

$$u(x,y,0) = \tfrac{1}{2}\cos\pi(x+y) + \tfrac{1}{2} . \tag{6.3b}$$

As in the previous example, we solve the IVP (6.3) on $-1 \leq x,y \leq 1$ and again apply periodic boundary conditions. In this case, due to the non–linearity of equation (6.3a), gradients immediately begin to steepen for $t > 0$, until a shock eventually forms at time $t = 1/\pi$. We apply the scheme using a CFL number of $3/4$, up to $t = 0.15$, when the solution remains smooth. Table B illustrates the accumulated errors for this test case for $r = 3$ and $r = 4$. The exact solution is computed by using Newton–Raphson iterations to solve the characteristic relation

$$u(x,y,t) = \tfrac{1}{2}\cos\pi(x+y-2ut) + \tfrac{1}{2} . \tag{6.4}$$

In this test case, we do notice the "drop" in the order of accuracy, with respect to the L_∞ norm, in the third–order case as well as for $r = 4$. The solution was then computed to and past the point of shock formation, with no visible oscillatory behavior near the discontinuity.

122

TABLE B

Solution Error for IVP (6.3)

$$t = 2.0 \qquad \text{CFL} = \tfrac{3}{4}$$

L_∞ ERROR

Grid	$r = 3$	r_c	$r = 4$	r_c
8 × 12	3.765 E-2		2.632 E-2	
16 × 24	9.549 E-3	1.98	4.373 E-3	2.59
32 × 48	2.111 E-3	2.18	4.192 E-4	3.38
64 × 96	4.264 E-4	2.31	4.374 E-5	3.26
128 × 192	8.988 E-5	2.25	4.361 E-6	3.33

L_1 ERROR

Grid	$r = 3$	r_c	$r = 4$	r_c
8 × 12	4.938 E-2		3.010 E-2	
16 × 24	8.661 E-3	2.51	3.834 E-3	2.97
32 × 48	1.237 E-3	2.81	3.245 E-4	3.56
64 × 96	1.844 E-4	2.75	2.497 E-5	3.70
128 × 192	2.783 E-5	2.73	1.968 E-6	3.67

123

6.3 Mach 3 Stepped Inlet

Our next numerical experiment involves the solution of the two–dimensional Euler equations of gas dynamics (5.29) in a Cartesian frame of reference. We take our test case from [56].

A simple two–dimensional inlet with a step is installed onto a uniform Cartesian grid, and the problem begins with a uniform Mach 3 flow directed towards the step, from left to right. We assume the height of the inlet's entrance to be one length unit. The inlet is then three units long, and the step is located 3/5 unit from the entrance and is 2/5 unit high. With the physical time adjusted to this length scale, the solution reaches a steady state at approximately $t = 12.0$. However, the structure of this steady state is relatively uninteresting, and therefore we compute the solution in a time accurate manner up to $t = 4.0$, when the flow–field attains a complicated shock structure. A third–order time progression is shown in Figures 16a–f on a 120×40 grid, using 30 equally spaced density contours.

At the problem's outset, we assume that the inlet is filled with air which we model as an ideal gas, with $\gamma = 1.4$, with normalized initial free–stream conditions

$$\rho_\infty = 1.4 \,, \quad P_\infty = 1.0 \,, \quad u_\infty = 3.0 \,, \quad v_\infty = 0 \,. \tag{6.5}$$

The supersonic inflow boundary condition is specified by (6.5) and held fixed. Because the outflow is supersonic, the exit boundary condition has no effect on the flow, and therefore we simply assume all gradients to vanish at this boundary. At the walls, we apply the tangency condition

$$\vec{V} \cdot \hat{n}_w = 0 \,, \tag{6.6}$$

where $\hat{n}_w$ is the unit vector normal to a given wall. The nature of this solution is such that the corner of the step is the center of a rarefaction fan, and hence is a singular point of the flow. We therefore apply a special treatment at this
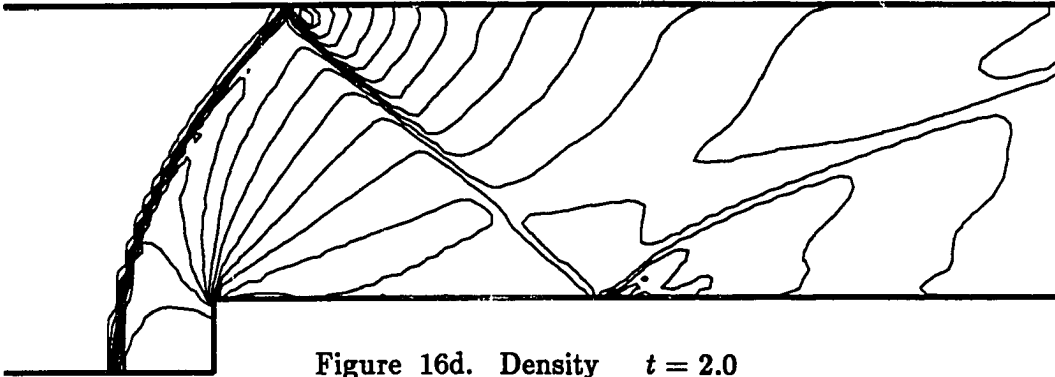
124

MACH 3 STEPPED INLET

(Third-Order ENO)

Figure 16a. Density $t = 0.5$

Figure 16b. Density $t = 1.0$

Figure 16c. Density $t = 1.5$

125

MACH 3 STEPPED INLET

(Third–Order ENO)

Figure 16d. Density $t = 2.0$

Figure 16e. Density $t = 3.0$

Figure 16f. Density $t = 4.0$

126
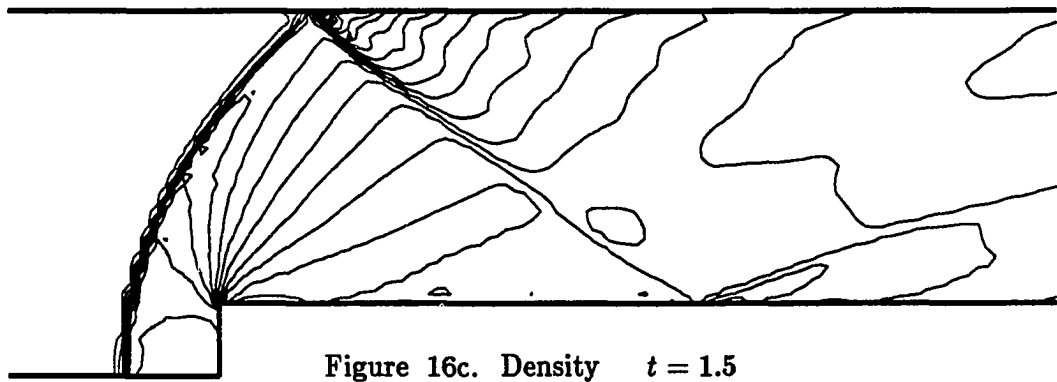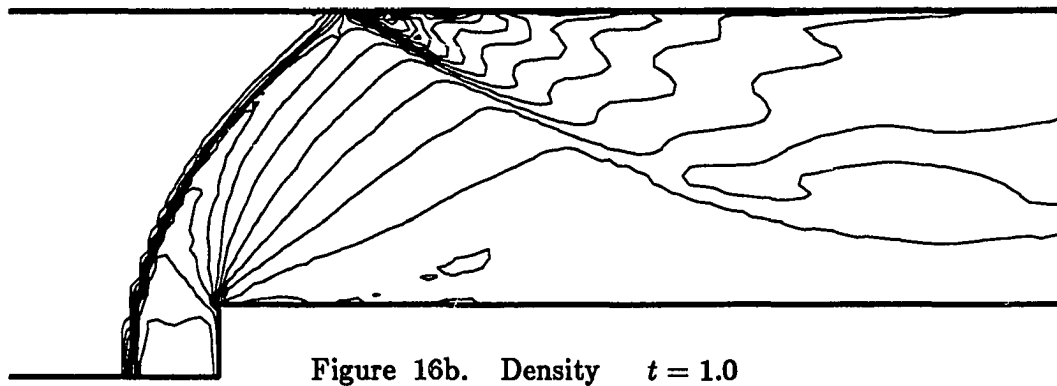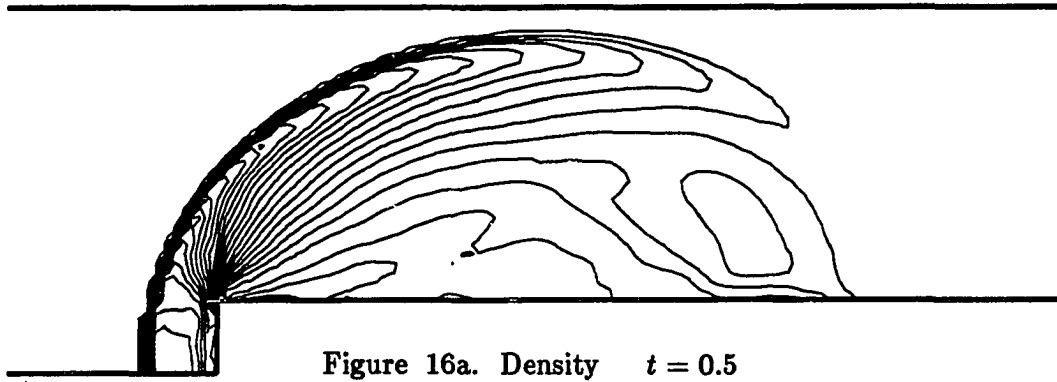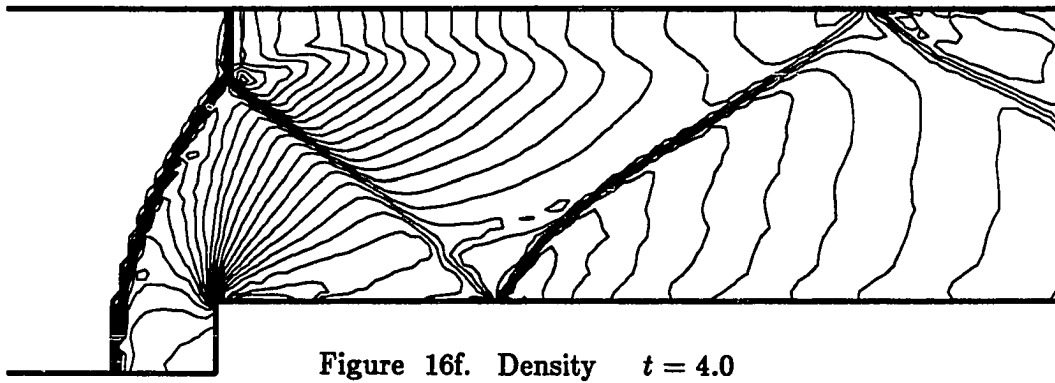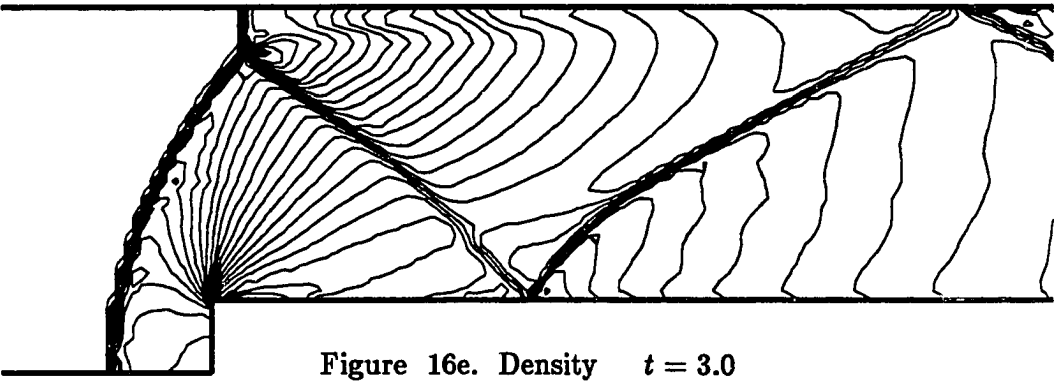
corner, as described in [56] in order to avoid large numerical errors generated in the neighborhood of this point which would hinder our qualitative comparison.

Since we approximate the required "Riemann fluxes" by Roe's method, the nature of this particular problem requires that we modify these fluxes with an entropy correction. We refer here to the expansion fan centered at the corner of the step. As the sonic line within this nearly steady rarefaction is virtually aligned with the computational mesh, the directional application of Roe's method will yield an "expansion shock" in this area of the flow. This unphysical solution arises from the fact that, in the x–direction, the eigenvalue $\lambda^1(U) = u - c$ passes from negative to positive values near the sonic line (See Section 3.4). It is suggested in the literature that when

$$\lambda^k(U_L) < 0 < \lambda^k(U_R) \ , \tag{6.7}$$

we require that the magnitude $|\lambda^k(\hat{U})|$ of the average eigenvalue be no less than some specified distance from zero (See e.g. [22,16,24]). In our calculation, we will use an entropy fix suggested by Harten [21]. As it applies to the general "Roe flux"

$$F^{ROE}(U_L, U_R) = \frac{1}{2} \left[F(U_L) + F(U_R) \right] - \frac{1}{2} \sum_{k=1}^{m} \hat{\delta}^k |\hat{\lambda}^k| \hat{r}^k \ , \tag{6.8}$$

we replace any $|\hat{\lambda}^k|$ in this expression which might locally satisfy (6.7) by a value $\bar{\alpha}^k$ which is dependent upon the difference in the left and right characteristic speeds, in particular,

$$\bar{\alpha}^k = \frac{(\lambda^k(U_R) - \hat{\lambda}^k) |\lambda^k(U_L)| + (\hat{\lambda}^k - \lambda^k(U_L)) |\lambda^k(U_R)|}{\lambda^k(U_R) - \lambda^k(U_L)} \ . \tag{6.9a}$$

It was found that this correction does an excellent job of accurately accounting for the spread of the expansion around the step's corner. It was also found, however, that another slight modification had to be made, due to the ability of our scheme to capture shocks so narrowly. The numerical instabilities associated with strong, stationary shocks are well known (See [56]). In our case, by the time $t = 3.0$, the

127

Mach stem at the top wall as well as the base of the bow shock are moving very slowly (Figures 16e–f). Since the shocks in these areas are also very narrow and nearly aligned with the mesh, it was found that numerical instabilities would arise, most notably an unphysical kink in the Mach stem. We therefore sought to increase the scheme's dissipation in these areas in order to slightly broaden such shocks. Our Roe flux (6.8) is then modified for this calculation by setting

$$|\hat{\lambda}^1| = \max(\bar{\alpha}^1, \varepsilon) \tag{6.9b}$$

where $\bar{\alpha}^k$ is (6.7) and ε is a small parameter.

We implement the scheme as detailed in Section 5.7 with the modifications mentioned above, using a CFL number of 0.8 on a 120 × 40 grid, and $\varepsilon = 0.1$ in (6.9b). We present second– and fourth–order accurate solutions at $t = 4.0$ in Figures 17a–d, choosing density and Mach number as the variables of comparison. Both variables are plotted using thirty equally spaced contours. Sharper discontinuities are the most notable improvement in the fourth–order case, particularly the slip line emanating from the triple point near the top wall. The Mach stem in this area is also more correct in its length and its position upstream. The weak shock from the corner of the step is also more pronounced in the fourth–order case, as is the other weaker slip line formed as this shock intersects the shock reflecting from the top of the step. This weaker slip line is virtually undetectable in the the second–order accurate solution on this computational mesh.

Because there is a tendency for less dissipative schemes to exhibit more numerical noise in the presence of strong stationary shocks on fine grids, we have tested our fourth–order scheme on a finer grid as well. Indeed, the instabilities arising from the narrow capturing of nearly steady shocks aligned with the grid were found to be more prominent on a finer computational mesh. Figures 18a–d display the results at $t = 4.0$ of a fourth–order solution on a 240 × 80 grid, with 50 contours. The only aspect of our numerical application which we have changed for this finer mesh
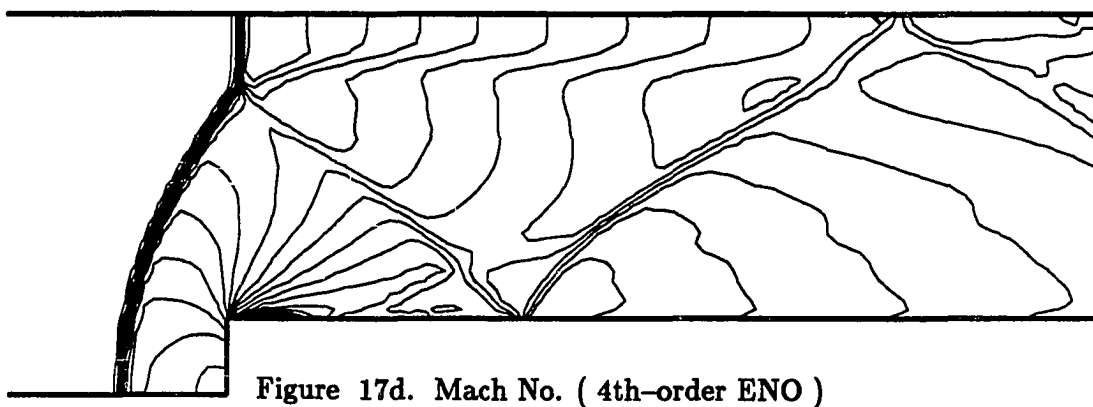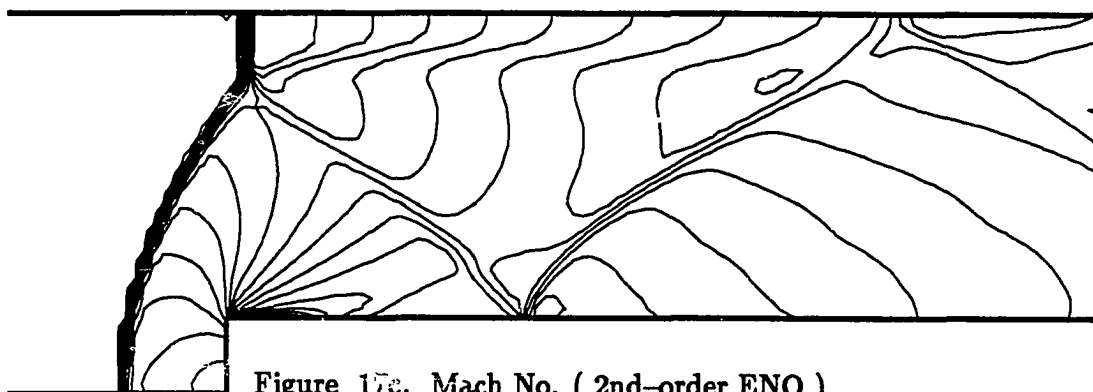
128

STEPPED INLET

$$M_\infty = 3.0$$

Figure 17a. Density (2nd-order ENO)

Figure 17b. Density (4th-order ENO)

129

STEPPED INLET

$$M_\infty = 3.0$$

Figure 17c. Mach No. (2nd-order ENO)
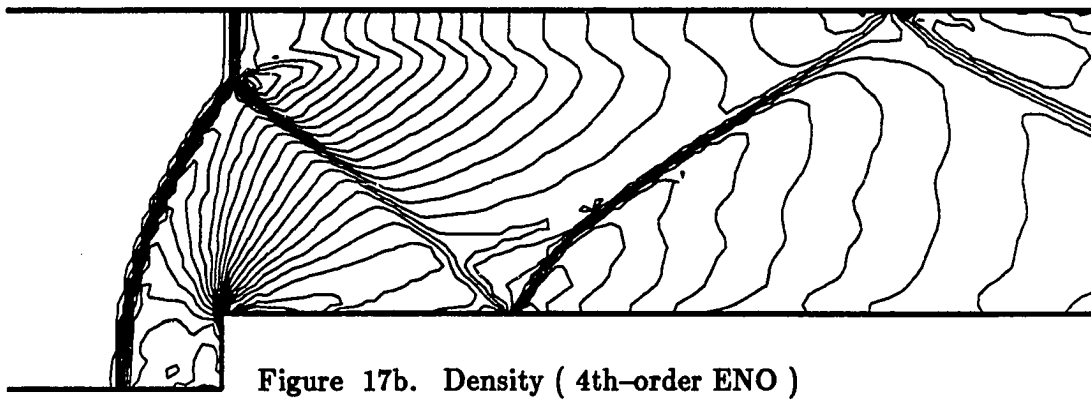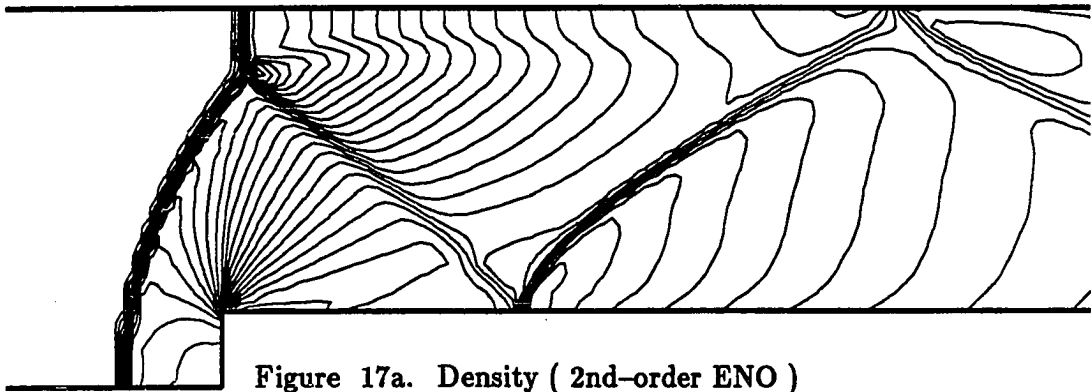
Figure 17d. Mach No. (4th-order ENO)
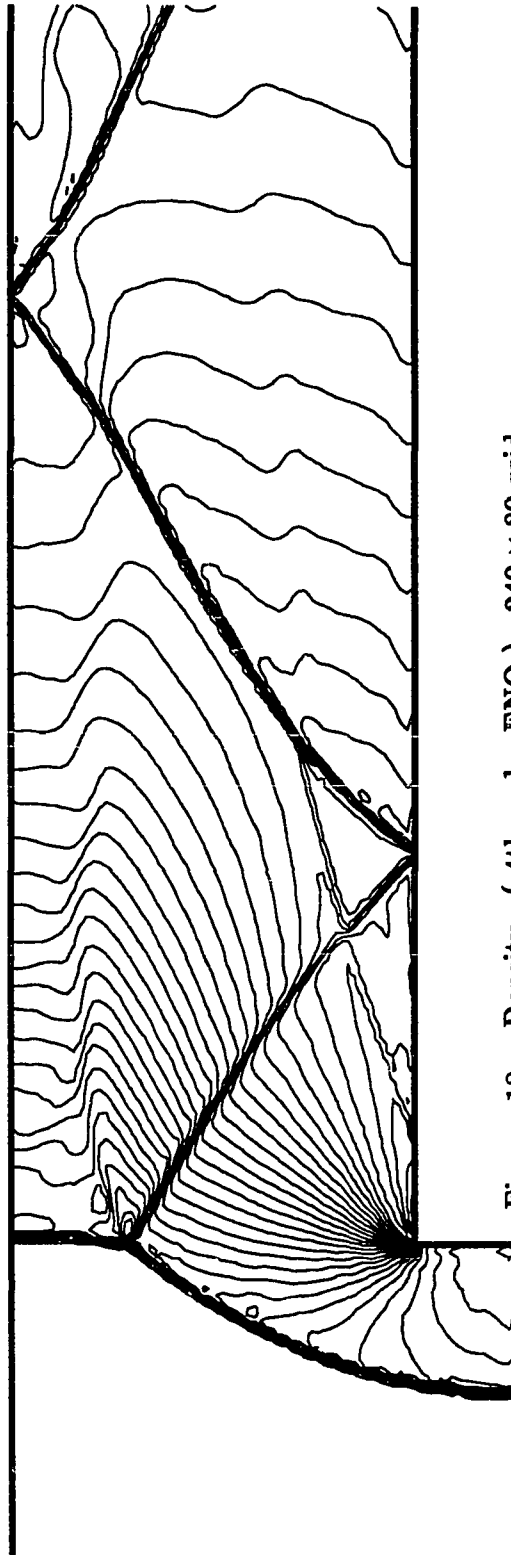
130

STEPPED INLET

$M_\infty = 3.0$

Figure 18a. Density (4th-order ENO)   240 × 80 grid
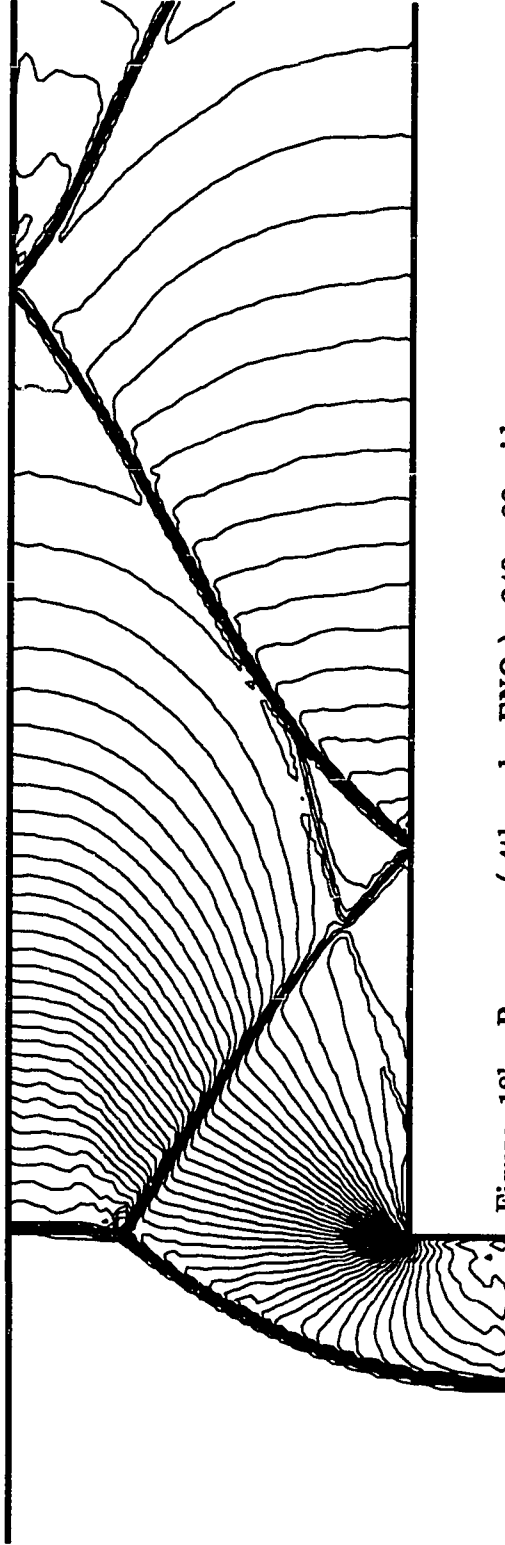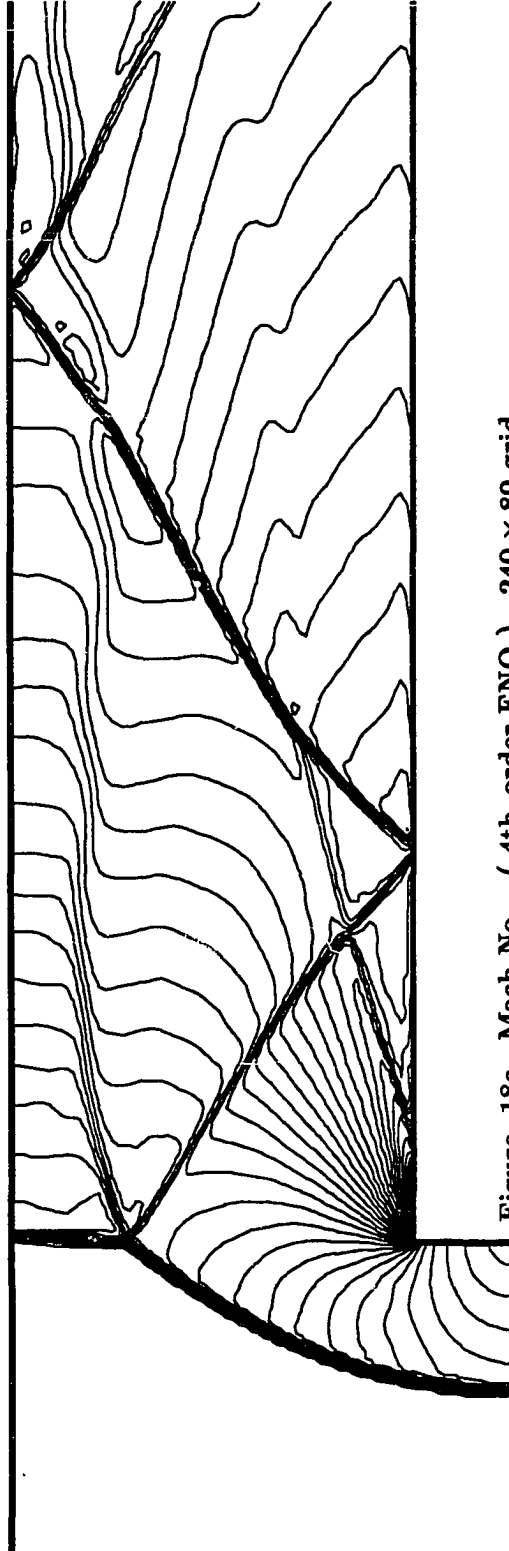
131

STEPPED INLET

$M_\infty = 3.0$

Figure 18b. Pressure (4th-order ENO)   240 × 80 grid

132

STEPPED INLET

$M_\infty = 3.0$

Figure 18c. Mach No. (4th-order ENO)   240 × 80 grid

133

STEPPED INLET

$M_\infty = 3.0$
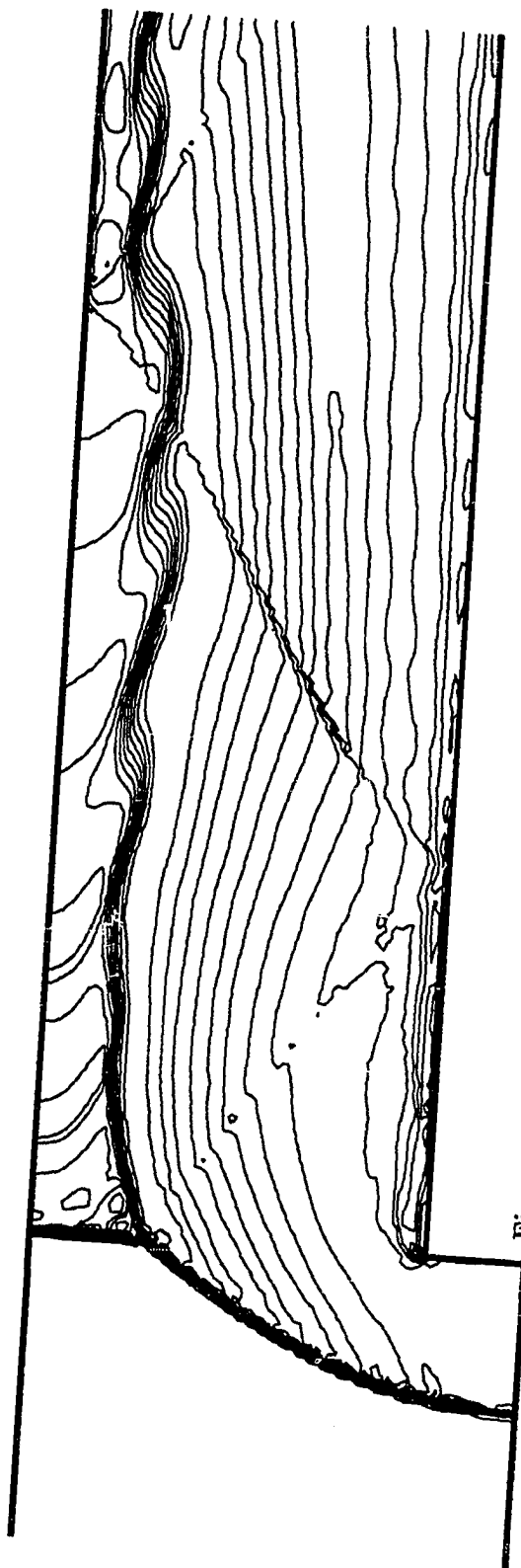
Figure 18d. Entropy (4th–order ENO) 240 × 80 grid

134

is to set $\epsilon = 0.2$ in (6.9b). The quantity "Entropy" which is plotted in Figure 18d is the entropy–like quantity $\ln(P/\rho^\gamma)$. Even with the added dissipation, it is this plot which displays the presence of the slight kink in the Mach stem. As the entropy generated by this instability propagates downstream, it reveals itself in the significant amount of "flag–waving" present in the slip line.

6.4 Mach 3 Shock Wave Over an Obstacle

We consider now an extension of the Euler equations (6.4) to a curvilinear co–ordinate system. For this purpose, let

$$x = \xi, \qquad\qquad 0 \le \eta \le H,$$

$$y = \begin{cases} \eta, & 0 \le \xi < a \\ \eta + A(\eta)[\cos(B\xi + C) + 1], & a \le \xi \le b \\ \eta, & b < \xi \le L, \end{cases} \qquad (6.10\text{a})$$

represent the relationship between a point (x, y) in physical space and a point (ξ, η) in the rectangle $[0, L] \times [0, H]$. Now, if

$$a = \frac{L}{4}, \qquad b = \frac{3L}{4}, \qquad B = \frac{4\pi}{L}, \qquad C = -2\pi, \qquad (6.10\text{b})$$

$$L = \frac{3H}{2}, \qquad A(\eta) = \frac{3}{16}(H - \eta), \qquad (6.10\text{c})$$

then by the transformation (6.10), we can generate a curvilinear mesh whose scale is exhibited in Figure 19a.

We now assume the curve determined by $\eta = 0$ to be a solid surface, and consider the motion of a strong shock toward the obstacle represented by the curved portion of this surface (Figure 19b). We wish to calculate the flow–field as determined by the solution of the Euler equations, which we solve in curvilinear co–ordinate form (See Section 5.6). For this experiment, all of the grid quantities necessary for the curvilinear flux computation (5.25) are evaluated exactly from the transformation (6.10).
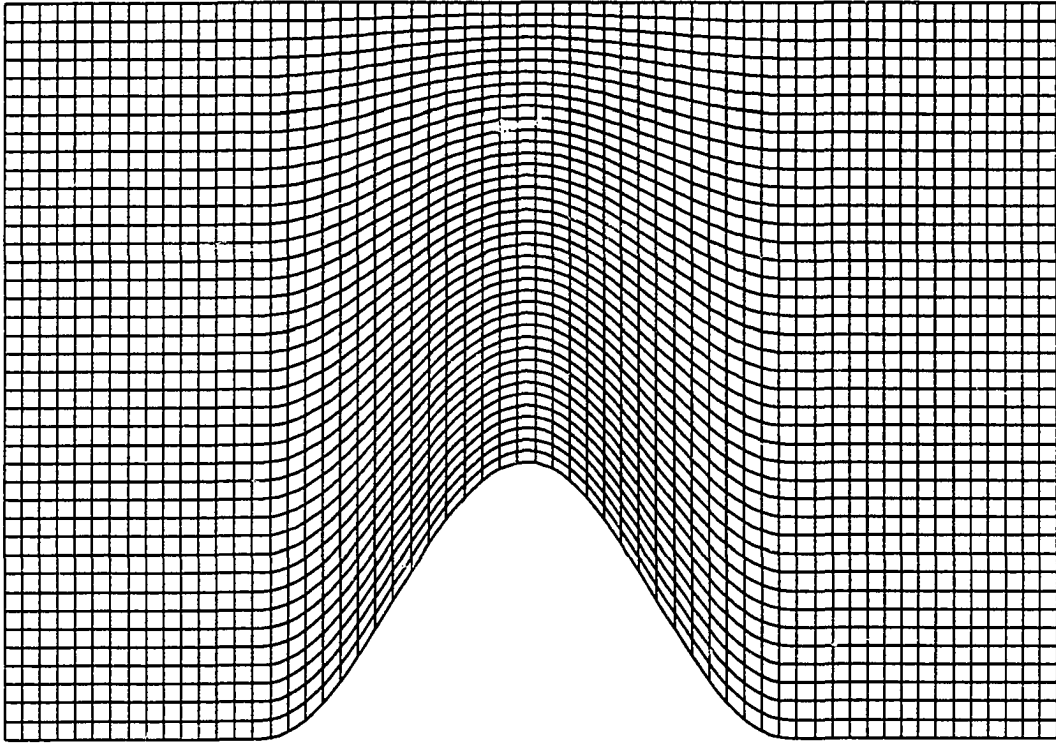
135

Figure 19a. Curvilinear mesh generated by the transformation (6.10).
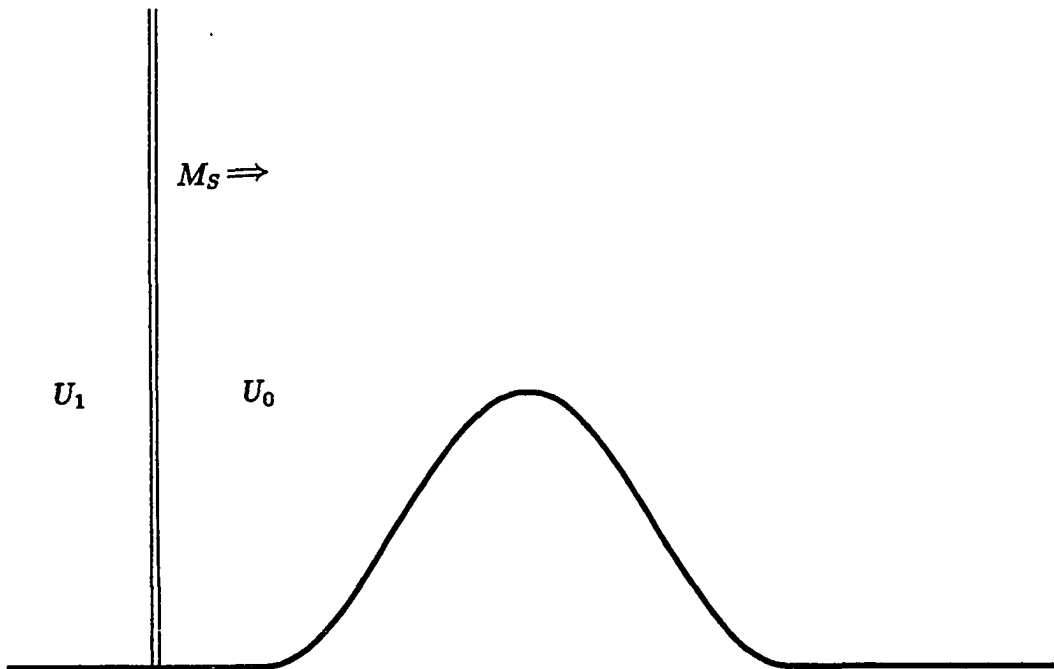

Figure 19b. Initial conditions for a shock wave over an obstacle.

136

Our initial conditions are the two constant states U_0 and U_1 which determine a normal shock whose shock Mach number is $M_S = 3.0$, in air. This shock is initially positioned along $x = x_d$, between 0 and $L/4$, as in Figure 19b. We normalize these conditions with respect to the still-air initial state U_0. These conditions are

$$\rho_0 = 1.0\,, \qquad P_0 = 1.0\,, \qquad u_0 = v_0 = 0\,. \tag{6.11}$$

Using these conditions and the shock Mach number M_S, we can then determine the state U_1 by the the Rankine–Hugoniot jump condition.

We note here that our mesh is not orthogonal, thereby providing a more general test of the curvilinear extension of our scheme. Furthermore, due to the significant amount of curvature posed by the geometry and the strength of the shock wave, this test is also quite demanding. Though this continuous curvature inhibits us from being able to predict the solution *a priori*, we can nonetheless expect the flow–field to exhibit certain characteristics as the shock travels over this body. For instance, there will be a compression near the base of the obstacle as the shock strikes, followed by a subsequent reflection. As the incident shock then passes over the top of the body, the fluid will expand, thereby weaking the incident shock and causing it to curve. By Crocco's theorem (*e.g.* [37]), once we have curvature in a shock, we can expect vorticity to be induced into an otherwise irrotational flow–field.

Provided we ensure that the entire reflection remains within the limits of the grid, the boundary conditions are relatively simple. The initial conditions U_0, U_1 are held constant on the left and right boundaries, respectively. The linear degeneration in η of the grid's curvature to a straight line at $\eta = H$ makes the far–field condition a simple matter of setting normal gradients to zero. On the solid surface, the tangency condition is imposed. We apply our scheme in its curvilinear version for 350 time steps using a CFL of 0.8 on a 120×80 grid. Figures 20a–d depict the time progression of our solution with regard to density, while Figures 20e–f then show the pressure and velocity fields of the final calculation. Here we display 30 equally

137

MACH 3 SHOCK WAVE OVER AN OBSTACLE
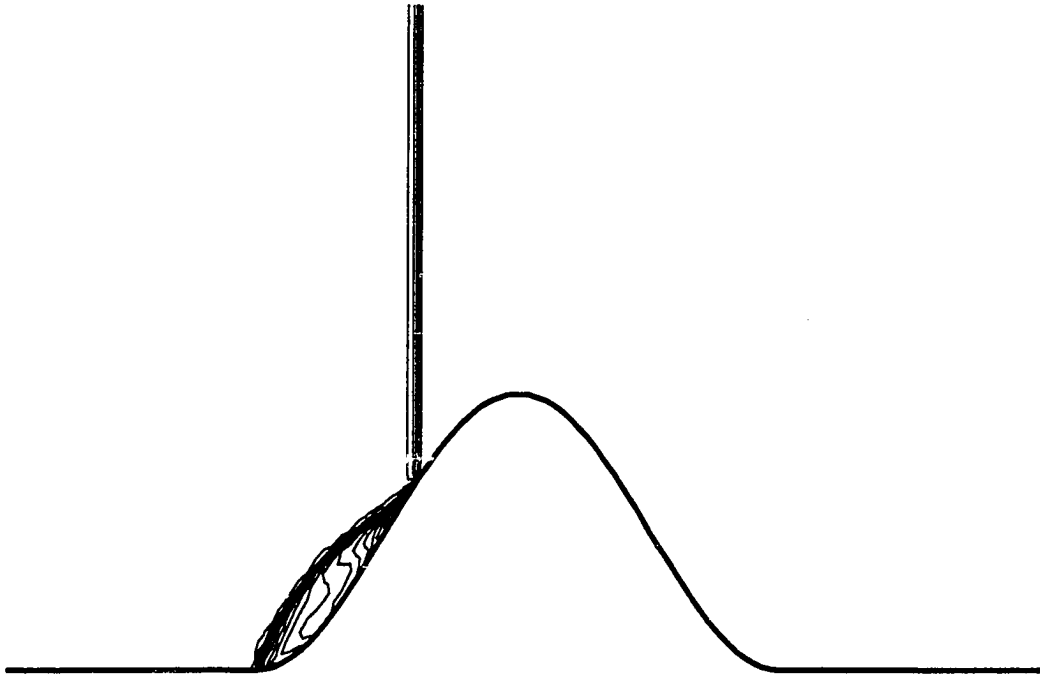
(Third-Order ENO)
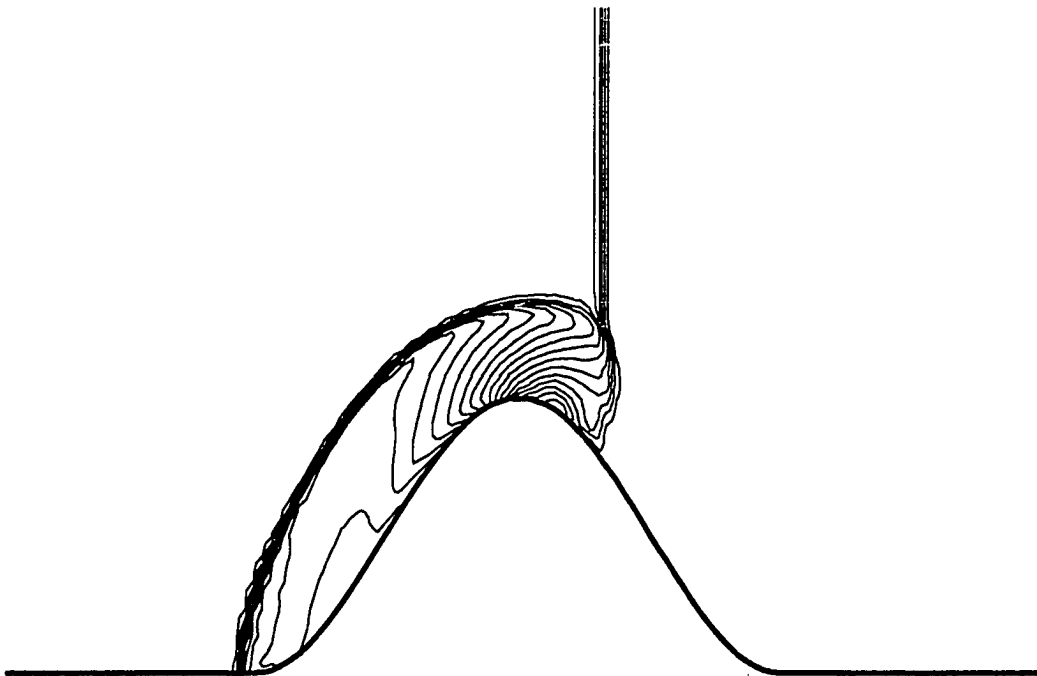

Figure 20a.   Density   100 Δt

Figure 20b.   Density   200 Δt

138

MACH 3 SHOCK WAVE OVER AN OBSTACLE
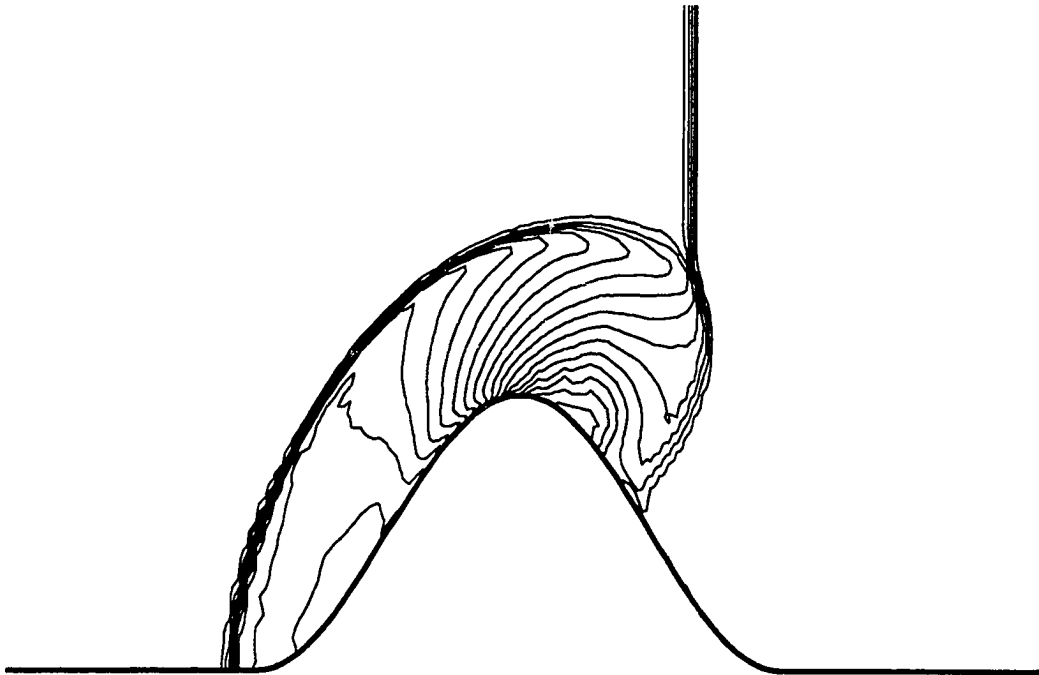
(Third–Order ENO)

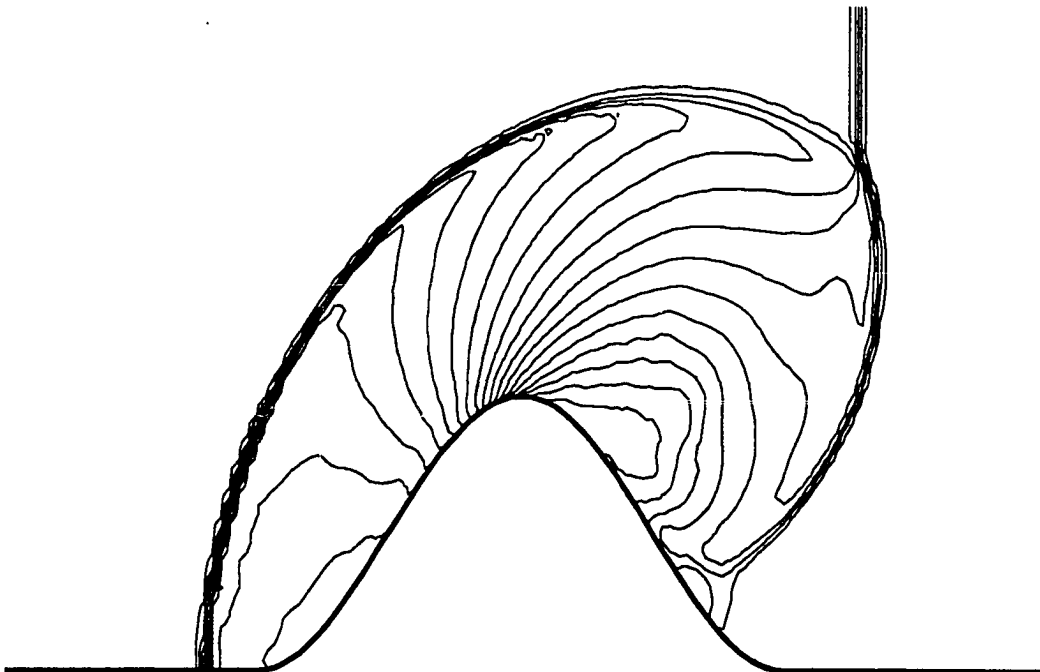Figure 20c. Density   250 Δt

Figure 20d. Density   350 Δt

139

MACH 3 SHOCK WAVE OVER AN OBSTACLE
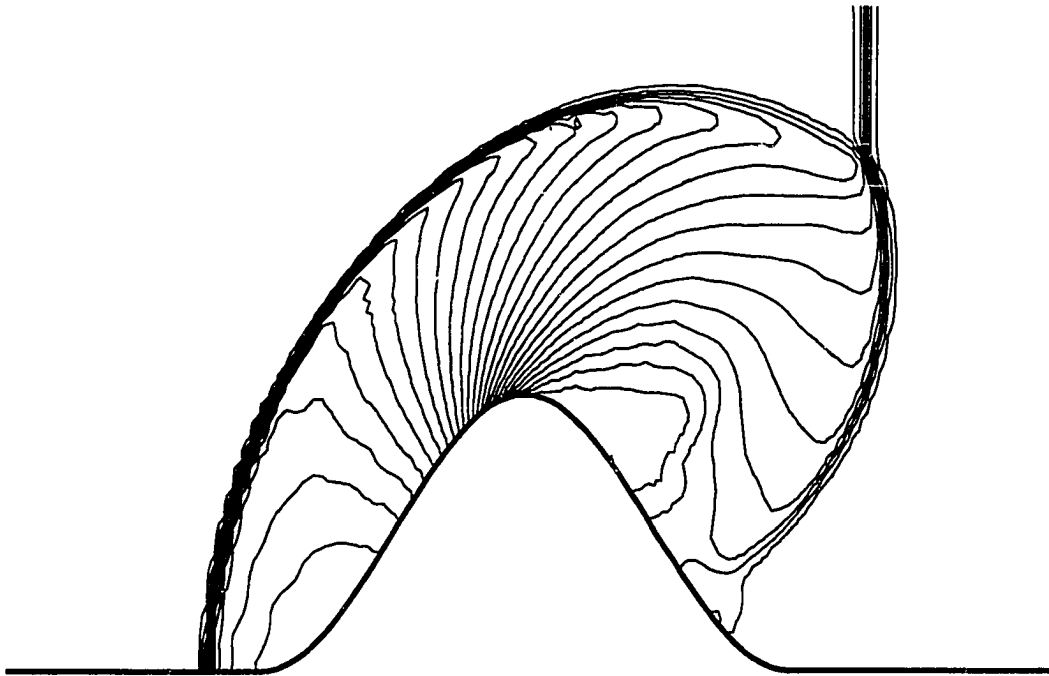
(Third-Order ENO)
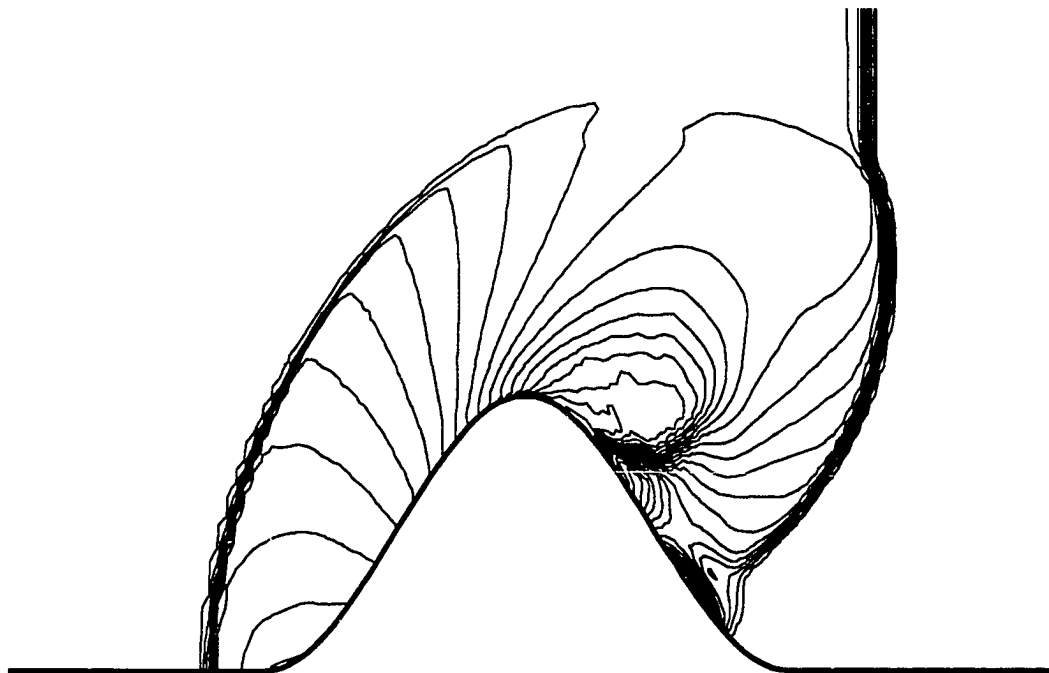
Figure 20e.   Pressure   350 Δt

Figure 20f.   Velocity magnitude   350 Δt

140

spaced contours for all six plots. The curvature of the incident shock as well as evidence of the presence of vorticity are apparent. On the downwind side of the obstacle, the flow becomes quite complex. We notice what appears to be another shock and a quantitative inspection in this flow area reveals a region of separated flow.

In order to "test" the validity of this coarse grid computation we performed the identical experiment on a 240 × 160 grid, the results of which are in Figures 21a–c. The flow variables of density, pressure, and velocity magnitude are displayed after 750 time steps, with 75 equally spaced contours. The smaller shock on the downwind side of the body is now apparently a detached shock, and the velocity vector plot in Figure 21–d clearly shows the flow separation.

6.5 Mach Reflections

Again, we solve the two–dimensional Euler equations of gas dynamics, this time as the solution pertains to the reflection of a moving shock wave from an inclined wall. The self–similar nature of such a solution lends itself to a rigorous analysis which is well documented in the literature (*e.g.* [27]), and we therefore omit any general discussion of this phenomenon. These oblique shock reflections have been the subject of extensive experimental and computational research and the interested reader is referred to [9,10,57] and the references therein.

Our problem begins with a plane shock, whose Mach number we denote by M_S, which is moving into still air toward a wall inclined by an angle θ_w to the direction of the shock's motion as shown in Figure 22a. The flow orientation is chosen to facilitate comparison with existing experimental interferograms. The problem becomes truly two–dimensional when the shock encounters the wall and forms a reflection whose structure can be quite complex. Analysis shows that the resulting similarity solution can be entirely determined by the values M_S and θ_w.
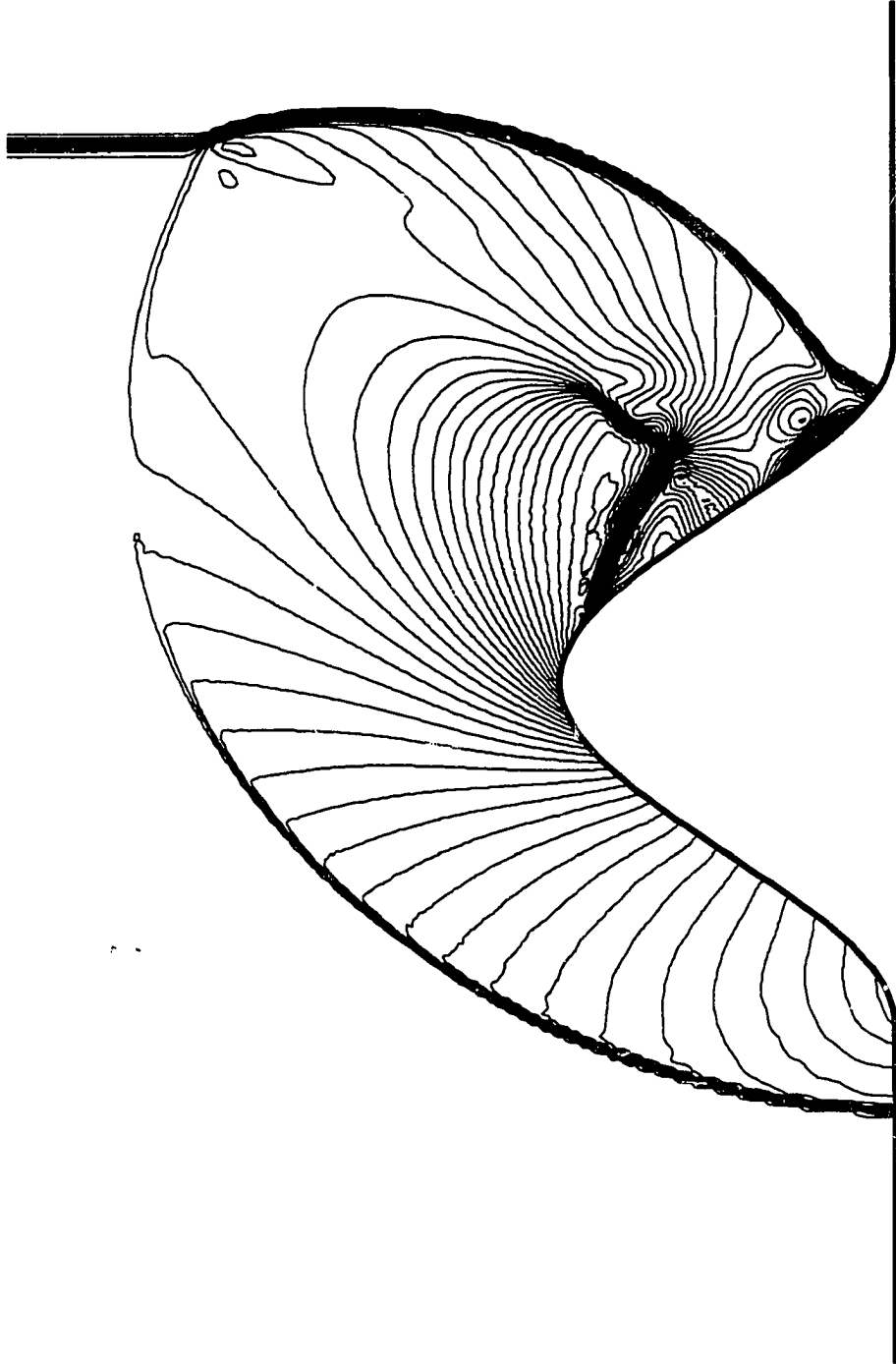
141
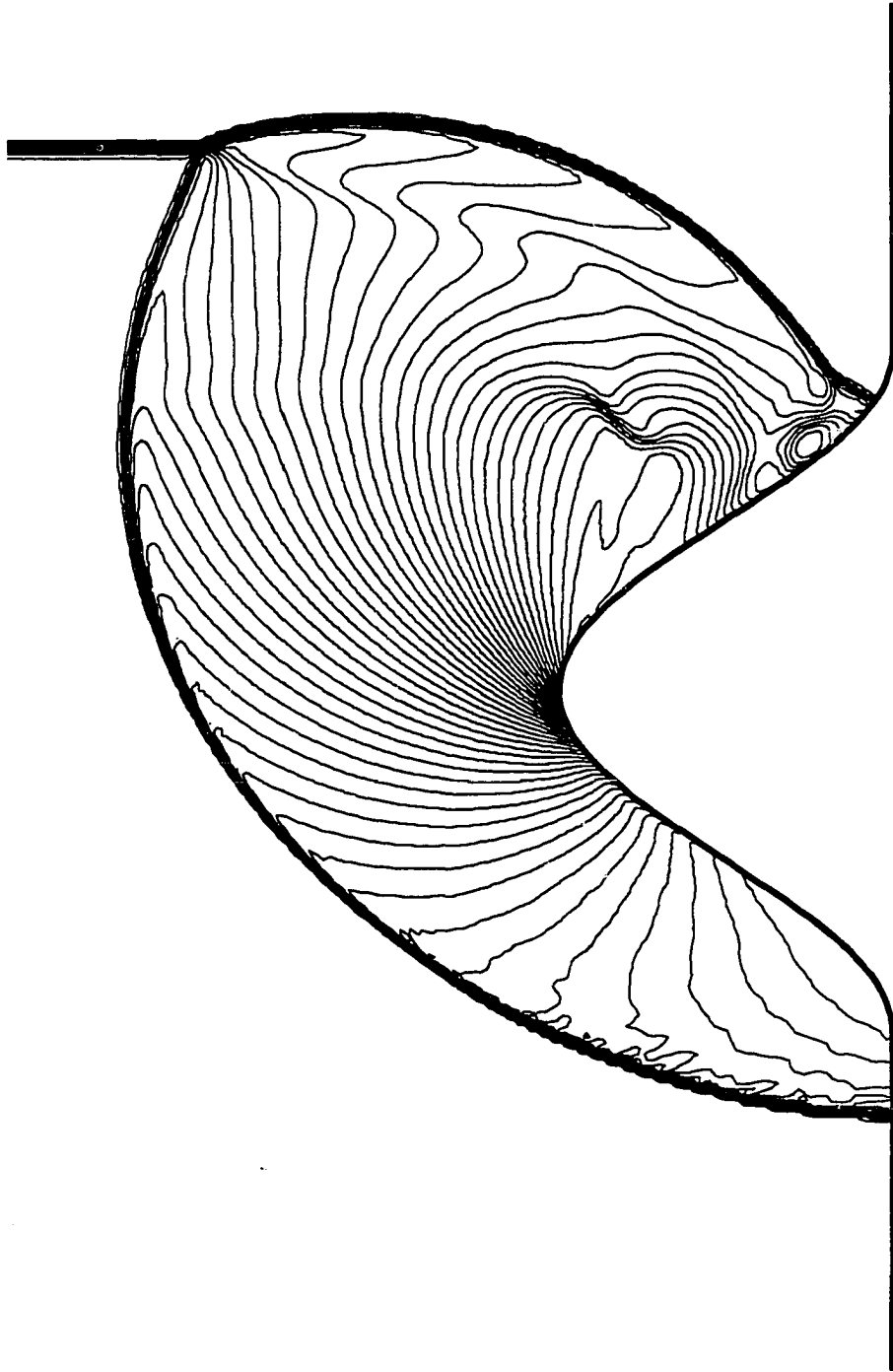
Figure 21a.   Density (3rd–order ENO)   240 × 160 grid

142

MACH 3 SHOCK WAVE OVER AN OBSTACLE
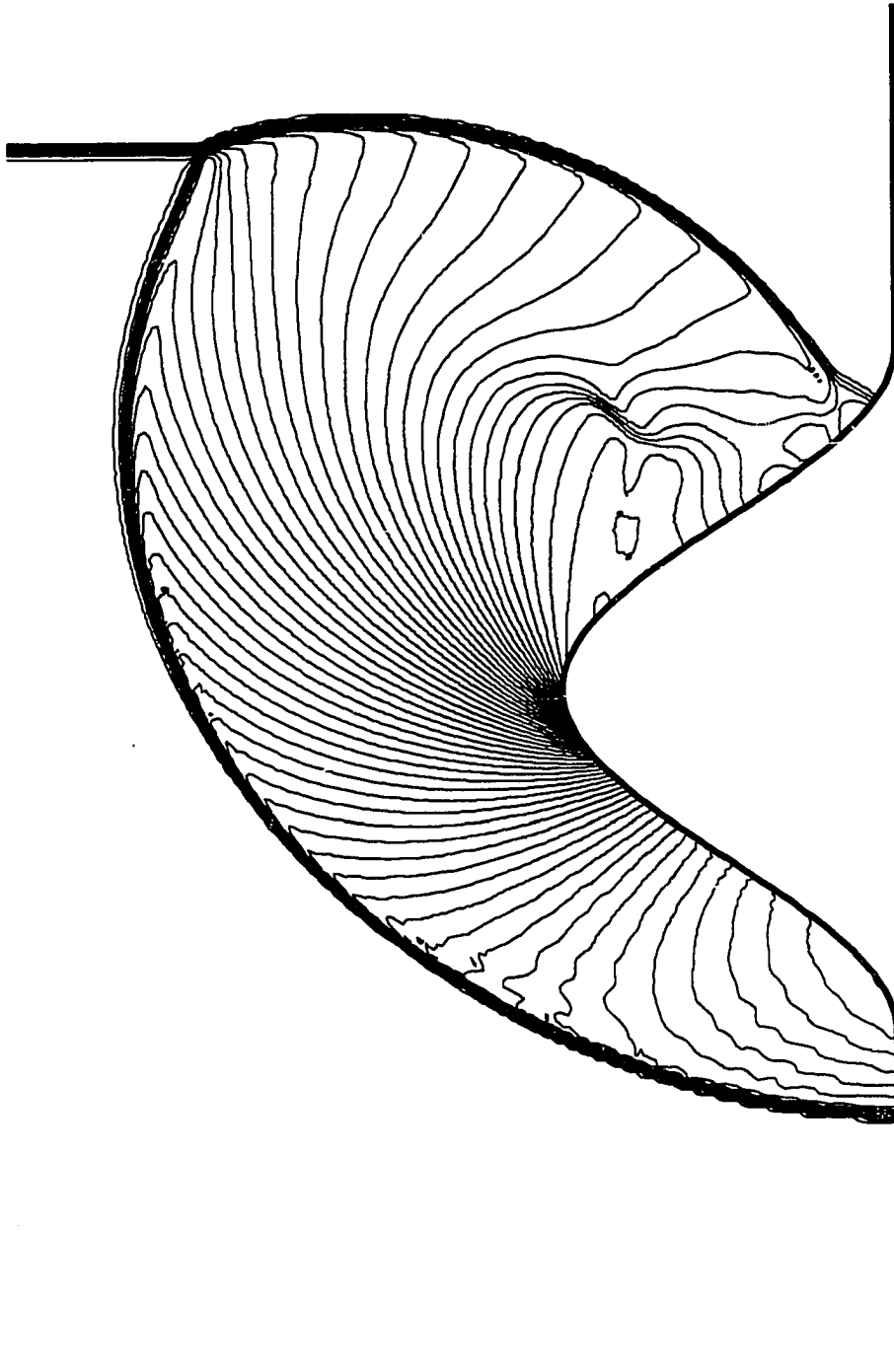
Figure 21b. Pressure (3rd-order ENO) 240 × 160 grid

143

MACH 3 SHOCK WAVE OVER AN OBSTACLE

Figure 21c.   Velocity magnitude (3rd-order ENO)   240 × 160 grid

144

MACH 3 SHOCK WAVE OVER AN OBSTACLE
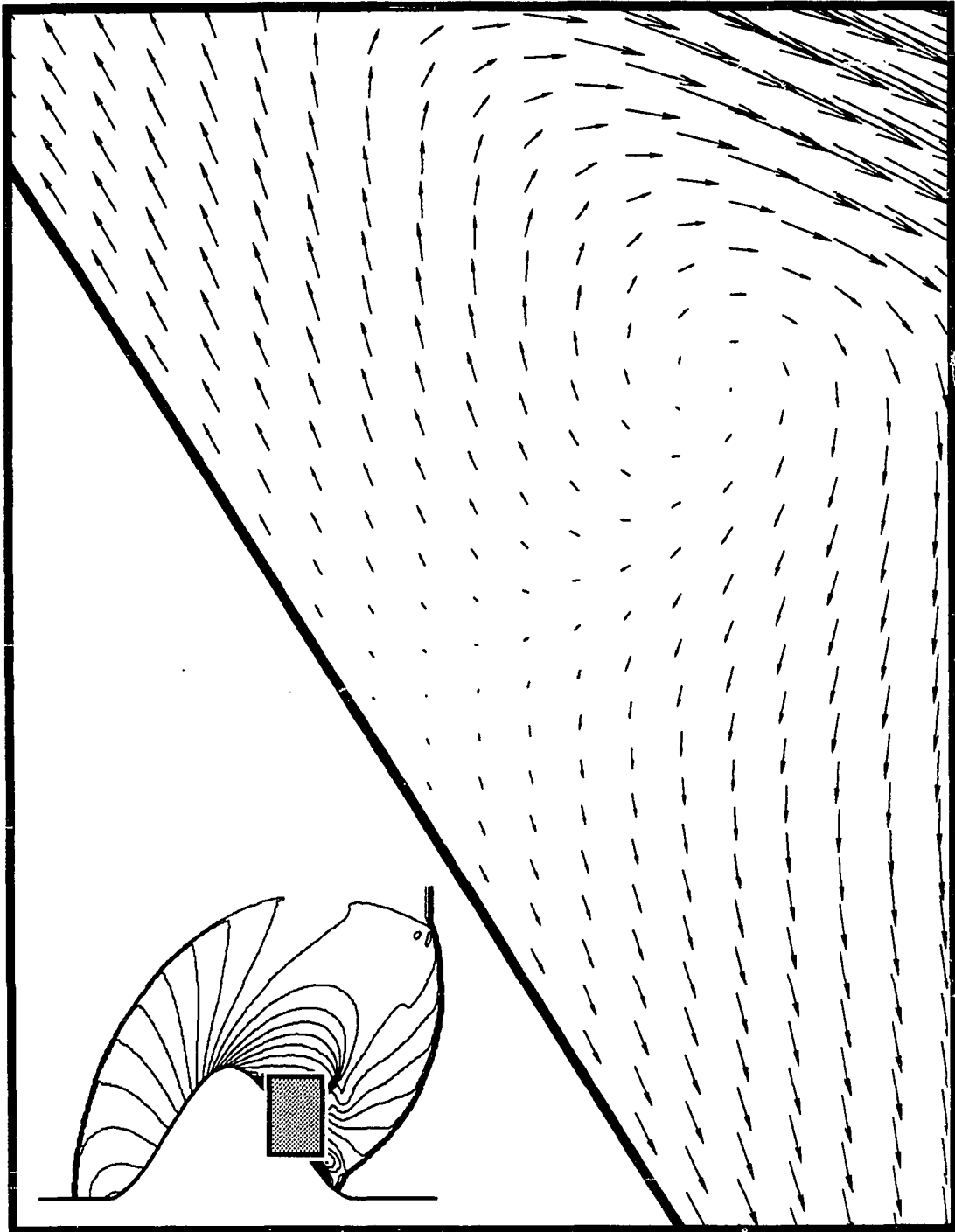
(Third-Order ENO)

Figure 21d.   Detail – Velocity vectors in region of flow separation (See insert).
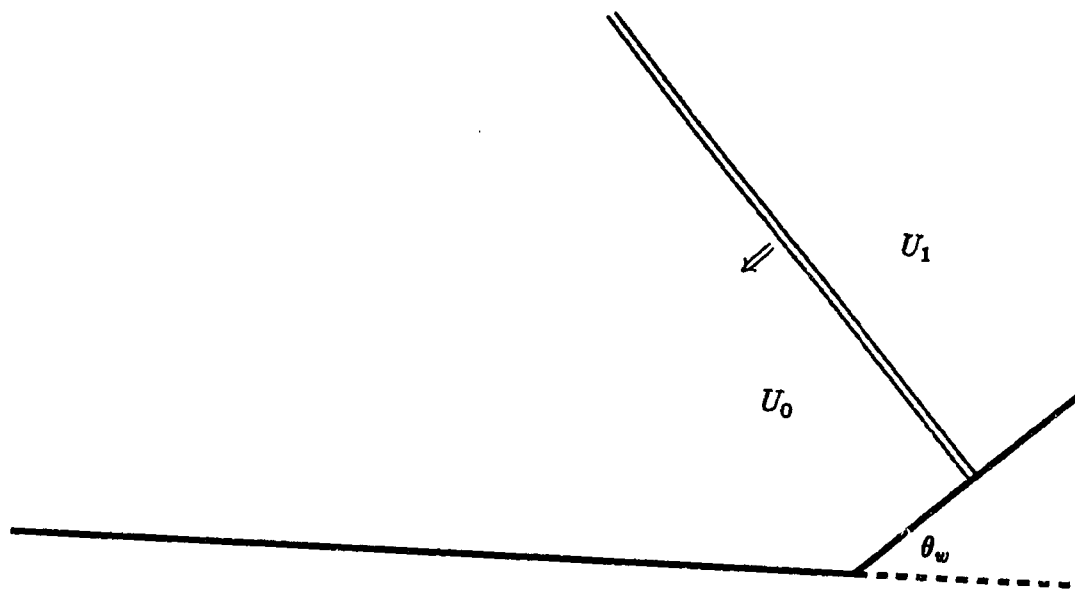
145

Figure 22a.   Initial conditions for the Mach reflection problem
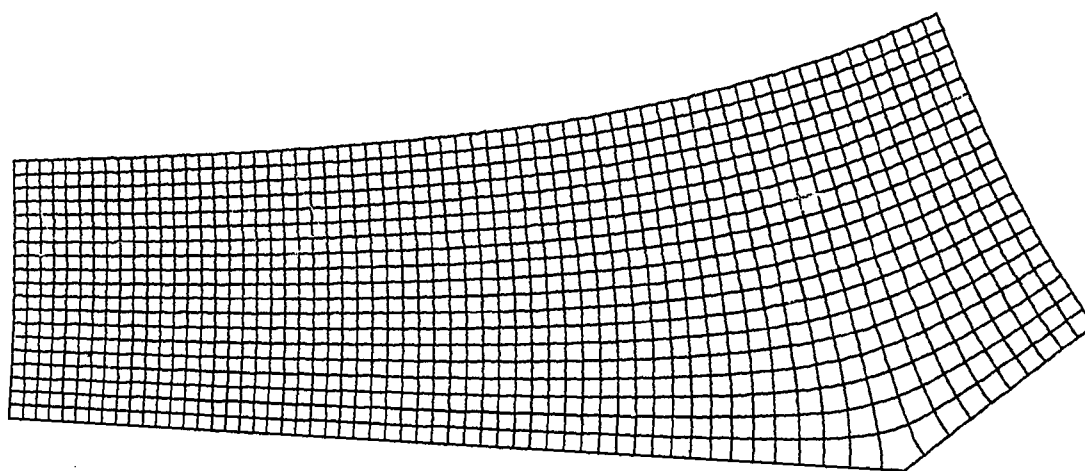

Figure 22b.   Computational grid for the Mach reflection problem

146

We examine two cases, both of which are double Mach reflections. The wall angle θ_w is 40 degrees in both cases, and the shock Mach numbers we examine are 2.87 and 3.72. This type of reflection exhibits a complex structure containing shock diffractions and slip lines, and is particularly demanding of any computational algorithm. For this reason, these computations are most commonly performed with the use of a self–similar transformation which effectively removes the time dependence of the solution. The resulting equations that are then solved resemble the steady Euler equations with the addition of source terms. However, because we wish to examine the temporal as well as the spatial accuracy of our scheme, we choose to compute our solutions in a time–accurate manner.

In addition to the demanding nature of the solution itself, such computations are also made difficult by geometric concerns, largely due to the presence of a sharp corner. Though there is no way to rid ourselves of the corner itself, we attempt to mitigate its presence by using a curvilinear grid transformation. A portion of our particular mesh is shown in Figure 22b, and is generated by a Schwarz–Christoffel transformation. We could use this transformation to derive all of the necessary grid quantities referred to in Section 5.5. However, we would like to test our scheme in its most general form. Therefore, given a collection of points generated by this transformation, we calculate all of our mesh variables from the approximate grid lines we generate by polynomial interpolation.

Our initial conditions are the two constant states U_0 and U_1 which determine the desired incident shock. We normalize these conditions with respect to the still–air initial state U_0. These conditions then are

$$\rho_0 = 1.0, \quad P_0 = 1.0, \quad u_0 = v_0 = 0. \tag{6.12}$$

Using these conditions and the shock Mach number M_S, we can then determine the state U_1 by means of the Rankine–Hugoniot jump condition. These initial states are then conservatively interpolated onto the computational mesh.

147

As in the preceding test case, the boundary conditions are relatively simple provided we ensure that the entire reflection remains within the limits of the grid. The initial conditions U_0, U_1 are applied on the left and right boundaries, while the only concern at the far-field is adequately accounting for the movement of the plane shock. Once again, the tangency condition is imposed on the wall. We apply our scheme in its curvilinear version to the two test cases for 400 time steps using a CFL of 0.8 on a 180×40 grid. Our purpose is not only to compare our numerical solutions by their order of accuracy, but also to compare each of them with experimental results, which we obtain from [11].

Figure 23a is an interferogram resulting from an experiment designed to photographically exhibit the density structure for the case $M_S = 2.87$. (The alphabetical labeling of this picture is not relevant to our discussion.) Density contour plots for the second- and fourth-order numerical solutions are compared in Figures 23b-c. Overall, crisper discontinuities are observed in the fourth-order solution. The shock structure itself is also more correct in Figure 23c, in that the more perpendicular orientation of the incident Mach stem with respect to the wall is more in line with the experimental observation. The "toeing out" of this Mach stem in Figure 23b appears to be due to the poorer resolution of the contact discontinuity emanating from the primary triple point.

In Figure 23d, we plot the fourth-order density solution on the wall in order to make a comparison with experimental measurements. The x-axis is scaled by the distance L from the incident Mach stem on the wall to the corner, the Mach stem location being $x = 0$, and the the corner at $x = 1$. Overall agreement with the experimental data is as good if not better than the similar comparison in [11] in which the numerical solution was achieved in a self-similar fashion and on a much finer grid.

$$M_S = 2.87 \qquad \Theta_w = 40°$$

Figure 23a. Experimental Isopycnics (Ref. [11])

Figure 23b. Density Contours (2nd-order ENO)

Figure 23c. Density Contours (4th-order ENO)

149

WALL DENSITY

$$M_S = 2.87 \qquad \Theta_w = 40°$$
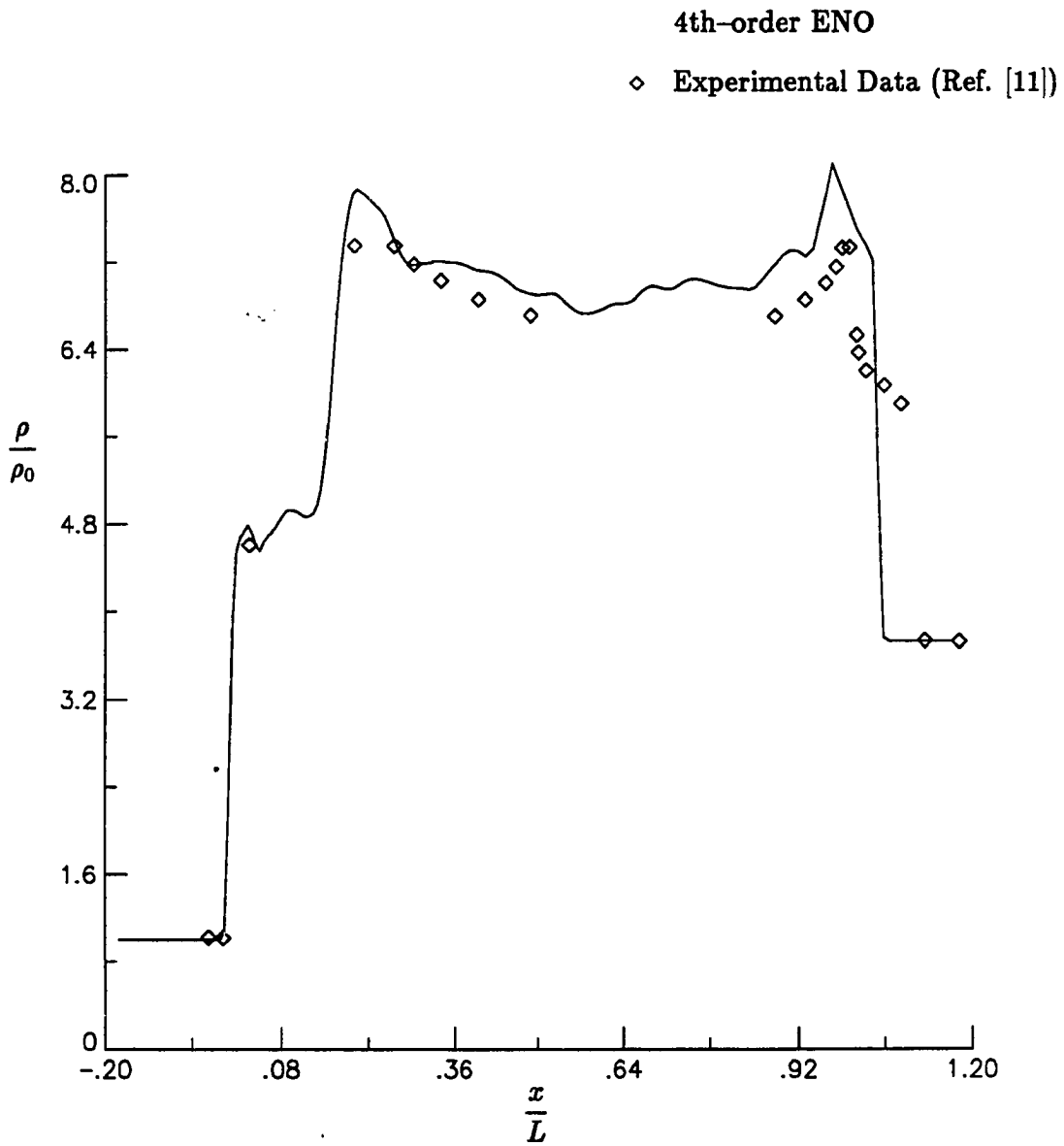
4th-order ENO

◇ Experimental Data (Ref. [11])

Figure 23d. Comparison of 4th-order ENO solution and experimental data

150

In Figures 24a–d, we have the analogous results for the case $M_S = 3.72$. In Figure 24b, the resolution of the contact discontinuity is much worse than in Figure 23b, causing a serious error in the formation of the incident Mach stem. This type of error has been experienced by other authors (e.g. [43]) when applying Roe's approximate Riemann solver in the presence of strong shear. There is no such problem, however, in the fourth-order case (Figure 24c) where the resolution of the contact discontinuity is excellent. Also, the density wall plot in Figure 24d appears to be an improvement over the similar result in [11].

Finally, we perform a third-order calculation for this test case on a 360×80 grid. Plots for the variables density, pressure, and Mach number are shown in Figures 25a–c, using 30 equally spaced contours. This calculation was run for 800 time steps using a CFL of 0.8.
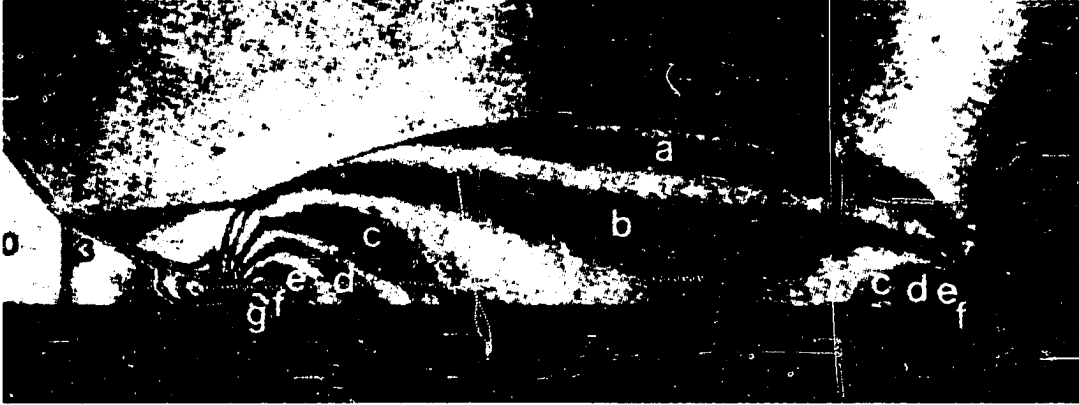
MACH REFLECTION

$$M_S = 3.72 \qquad \Theta_w = 40°$$
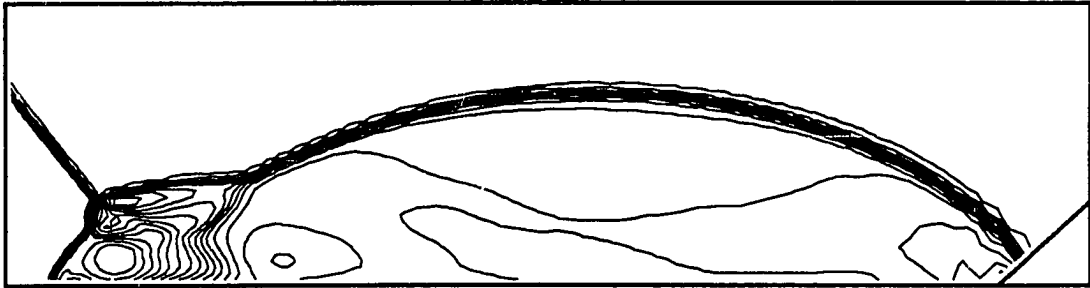

Figure 24a.   Experimental Isopycnics (Ref. [11])

Figure 24b.   Density Contours (2nd–order ENO)
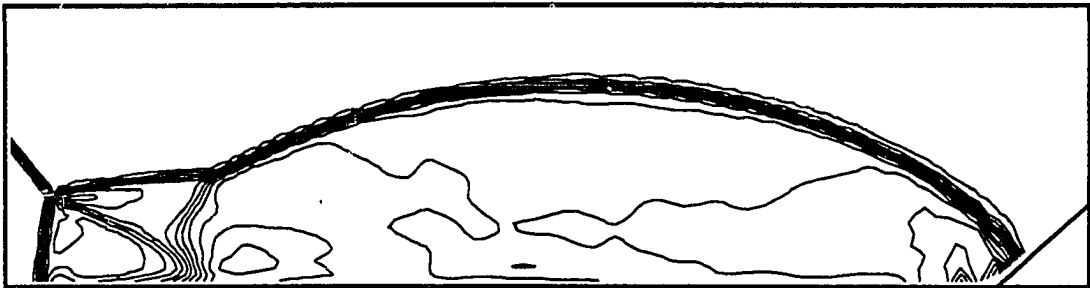

Figure 24c.   Density Contours (4th–order ENO)

152

WALL DENSITY

$$M_S = 3.72 \qquad \Theta_w = 40°$$

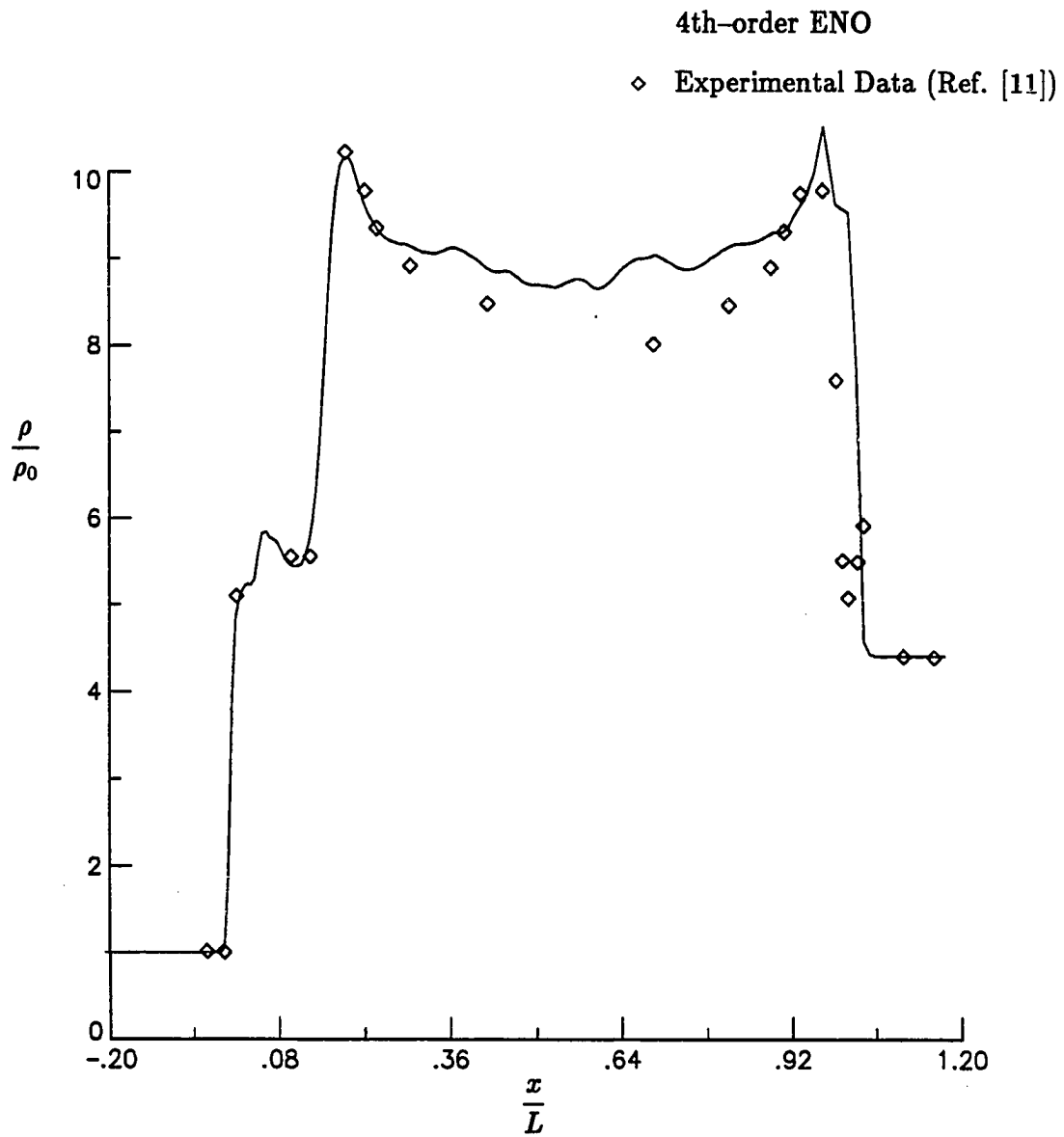4th–order ENO

◊ Experimental Data (Ref. [11])

Figure 24d.   Comparison of 4th–order ENO solution and experimental data

153

MACH REFLECTION

$M_S = 3.72$ $\Theta_w = 40°$

360 × 80 Grid
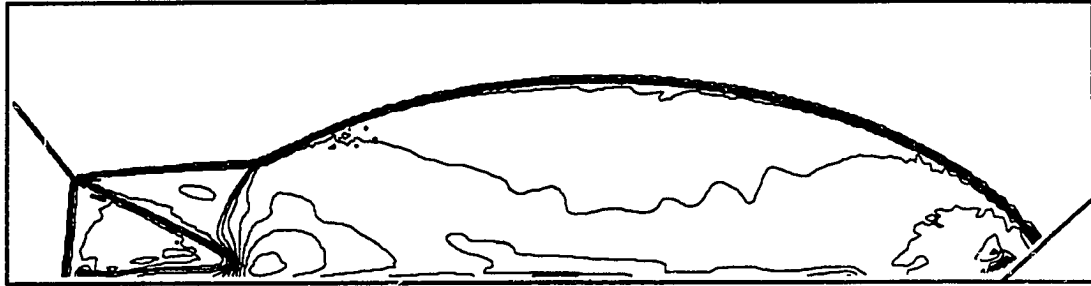

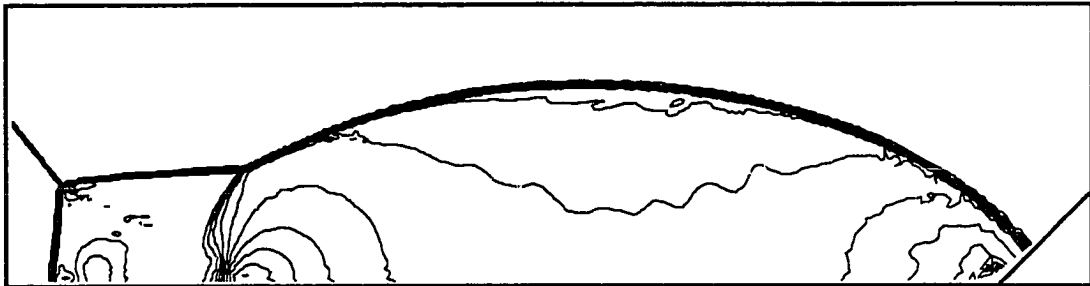**Figure 25a.   Density (3rd–order ENO)**

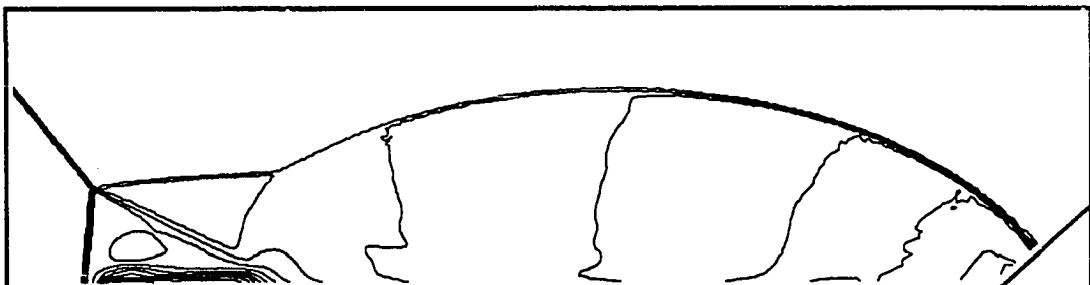**Figure 25b.   Pressure (3rd–order ENO)**

Figure 25c.   Mach No. (3rd–order ENO)

154

Chapter 7

Concluding Remarks

We conclude this dissertation with a few remarks on the accomplishments of our research, as well as some comments concerning a possible future for the use of high–order shock–capturing methods.

We have developed a rigorous finite–volume formulation for high–order ENO schemes in two spatial dimensions. Moreover, this development is complete in the sense that we have carefully considered the important issues which distinguish two–dimensional problems from those in one dimension. Specifically, these issues include the application of our scheme to boundary–value problems, non–trivial geometries, and solid walls. We further note that our scheme can be further extended to three dimensions in a straight–forward manner, in that there is nothing inherent in our two–dimensional formulation which would prohibit this extension.

Numerical test cases involving the two–dimensional Euler equations of gas dynamics have shown our scheme to be robust in the sense that certain types of flow phenomena which cannot exist in one dimension have been exhibited. We refer here to such phenomena as the slip lines, vorticity, and separation regions in our various examples in Chapter 6.

The outlook is hopeful for the usefulness of shock–capturing schemes which are of formal high–order accuracy in more than one spatial dimension. For instance, the comparison in Figures 17a–d suggest that these types of schemes might provide

155

better results than are currently available in scientific areas where the resolution of weak waves is crucial, such as in the area of acoustics. Furthermore, when compressible flow solutions are also required to be viscous, these schemes could play a major role in the computation of such flows, in which there are regions containing many local extrema, particularly when such flows also develop shocks, *e.g.* shock–turbulence interactions.

Bibliography

[1] M. Ben-Artzi, J. Falcovitz, "An Upwind Secon-Order Scheme for Compressible Duct Flows," *SIAM J. Sci. Stat. Comput.*, Vol. 7 (1986), pp. 744-768.

[2] S. R. Chakravarthy, A. Harten, and S. Osher, "Essentially Non-Oscillatory Shock-Capturing Schemes of Arbitrarily High Accuracy," *AIAA* 86-0339 (1986), Reno, Nevada.

[3] P. Colella and H. M. Glaz, "Efficient Solution Algorithms for the Riemann Problem for Real Gases," *J. Comp. Phys.*, Vol. 59 (1985), pp. 264-289.

[4] P. Colella and P. R. Woodward, "The Piecewise-Parabolic Method for Gas Dynamical Equations," *J. Comp. Phys.*, Vol. 54 (1984), pp. 174-201.

[5] R. Courant and K. O. Freidrichs, <u>Supersonic Flow and Shock Waves</u>, Wiley-Interscience, New York, (1948).

[6] R. Courant, E. Isaacson, M. Reeves, "On the Solution of Non-linear Hyperbolic Differential Equations by Finite Differences," *Communications in Pure and Applied Mathematics*, Vol. 5 (1952), pp. 243-249.

[7] J. K. Dukowicz, "A General Non-Iterative Riemann Solver for Godunov's Method," *J. Comp. Phys.*, Vol. 61 (1985), pp. 119-137.

[8] C. W. Gear, <u>Numerical Initial Value Problems in Ordinary Differential Equations</u>, Prentice-Hall (1971).

[9] I. I. Glass, "Some Aspects of Shock Wave Research," *AIAA Journal*,lVol 25, No. 2, (1987), pp. 214–229.

[10] H. M. Glaz, P. Colella, I. I. Glass, and R. L. Deschambault, "A Numerical Study of Oblique Shock Wave Reflections with Experimental Comparisons," *Proc. R. Soc. London, A 398 (1985), pp. 117-140.*

[11] H. M. Glaz, P. Colella, I. I. Glass, and R. L. Deschambault, "A Detailed Numerical, Graphical, and Experimental Study of Oblique Shock Wave Reflections," *UTIAS Report No.* 285, Institute for Aerospace Studies, University of Toronto, (1986).

[12] J. Glimm, "Solutions in the Large for Nonlinear Hyperbolic Systems of Equations," *Comm. Pure Appl. Math.*, Vol. 18 (1965), pp. 697–715.

[13] S. K. Godunov, "A Finite–Difference Method for the Numerical Computation of Discontinuous Solutions of the Equations of Fluid Dynamics," *Mat. Sb.*, Vol. 47 (1959), pp. 271–290.

[14] E. Harabetian, "A Convergent Series Expansion for Hyperbolic Systems of Conservation Laws," *NASA Contractor Report* 172557, ICASE Report No. 85–13, (February, 1985).

[15] A. Harten, "The Artificial Compression Method for Computation of Shocks and Contact Discontinuities: III. Self–Adjusting Hybrid Schemes," *Math. Comp.*, Vol. 32 (1978), pp. 363–389.

[16] A. Harten, "High Resolution Schemes for Hyperbolic Conservation Laws," *J. Comp. Phys.*, Vol. 49 (1983), pp. 357–393.

[17] A. Harten, "On a Class of High Resolution Total–Variation–Stable Finite–Difference Schemes," *SIAM J. Numer. Anal.*, Vol. 21 (1984), pp. 1–23.

[18] A. Harten, "On High–Order Accurate Interpolation for Non–Oscillatory Shock–Capturing Schemes," *MRC Technical Summary Report* No. 2829, University of Wisconsin, (1985).

[19] A. Harten, "On the Nonlinearity of Modern Shock–Capturing Schemes," *NASA Contractor Report* 178206, ICASE Report No. 86–69, (October, 1986).

[20] A. Harten, "Preliminary Results on the Extension of ENO Schemes to Two–Dimensional Problems," *Proceedings of the International Conference on Hyperbolic Problems*, Saint–Etienne, (January, 1986).

[21] A. Harten, Department of Mathematics, Tel–Aviv University, *private communication*, ICASE, 1989.

[22] A. Harten and J. M. Hyman, "A Self–Adjusting Grid for the Computation of Weak Solutions of Hyperbolic Conservation Laws," *J. Comp. Phys.*, Vol. 50 (1983), pp.235–269.

[23] A. Harten and P. D. Lax, "A Random Choice Finite–Difference Scheme for Hyperbolic Conservation Laws," *SIAM J. Numer. Anal.*, Vol. 18 (1981), pp. 289–315.

[24] A. Harten, P. D. Lax, and B. van Leer, "On Upstream Differencing and Godunov–Type Schemes for Hyperbolic Conservation Laws," *SIAM Rev.*, Vol 25 (1983), pp. 35–61.

[25] A. Harten and S. Osher, "Uniformly High–Order Accurate Non–Oscillatory Schemes I.," *MRC Technical Summary Report* No. 2823, (May, 1985).

[26] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy, "Uniformly High–Order Accurate Essentially Non–Oscillatory Schemes III.," *NASA Contractor Report* 178101, ICASE Report No. 86–22, (April, 1986).

[27] D. H. Jones, P. M. Martin, E. Thornhill, "A Note on the Pseudo–Stationary Flow Behind a Strong Shock Diffracted or Reflected at a Corner," *Proc. R. Soc. London*, A 209 (1951), pp. 238–247.

[28] P. D. Lax, "Hyperbolic Systems of Conservation Laws II.," *Comm. Pure Appl. Math.*, Vol. 10 (1957), pp. 537–566.

[29] P. D. Lax, "Weak Solutions of Nonlinear Hyperbolic Equations and Their Numerical Computation," *Comm. Pure Appl. Math.*, Vol. 7 (1954), pp. 159–193,

[30] P. D. Lax, "Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves," *Society for Industrial and Applied Mathematics*, Philadelphia, (1972).

[31] P. D. Lax and B. Wendroff, "Systems of Conservations Laws," *Comm. Pure Appl. Math.*, Vol. 13 (1960), pp. 217–237.

[32] P. D. Lax and B. Wendroff, "Difference Schemes for Hyperbolic Equations with High–Order of Accuracy," *Comm. Pure Appl. Math.*, Vol. 17 (1964), pp. 381-398.

[33] B. van Leer, "On the Relation Between the Upwind–Differencing Schemes of Godunov, Engquist–Osher, and Roe," *SIAM J. Sci. Stat. Comput.*, Vol. 5 (1984), pp. 1–20.

[34] B. van Leer, "Towards the Ultimate Conservative Difference Scheme I. The Quest of Monotonicity," *Lecture Notes in Physics*, Vol. 18, Springer–Verlag, New York,(1973), pp. 163–168.

[35] B. van Leer, "Towards the Ultimate Conservative Difference Scheme II. Monotonicity and Conservation Combined in a Second–Order Scheme," *J. Comp. Phys.*, Vol. 14 (1974), pp. 361–370.

[36] B. van Leer, "Towards the Ultimate Conservative Difference Scheme V. A Second–Order Sequel to Godunov's Method," *J. Comp. Phys.*, Vol. 32 (1979), pp. 101–136.

[37] H. W. Liepmann, A. Roshko, Elements of Gas Dynamics, Wiley & Sons, New York, (1957).

[38] R. W. MacCormack, "Numerical Solution of the Interaction of a Shock Wave with a Laminar Boundary Layer," *Proc. 2nd Inter. Conf. on Numerical Methods in Fluid Dynamics*, M. Holt (ed.), also *Lect. Notes Phys.*, Vol. 8 (1970), Springer–Verlag, New York, pp. 151–163.

[39] E. M. Murman, "Analysis of Embedded Shock Waves Calculated by Relaxation Methods," *AIAA Journal*, Vol 12 (1974), pp. 626–633.

[40] O. Oleinik, "Discontinuous Solutions of Nonlinear Differential Equations," *Usp. Mat. Nauk.*, Vol. 12 (1957), pp. 3–73; English translation in *Amer. Math. Soc. Transl. Ser. 2*, Vol. 26, pp. 95–172.

[41] S. Osher, "Riemann Solvers, the Entropy Condition, and Difference Approximations," *SIAM J. Numer. Anal.*, Vol. 21 (1984), pp. 217–235.

[42] S. Osher, Department of Mathematics, UCLA, *private communication*, ICASE, 1989.

[43] J. J. Quirk, College of Aeronautics, Cranfield Institute of Technology, Cranfield, U.K., *private communication*.

[44] R. D. Richtmeyer and K. W. Morton, Difference Methods for Initial Value Problems, 2nd ed., Wiley–Interscience, New York, (1967)

[45] P. L. Roe, "Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes," *J. Comp. Phys.*, Vol. 43 (1981), pp. 357–372.

[46] P. L. Roe, "Characteristic-Based for the Euler Equations," *Ann. Rev. Fluid Mech.*, Vol. 18 (1986), pp. 337–365.

[47] P. L. Roe, "The Use of the Riemann Problem in Finite–Difference Schemes," *Lect. Notes Phys.*, Vol. 141 (1980), pp. 354–359.

[48] P. L. Roe and J. Pike, "Efficient Construction and Utlisation of Approximate Riemann Solutions," *Computing Methods in Applied Sciences and Engineering*, R. Glowinski and J. L. Lions (ed.), Amsterdam : North–Holland, Vol. 6, pp 499–518.

[49] P. L. Roe, H. Deconick, R. Struijs, "Recent Progress in Multi–Dimensional Upwinding," *12th International Conference on Numerical Methods in Fluid Dynamics*, Oxford, England.

[50] C. Shu and S. Osher, "Efficient Implementation of Essentially Non–Oscillatory Shock–Capturing Schemes," *NASA Contractor Report* 178304, ICASE Report No. 87–33, (May, 1987).

[51] C. Shu, "TVD Time Discretizations," *UCLA Mathematics Department Report*, Los Angeles, CA 90024 (May, 1986).

[52] C. Shu and S. Osher, "Efficient Implementation of Essentially Non–Oscillatory Shock–Capturing Schemes II.," *NASA Contractor Report* 181656, ICASE Report No. 88–24, (April, 1988).

[53] J. Smoller, Shock Waves and Reaction-Diffusion Equations, Springer–Verlag, New York, (1983).

[54] J. Stoer, R. Bulirsch, Introduction to Numerical Analysis, Springer–Verlag, New York, (1980).

[55] G. A. Sod, "A Survey of Several Finite–Difference Methods for Systems of Nonlinear Hyperbolic Conservation Laws," *J. Comp. Phys.*, Vol. 27 (1978), pp. 1–31.

[56] P. Woodward and P. Colella, "The Numerical Simulation of Two–Dimensional Fluid Flow with Strong Shocks," *J. Comp. Phys.*, Vol. 54 (1984) pp. 115–173.

[57] V. Young, H. Yee, "Numerical Simulation of Shock Wave Diffraction by TVD Schemes," *AIAA-87-0112*, AIAA 25th Aerospace Sciences Meeting, January 1987.

Biographical Statement

The author was born Jay Hugh Casper, Jr. on February 24, 1954 in Quantico, Virginia. He has resided in the Tidewater area since 1959. As an undergraduate student, Mr. Casper attended Virginia Wesleyan College in Norfolk, Virginia where he received a Bachelor of Arts degree in 1977.

Several years later, he resumed his education at Old Dominion University in Norfolk, Virginia and received a Master of Science degree in Applied and Computational Mathematics in 1987. During his attendance at Old Dominion University, Mr. Casper was a teaching assistant in 1986 and in July, 1987 began working as a graduate research assistant at NASA Langley Research Center under a grant from the Institute for Computational and Applied Mechanics. Beginning July, 1989 the final year of his doctoral research was spent as a graduate research asistant, funded through the ODU Research Foundation.

In July of 1990, Mr. Casper attended the 12th International Conference on Numerical Methods in Fluid Dynamics in Oxford, England where he presented results pertaining to his doctoral dissertation, an abstract of which will appear in the conference proceedings in Lecture Notes in Physics (Decemeber, 1990). Mr. Casper will receive his degree of Doctor of Philosophy in Applied and Computational Mathematics in December of 1990.