ABSTRACT

Title of dissertation: Central Compact-Reconstruction WENO Methods

Kilian Cooley
Doctor of Philosophy, 2018

Dissertation directed by: Professor James Baeder
Department of Aerospace Engineering

High-order compact upwind schemes produce block-tridiagonal systems due to performing the reconstruction in the characteristic variables, which is necessary to avoid spurious oscillations. Consequently they are less efficient than their non-compact counterparts except on high-frequency features. Upwind schemes lead to many practical drawbacks as well, so it is desirable to have a compact scheme that is more computationally efficient at all wavenumbers that does not require a characteristic decomposition. This goal cannot be achieved by upwind schemes so we turn to the central schemes, which by design require neither a Riemann solver nor a determination of upwind directions by characteristic decomposition. In practice, however, central schemes of fifth or higher order apparently need the characteristic decomposition to fully avoid spurious oscillations. The literature provides no explanation for this fact that is entirely convincing; however, a comparison of two central WENO schemes suggests one. Pursuing that possibility leads to the first main contribution of this work, which is the development of a fifth-order, central

compact-reconstruction WENO (CCRWENO) method. That method retains the key advantages of central and compact schemes but does not entirely avoid characteristic variables as was desired. The second major contribution is to establish that the role of characteristic variables is to to make flux Jacobians within a stencil more diagonally dominant, having ruled out some plausible alternative explanations. The CCRWENO method cannot inherently improve the diagonal dominance without compromising its key advantages, so some strategies are explored for modifying the CCRWENO solution to prevent the spurious oscillations. Directions for future investigation and improvement are proposed.

Central Compact-Reconstruction WENO Methods

by

Kilian Cooley

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2018

Advisory Committee:
Professor James Baeder, Chair/Advisor
Professor Eitan Tadmor
Professor Howard Elman
Professor M. Pino Martin
Professor Maria Cameron

Acknowledgments

I would like to thank my advisor, Dr. James Baeder, for having been a source of ideas and guidance and for providing me the latitude to see this project through to its fullest conclusion. My professional development has benefited from my association with him. I would also like to thank Dr. Eitan Tadmor for riveting lectures and genuine interest in my work. Finally, I would like to express my gratitude to my committee members: Dr. Howard Elman, Dr. Pino Martin, and Dr. Maria Cameron. I appreciate your input and the time you devoted on my behalf.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ENO	Essentially Non-Oscillatory
WENO	Weighted Essentially Non-Oscillatory
CWENO	Central Weighted Essentially Non-Oscillatory
CRWENO	Compact-Reconstruction Weighted Essentially Non-Oscillatory
CCRWENO	Central Compact-Reconstruction Weighted Essentially Non-Oscillatory
TENO	Targeted Essentially Non-Oscillatory
PDE	Partial Differential Equation
IVP	Initial Value Problem
BC	Boundary Condition
SA	Subcell Average
PV	Point Value
FD	Flux Derivative
RK	Runge-Kutta
SSP	Strong-Stability-Preserving

Chapter 1: Introduction

1.1 Physical Motivation

Engineers need to accurately predict the behavior of flow around an aircraft in order to predict its performance, which requires solving numerically the governing equations - the Euler or Navier-Stokes equations. Flows of aerodynamic interest often contain discontinuities (shocks) and small-scale features (turbulent eddies) which must both be computed accurately to predict, for example, drag or jet noise. Shocks cause some numerical methods to produce spurious oscillations in the solution, which can be mistaken for actual flow features. Turbulent eddies can be small, so to resolve them properly requires a fine discretization of the problem which leads to large memory requirements for the computation. Efficient methods with a high order of accuracy can resolve the same flow features with a coarser discretization, thus saving memory, but if they are linear will produce oscillations near shocks [1]. Thus, to resolve both types of features with one method requires a nonlinear method. The present work develops a method for solving such equations that retains beneficial properties of one class of methods and avoids a severe drawback of another such class.

1.2 Theoretical Background

Consider a hyperbolic system of conservation laws

$$\frac{\partial q}{\partial t} + \frac{\partial}{\partial x_d} f_d(q) = 0 \tag{1.1}$$

with a prescribed initial condition

$$q(x, 0) = q_0(x) \tag{1.2}$$

where $x = (x_1, \cdots, x_D) \in \mathbb{R}^D$ is the vector of spatial coordinates, t is time, $q = q(x, t) \in \mathbb{R}^C$ is the state vector, and $f_d = f_d(q) : \mathbb{R}^C \to \mathbb{R}^C$ are the flux vectors. Throughout this work the Einstein summation notation will be used, meaning that repeated subscripts in a single additive term indicate summation over all possible values of that subscript. A system of the form Eq. (1.1) is called hyperbolic if, for all real w_i, the matrix $J = \sum_{d=1}^{D} w_d J_d$, where J_d are the Jacobians of the fluxes f_d, is real-diagonalizable [2]. For purposes of this introduction, we consider the one-dimensional case $D = 1$.

Even if the initial data is smooth, solutions to the initial value problem Eqs. (1.1)-(1.2) can become nonsmooth and even discontinuous. Thus it is to be understood that solutions are meant in the sense of distributions, meaning that they satisfy the weak form of the conservation law Eq. (1.1):

$$\int_0^\infty \int_{\mathbb{R}^D} q \frac{\partial \phi}{\partial t} + f_d \frac{\partial \phi}{\partial x_d} \, dx \, dt + \int_{\mathbb{R}^D} \phi(x, 0) q(x, 0) \, dx = 0 \tag{1.3}$$

for any smooth function $\phi(x, t)$ with compact support in $\mathbb{R}^D \times [0, \infty)$. These weak solutions are not necessarily unique [2], so an additional criterion is required to select

the unique physically relevant solution. Equations of the form Eq. (1.1) often arise as models of systems with small diffusive effects in the limit as those effects vanish. For example, the Navier-Stokes equations governing viscous fluid flow become the Euler equations for inviscid flow as the viscosity parameter approaches zero. Consider a modified version of Eq. (1.1):

$$\frac{\partial q^\epsilon}{\partial t} + \frac{\partial}{\partial x_d} f_d(q^\epsilon) = \epsilon \frac{\partial q^\epsilon}{\partial x_d \partial x_d} \tag{1.4}$$

The limit as $\epsilon \to 0$ of this family of solutions may be a solution of the weak form Eq. (1.3). The details of this process are not relevant to the present work but the reader may consult e.g. [2] for a thorough discussion.

Alternatively, one may require an entropy inequality [3] to hold for the unique physical solution. An entropy pair associated with the conservation law Eq. (1.1) consists of a convex entropy $\eta : \mathbb{R}^C \to \mathbb{R}$ and an entropy flux $E = (E_1, \cdots E_D) : \mathbb{R}^C \to \mathbb{R}^D$ for which:

$$\frac{\partial \eta}{\partial q_c} \frac{\partial f_{cd}}{\partial q_j} = \frac{\partial E_d}{\partial q_j} \tag{1.5}$$

(note that f_{cd} refers to the cth component of the flux for dimension d). This definition is constructed to enable the formal equivalence of the entropy conservation law:

$$\frac{\partial}{\partial t} \eta(q) + \frac{\partial}{\partial x_d} E_d(q) = 0 \tag{1.6}$$

and the original conservation law Eq. (1.1). Indeed:

$$\begin{aligned}\frac{\partial \eta}{\partial t}+\frac{\partial E_d}{\partial x_d} &= \frac{\partial \eta}{\partial q_c}\frac{\partial q_c}{\partial t}+\frac{\partial E_d}{\partial q_j}\frac{\partial q_j}{\partial x_d} \\ &= \frac{\partial \eta}{\partial q_c}\frac{\partial q_c}{\partial t}+\frac{\partial \eta}{\partial q_c}\frac{\partial f_{cd}}{\partial q_j}\frac{\partial q_j}{\partial x_d} \\ &= \sum_{c=1}^{C}\frac{\partial \eta}{\partial q_c}\left(\frac{\partial q_c}{\partial t}+\frac{\partial f_{cd}}{\partial x_d}\right)=0\end{aligned} \tag{1.7}$$

Therefore if q is a continuously differentiable classical solution to the original conservation law then the entropy η is also conserved with fluxes E_d. If, however, the solution is not differentiable then these manipulations are invalid. In that case, a vanishing-viscosity solution q^ϵ satisfies:

$$\begin{aligned}\frac{\partial \eta(q^\epsilon)}{\partial t}+\frac{\partial}{\partial x_d}E_d(q^\epsilon) &= \frac{\partial \eta(q^\epsilon)}{\partial q_c}\frac{\partial q_c^\epsilon}{\partial t}+\frac{\partial \eta(q^\epsilon)}{\partial q_c}\frac{\partial f(q^\epsilon)_{cd}}{\partial x_d} \\ &= -\epsilon\frac{\partial \eta}{\partial q_c}\frac{\partial q_c^\epsilon}{\partial x_d \partial x_d} \\ &= \epsilon\frac{\partial^2 \eta}{\partial x_d \partial x_d}-\epsilon\frac{\partial q_r}{\partial x_d}\frac{\partial^2 \eta}{\partial q_r \partial q_c}\frac{\partial q_c}{\partial x_d} \\ &\leq \epsilon\frac{\partial^2 \eta}{\partial x_d \partial x_d}\end{aligned} \tag{1.8}$$

Convexity of η implies that the right-hand side of the last inequality is non-negative, from which it follows [3] that bounded limits of vanishing-viscosity solutions satisfy (a.e.):

$$\frac{\partial \eta}{\partial t}+\frac{\partial E_d}{\partial x_d}\leq 0 \tag{1.9}$$

A weak solution that satisfies this entropy inequality (in the sense of distributions) for all η and corresponding E_d is called an entropy or entropic solution. For details, see [2]. Other characterizations of the entropy solution exist, of which Eq. (1.9) is especially useful because numerical methods that satisfy a discrete analog of it can be shown to converge to the entropy solution [4].

1.2.1 The Euler Equations

This work deals mostly with the Euler equations of inviscid fluid flow. They are restatements of the physical principles of mass, momentum, and energy conservation for a fluid with no viscosity. Let ρ be the density of the fluid, u_i its velocity in each coordinate direction $i = 1, \cdots, D$, and E be the energy per unit volume. Then:

$$q = \begin{bmatrix} \rho \\ \rho u_i \\ E \end{bmatrix}, \quad f_d = \begin{bmatrix} \rho u_d \\ \rho u_i u_d + P\delta_{id} \\ (E+P)u_d \end{bmatrix} \tag{1.10}$$

where δ_{id} is the Kronecker delta and P is the pressure. As written this system is not closed, as P must be specified from the components of q. To calculate P we need an equation of state, which depends on the physical model of the fluid being considered. In this work we will assume a perfect gas, for which the specific heat capacities at constant volume and pressure are constant and the ideal gas law $P = \rho RT$ holds, where T is the temperature and R is a gas-specific constant.

$$P = (\gamma - 1)(E - \rho u_i u_i/2) \tag{1.11}$$

where γ is the ratio of specific heat capacities and is also a constant, which we set to the value for air unless otherwise noted: $\gamma = 1.4$. With this addition the Euler equations are closed and can be solved numerically.

1.3 The Finite Volume Framework

The finite-volume framework provides the means for obtaining a discrete system that approximates the behavior of the continuous problem Eq. (1.1) and we describe it in this section. For further details, consult [5] or [6]. Integrate the conservation law Eq. (1.1) over a space-time control volume $\Omega \times [t, t + \Delta t]$:

$$\int_\Omega q(x, t + \Delta t)\, dx = \int_\Omega q(x, t)\, dx - \int_t^{t+\Delta t} \int_{\partial\Omega} f_d(q(x), t) n_d\, dS dt \tag{1.12}$$

n_d are the components of the outward unit normal vector along the boundary $\partial\Omega$ of Ω. If this equation holds for any time interval $[t, t + \Delta t]$ and any spatial domain Ω, then the weak formulation Eq. (1.3) and the classical formulation Eq. (1.1) are equivalent for weak solutions that are piecewise smooth.

To simplify the forthcoming discussion, consider the one-dimensional case of Eq. (1.12) and let the spatial domain be an interval of small width Δx: $\Omega = [x - \Delta x/2, x + \Delta x/2]$. Then Eq. (1.12) can be written as:

$$\bar{q}(x, t + \Delta t) = \bar{q}(x, t) - \lambda \left[\hat{f}(x + \Delta x/2, t) - \hat{f}(x - \Delta x/2, t) \right] \tag{1.13}$$

where $\bar{q}(x, t)$ indicates an average over an interval centered at the point x:

$$\bar{q}(x, t) = \frac{1}{\Delta x} \int_{x-\Delta x/2}^{x+\Delta x/2} q(\xi, t)\, d\xi \tag{1.14}$$

the $\hat{f}$ are time-averaged fluxes through the boundaries of Ω:

$$\hat{f}(x, t) = \frac{1}{\Delta t} \int_t^{t+\Delta t} f(q(x, \tau))\, d\tau \tag{1.15}$$

and the mesh ratio $\lambda = \Delta t/\Delta x$. One sees that if an approximate solution is available that can be evaluated at an arbitrary point x and if the time integrals can be evaluated, then the sliding averages $\bar{q}(x,t)$ can be evolved exactly using Eq. (1.13). Since the averages at time $t+\Delta t$ are centered at the same points as the averages at time t, the first step in applying a finite-volume method is to cover a large domain Ω with small cells $\Omega_j, j = 1, \cdots, N$. The averages of $q(x,t)$ over the cell Ω_j at time $t_n = n\Delta t$ are denoted $\bar{q}_j^n$ and Eq. (1.13) is applied to them. This setup leaves the user with two decisions, which distinguish the varieties of finite-volume method: first, the choice of control volumes; and second, the strategy for obtaining the interface flux from the cell averages. In the next two subsections we discuss two categories of schemes that differ in the choice of control volumes. Section 1.4 discusses some of the many alternatives for reconstructing interface values from cell averages.

1.3.1 Upwind Schemes

Taking the space-time control volumes to coincide with the cells, i.e. $\Omega_j \times [t^n, t^{n+1}]$ is a control volume for each j, gives rise to the family of upwind schemes. One obtains the evolution equation for the cell averages $\bar{q}_j^n$ simply by setting $x = x_j, t = t^n$ in the evolution equation for the sliding averages, Eq. (1.13), which gives:

$$\bar{q}_j^{n+1} = \bar{q}_j^n - \lambda \left[\hat{f}(x_j + \Delta x/2, t^n) - \hat{f}(x_j - \Delta x/2, t^n) \right] \tag{1.16}$$

The quantities $\hat{f}$ are time-averages of the exact flux, so they must be evaluated approximately by numerical fluxes $H_{j+1/2}^n \approx \hat{f}(x_{j+1/2}, t^n)$. The original upwind

scheme of Godunov [1] computes the $\hat{f}$ exactly by the following approach. Suppose the solution is globally represented as a function $\tilde{q}(x)$ that is piecewise constant in each cell Ω_j and that matches the averages $\bar{q}_j$ over those cells.

$$\tilde{q}(x) = \bar{q}_j \qquad \text{if } x_{j-1/2} < x < x_{j+1/2} \text{ for each } j \tag{1.17}$$

Then each interface can be considered as a (shifted) Riemann problem, i.e. the original conservation law Eq. (1.1) with an initial condition of the form:

$$q(x,0) = \begin{cases} q_L & x < 0 \\ \\ q_R & x > 0 \end{cases} \tag{1.18}$$

For sufficiently small times, the solutions near each interface evolve independently of each other due to the finite propagation speed that is a feature of hyperbolic systems [6]. That is, any disturbance travels at a speed bounded by the spectral radius $\rho(J)$ of the flux Jacobian J, where $J_{ij} = \frac{\partial f_i}{\partial q_j}$. Thus the piecewise-constant representation of the solution evolves for small times as a set of non-interacting Riemann problems.

It can be shown [2] that solutions to hyperbolic systems of conservation laws are self-similar, so that the solution $q(x,t)$ depends only on the parameter x/t. In particular, the solution is constant along the line $x = 0$ in the space-time plane. This definition assumes the initial discontinuity to be located at $x = 0, t = 0$, so for the purpose of evolving the piecewise-constant numerical solution the Riemann solutions are translated in space and time to their corresponding interfaces. Denote by $\mathcal{R}^n_{j+1/2}(\xi)$ the solution to the Riemann problem at interface $j + 1/2$ at time t^n,

which has initial data $q_L = \bar{q}^n_j, q_R = \bar{q}^n_{j+1}$ and with self-similarity coordinate $\xi = (x - x_{j+1/2})/(t - t^n)$. Then the solution at $x = x_{j+1/2}$ is simply $q(x_{j+1/2}, t) = \mathcal{R}^n_{j+1/2}(0)$ for sufficiently short times. This construction naturally leads to the following definition for the numerical interface flux in terms of the Riemann solution:

$$H^n_{j+1/2} = \hat{f}(x_{j+1/2}, t^n) = \frac{1}{\Delta t} \int_t^{t+\Delta t} f(\mathcal{R}^n_{j+1/2}(0))\, d\tau = f(\mathcal{R}^n_{j+1/2}(0)) \tag{1.19}$$

The Godunov strategy solves the problem of defining a single numerical flux at a point where the numerical solution is discontinuous. It has, however, two serious drawbacks. First, it requires an analytical solution to the Riemann problem, which may be difficult to compute or may not even exist. Several strategies for obtaining approximate Riemann solutions have been proposed and used with success, notably the Roe scheme [7]. Numerous such approximations are available: see [5], [8], or, for a summary of the simplest, [6]. The second drawback is that the piecewise-constant representation of the solution limits the scheme to first-order accuracy in space. This limit can be improved by using higher-order reconstructions (see Section 1.4), replacing the piecewise constants with higher-degree polynomials, but then instead of solving Riemann problems at each interface one must solve generalized Riemann problems (where the initial data is not piecewise constant). This task is even more difficult than the original task of solving bona fide Riemann problems, though in practice the generalized Riemann problems are often treated as true Riemann problems [9]. In practical applications, the use of Riemann solvers introduces several details with annoying consequences. The availability of multiple options introduces an element of arbitrariness, and one may need to experiment with different solvers

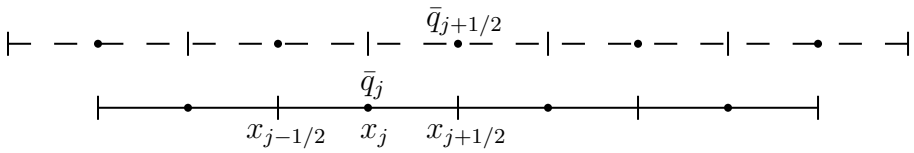

Figure 1.1: A main grid (solid line) and its staggered counterpart (dashed line).

for a single problem. The Riemann solvers also incur computational expense per cell interface which can become costly on large problems. Finally, one might question the physical accuracy of a method that places discontinuities at every interface even where the solution is smooth. Thus there are several conceptual and practical advantages to a scheme that does not require any Riemann solver. The next subsection describes how such a scheme can be obtained by making a different choice of the control volumes.

1.3.2 Central Schemes

Nessyahu and Tadmor [10] originated the idea of representing the solution at the next time step on a staggered grid whose cells are centered on the cell interfaces of the grid used at the current time step, as in Figure 1.1. In effect, the control volumes are of the form $[x_j, x_{j+1}] \times [t^n, t^n + 1]$ rather than $[x_{j-1/2}, x_{j+1/2}] \times [t^n, t^n + 1]$ as in the upwind schemes.

Considering the same piecewise-constant numerical solution as before, we see that for sufficiently small time steps the waves from Riemann solutions at an interface $x_{j+1/2}$ will not reach the edges of the control volume. Therefore at the cell centers, where point values are evaluated, never encounter a discontinuity. Thus we

can write the evolution equation for the averages $\bar{q}^n_{j+1/2}$ over the staggered grid as:

$$\bar{q}^{n+1}_{j+1/2} = \bar{q}^n_{j+1/2} - \lambda(\hat{f}(x_{j+1}, t^n) - \hat{f}(x_j, t^n)) \tag{1.20}$$

The solution at the next time step is realized on the staggered grid whose cell centers coincide with the cell interfaces on the main grid. With this choice of control volume the flux function is evaluated only at points where the numerical solution $\tilde{q}(x)$ is continuous (i.e. single-valued), so no Riemann problems ever appear. This is the key advantage of such schemes. Alternatively, one may think of the central schemes as averaging over all the waves emanating from the discontinuities at interfaces, as opposed to resolving them with a Riemann solver [6].

The initial averages over the staggered grid can be reconstructed in a way that guarantees conservation by computing the half-averages $\bar{q}^L_j$ of $\tilde{q}(x)$ over the left subcells $[x_{j-1/2}, x_j]$

$$\bar{q}^L_j = \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_j} \tilde{q}(x)\, dx \tag{1.21}$$

and defining the right subcell average as $\bar{q}^R_j = \bar{q}_j - \bar{q}^L_j$. Then it is also necessary to reconstruct the point values of the solution at the cell midpoints in order to evaluate the numerical fluxes. Both of these reconstructions are trivial if the numerical solution is taken as the piecewise-constant approximation, which leads to the following evolution equation for the cell-averages:

$$\bar{q}^{n+1}_{j+1/2} = \frac{1}{2}(\bar{q}^n_j + \bar{q}^n_{j+1}) - \lambda(f(q^n_{j+1}) - f(q^n_j)) \tag{1.22}$$

Note the similarity to the classical Lax-Friedrichs scheme [11] [12]. As with the upwind schemes, the piecewise-constant reconstruction limits this scheme to first-order

accuracy which can be improved by using higher-degree polynomials in the numerical solution $\tilde{q}$. The Nessyahu-Tadmor scheme [10], for example, uses a piecewise-linear reconstruction with a slope limiter which both enables second-order accuracy and ensures that the scheme is total-variation diminishing in the scalar case.

Central schemes also have two drawbacks. First, because the waves of the Riemann fan must not reach cell centers from interfaces, as opposed to reaching other interfaces in the upwind scheme, the maximum time step for a central scheme is half that for the corresponding upwind scheme [10] [13]. Second, and more importantly, central schemes have numerical dissipation that scales with Δt^{-1} which causes them to excessively smear shocks and fine features [13] [6]. For example, by rearranging (1.22), we obtain:

$$\frac{\bar{q}^{n+1}_{j+1/2} - \bar{q}^{n}_{j+1/2}}{\Delta t} = \frac{\bar{q}^{n}_{j} - 2\bar{q}^{n}_{j+1/2} + \bar{q}^{n}_{j+1}}{2\Delta t} - \frac{1}{\Delta x}(f(q^{n}_{j+1}) - f(q^{n}_{j})) \tag{1.23}$$

Taylor series analysis shows that:

$$\bar{q}^{n}_{j} - 2\bar{q}^{n}_{j+1/2} + \bar{q}^{n}_{j+1} = \frac{1}{8}(\Delta x^2 q''(x_{j+1/2})) + \mathcal{O}(\Delta x^4) \tag{1.24}$$

Because the staggered cells have the same shape regardless of Δt, the leading coefficient of this expansion is also independent of Δt [13]. This fact also means that the scheme (1.22) cannot be reduced to a semi-discrete form that can be used with a Runge-Kutta scheme for advancing in time. Kurganov and Tadmor [13] provided the remedy for this problem, which is to let the width of the control volumes depend on Δt while still containing all waves of the Riemann fan at its interface [13]. If $a^{+}_{j+1/2}, a^{-}_{j+1/2}$ are upper bounds for the speeds of the waves traveling in the $+x$ and $-x$ directions respectively, then the corresponding control volume

is $[x_{j+1/2} - a^-_{j+1/2}\Delta t, x_{j+1/2} + a^+_{j+1/2}\Delta t] \times [t^n, t^{n+1}]$. The so-called central upwind schemes resulting from this modification leads to a semi-discrete form and avoids the Δt^{-1} scaling of the numerical dissipation [13].

1.4 The Reconstruction Problem

In either category of scheme, a key problem is to obtain from the cell averages other quantities of interest: interface values in the upwind schemes, and subcell averages and midpoint values in the central schemes. A pth-order accurate approximation $\hat{Q}_j$ of a desired quantity Q_j can be obtained by, for example, a linear combination of averages in adjacent cells:

$$\hat{Q}_j = \sum_{v=a}^{b} R_v \bar{q}_{j+v} = Q_j + \mathcal{O}(\Delta x^p) \tag{1.25}$$

As alluded to in the first section, such schemes have a devastating drawback preventing their straightforward application to hyperbolic conservation laws. If the coefficients R_v are fixed, then the approximation $\hat{Q}_j$ will be oscillatory. Figure 1.2 shows an example of this behavior, where the scheme $q_{j+1/2} = \frac{1}{3}\bar{q}_{j-2} - \frac{7}{6}\bar{q}_{j-1} + \frac{11}{6}\bar{q}_j$ is applied to the cell averages of a piecewise constant function to estimate interface values. Beyond merely degrading solution accuracy, these oscillations can make solutions nonphysical by violating physical constraints such as positivity of density. Such oscillations always arise, however, from any linear reconstruction whose order of accuracy exceeds 1 [1]. To obtain the desired high-order accuracy without spurious oscillations therefore requires a nonlinear scheme, i.e. one in which the coefficients are not predetermined constants. In the next section we discuss the WENO family

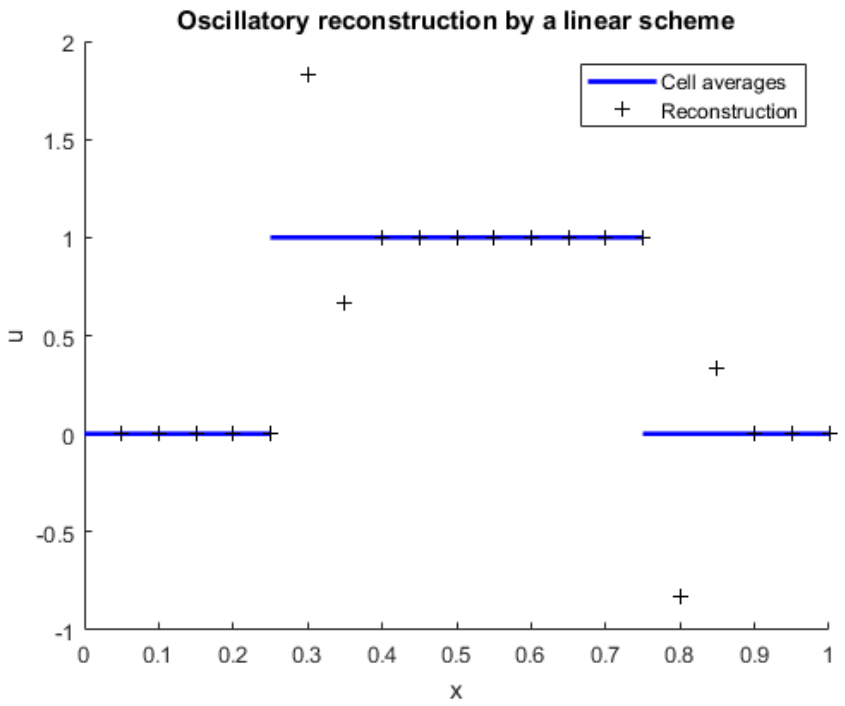

Figure 1.2: A high-order linear reconstruction produces oscillations near discontinuities.

of such schemes.

1.4.1 WENO and CRWENO Reconstructions

Consider a scheme that computes a quantity associated with cell j using information in cells $j + a, \cdots, j + b$. We refer to this set of cells as the *stencil* of the scheme. The oscillations in Figure 1.2 occur when the reconstruction stencil includes a discontinuity, which suggests that the role of the necessarily nonlinearity should be to automatically adapt the stencil to avoid crossing discontinuities. The Weighted Essentially Non-Oscillatory (WENO) schemes (originally presented by Liu, Osher, and Chan in [14]) achieve this by taking the final reconstruction

to be a convex combination of candidate reconstructions, with the weights in the convex combination approaching zero (as the cell size approaches zero) when their corresponding stencil contains a discontinuity. For example:

$$\begin{aligned} \hat{Q}_j &= \sum_{s=1}^{S} \omega_s \hat{Q}_j^{(s)}, \ \sum_{s=1}^{S} \omega_s = 1 \\ \hat{Q}_j^{(s)} &= Q_j + \mathcal{O}(\Delta x^q) \ \ \forall s \end{aligned} \tag{1.26}$$

For example, the fifth-order WENO scheme popularized by [15] is given by:

$$\begin{aligned} \hat{q}_{j+1/2}^{(1)} &= \frac{1}{3}\bar{q}_{j-2} - \frac{7}{6}\bar{q}_{j-1} + \frac{11}{6}\bar{q}_j & \bar{\omega}_1 &= \frac{1}{10} \\ \hat{q}_{j+1/2}^{(2)} &= -\frac{1}{6}\bar{q}_{j-1} + \frac{5}{6}\bar{q}_j + \frac{1}{3}\bar{q}_{j+1} & \bar{\omega}_2 &= \frac{3}{5} \\ \hat{q}_{j+1/2}^{(3)} &= \frac{1}{3}\bar{q}_j + \frac{5}{6}\bar{q}_{j+1} - \frac{1}{6}\bar{q}_{j+2} & \bar{\omega}_3 &= \frac{3}{10} \end{aligned} \tag{1.27}$$

We refer to the schemes that provide the candidate reconstructions as *subschemes*. Because the weight for a discontinuous stencil is (approximately) zero, the subscheme that would produce oscillations has its contribution reduced to (approximately) zero thus preventing the spurious oscillations of Figure 1.2. On the other hand, when no stencil contains a discontinuity, all the weights approach ideal values $\bar{\omega}_s$ for which the convex combination attains its maximum order of accuracy:

$$\sum_{s=1}^{S} \bar{\omega}_s \hat{Q}_j^{(s)} = Q_j + \mathcal{O}(\Delta x^{q+S-1}) \tag{1.28}$$

This maximum order of accuracy can be achieved when the weights $\omega_s \neq \bar{\omega}_s$ if the non-oscillatory weights ω_s approach their ideal values quickly enough. The

non-oscillatory reconstruction Eq. (1.26) can be written:

$$\begin{aligned}\hat{Q}_j &= \sum_s (\omega_s - \bar{\omega}_s)\hat{Q}_j^{(s)} + \sum_s \bar{\omega}_s \hat{Q}_j^{(s)} \\ &= \left(\sum_s (\omega_s - \bar{\omega}_s)(Q_j + \mathcal{O}(\Delta x^q))\right) + Q_j + \mathcal{O}(\Delta x^{q+S-1}) \\ &= Q_j + \left(\sum_s (\omega_s - \bar{\omega}_s)\mathcal{O}(\Delta x^q)\right) + \mathcal{O}(\Delta x^{q+S-1})\end{aligned} \tag{1.29}$$

Thus to attain the maximum order of accuracy we require the sufficient condition (cf. Henrick et al. [16]):

$$\omega_s - \bar{\omega}_s = \mathcal{O}(\Delta x^{S-1}) \tag{1.30}$$

The order of convergence here can be relaxed by considering the specific coefficients of the truncation error expansions of the subschemes (see [16]) but it is often extremely difficult to design schemes that meet the resulting condition. Therefore in Section 1.4.2 we will seek weights that satisfy Eq. (1.30).

A WENO scheme is completely determined by its constituent subschemes, its ideal weights $\bar{\omega}_s$, and the algorithm for computing the nonlinear weights ω_s from the ideal weights. The ideal weights can be determined by examining the truncation errors of the individual subschemes. Section 1.4.2 will describe several strategies for relating ω_s to $\bar{\omega}_s$. The subschemes themselves can be of the form Eq. (1.25). Alternatively, Ghosh and Baeder proposed in [17] and [18] that the subschemes be the compact (i.e. spatially implicit) schemes of Lele [19]. These have the form:

$$\sum_v L_v \hat{Q}_{j+v} = \sum_v R_v \bar{q}_{j+v} \tag{1.31}$$

A convex combination of compact schemes leads to a linear system to be solved for the unknown interface values. If the left-hand side coefficients L_v are chosen with

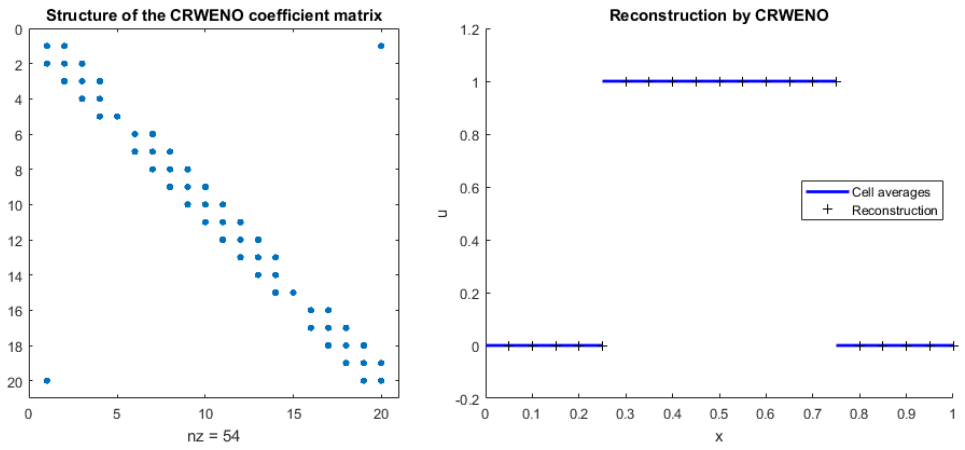

Figure 1.3: The CRWENO matrix partitions into blocks at discontinuities to provide a non-oscillatory reconstruction of a discontinuous function.

zeros in appropriate places then discontinuities cause the linear system to partition into decoupled blocks at shocks. For example, the original compact-reconstruction WENO (CRWENO) scheme in [18] is:

$$\begin{aligned} \frac{2}{3}\hat{q}_{j-1/2}+\frac{1}{3}\hat{q}_{j+1/2} &= \frac{1}{6}\bar{q}_{j-1}+\frac{5}{6}\bar{q}_j & \bar{\omega}_1 &= \frac{1}{5} \\ \frac{1}{3}\hat{q}_{j-1/2}+\frac{2}{3}\hat{q}_{j+1/2} &= \frac{5}{6}\bar{q}_j+\frac{1}{6}\bar{q}_{j+1} & \bar{\omega}_2 &= \frac{1}{2} \\ \frac{2}{3}\hat{q}_{j+1/2}+\frac{1}{3}\hat{q}_{j+3/2} &= \frac{1}{6}\bar{q}_j+\frac{5}{6}\bar{q}_{j+1} & \bar{\omega}_3 &= \frac{3}{10} \end{aligned} \tag{1.32}$$

Applying this scheme to the box function in Figure 1.2 leads to a linear system with the sparsity pattern shown in Figure 1.3. Not only are clear block divisions visible, indicating decoupling across the discontinuities, but the reconstruction has none of the spurious oscillations seen in Figure 1.2.

CRWENO has the same advantage over WENO that the compact schemes of [19] have over the non-compact schemes of the form Eq. (1.25). The coupling of unknowns allows waves of higher frequency to be resolved on a given grid, and the

truncation errors for compact schemes are smaller in absolute terms than those of the non-compact schemes. For applications in aerodynamics, these properties mean CRWENO schemes require coarser grids to resolve the same turbulent eddies while still preventing spurious oscillations at shocks. We will see in section 1.7, however, that CRWENO schemes have a severe drawback.

1.4.2 Choice of Weights

WENO and CRWENO schemes require an algorithm to modify the ideal weights $\bar{\omega}_s$ so that the resulting ω_s approach zero if the stencil s contains a discontinuity and approach $\bar{\omega}_s$ if no stencils contain discontinuities. In all cases the ω_s must form a convex combination i.e. we must have $\omega_s \geq 0 \; \forall s$ and $\sum_s \omega_s = 1$. A typical WENO scheme meets these conditions by setting:

$$\omega_s = \frac{\alpha_s}{\sum_v \alpha_v} \tag{1.33}$$

$$\alpha_s = \frac{\bar{\omega}_s}{(\epsilon + IS_s)^p} \tag{1.34}$$

Where IS_s is a smoothness indicator for the stencil s, which is large when the stencil contains a discontinuity. Typically the exponent $p = 2$. The parameter ϵ is a small value whose sole purpose is to prevent division by zero. The original WENO scheme [14] defined the smoothness indicators associated with cell j in terms of undivided differences of the solution:

$$IS_s = \sum_{l=1}^{S-1} \sum_{i=1}^{S-l} \frac{(q[j - S + s + i, l])^2}{S - l} \tag{1.35}$$

where S is the number of stencils and the $u[\cdot,\cdot]$ is an undivided difference:

$$\begin{aligned} q[j,0] &= q_j \\ q[j,l] &= q[j+1,l-1] - q[j,l-1] \end{aligned} \tag{1.36}$$

Thus for $S = 3$ we have

$$\begin{aligned} IS_s &= \frac{1}{2}(q_{j+s-1} - q_{j+s-2})^2 + \frac{1}{2}(q_{j+s} - q_{j+s-1})^2 + (q_{j+s} - 2q_{j+s-1} + q_{j+s-2})^2 \\ &= \frac{1}{2}(\Delta x q_j' + \mathcal{O}(\Delta x))^2 + \frac{1}{2}(\Delta x q_j' + \mathcal{O}(\Delta x))^2 + (\Delta x^2 q_j'' + \mathcal{O}(\Delta x^3))^2 \\ &= (\Delta x q_j')^2(1 + \mathcal{O}(\Delta x)) \end{aligned} \tag{1.37}$$

This weight definition therefore does not satisfy the convergence criterion Eq. (1.30) to ensure maximum order of convergence in smooth regions. Jiang and Shu [15] presented another smoothness indicator that measures the smoothness in terms of L^2 norms of derivatives of the reconstruction polynomial $q_s(x)$ (i.e. the polynomial whose averages over the cells in the stencil match the given cell averages):

$$IS_s = \sum_{r=1}^{S-1} \int_{x_{j-1/2}}^{x_{j+1/2}} (\Delta x^r q_s^{(r)})^2 \, \frac{dx}{\Delta x} \tag{1.38}$$

At points where the first derivative of $q(x)$ is nonzero, these indicators satisfy the weight convergence criterion Eq. (1.30). They fail to do so, however, at points where first and higher derivatives vanish. Several alternative weighting strategies have been proposed that do not have this defect. In [16], a mapping function modifies the ω_s computed with the Jiang-Shu indicators in such a way as to accelerate convergence to their ideal values at critical points. The mapping process adds computational expense to the scheme, so Borges et al. in [20] modified the definitions of α_s in Eq. (1.34) to incorporate information about higher derivatives:

$$\alpha_s = \bar{\omega}_s \left(1 + \frac{\tau}{\epsilon + IS_s}\right)^q, \;\; \tau = |IS_0 - IS_2|, \;\; s = 1,2,3 \tag{1.39}$$

where IS_s are the Jiang-Shu indicators. This definition allows the maximum order of accuracy to be achieved at critical points but fails to do so when the first, second, and third derivatives vanish at a point [21]. Furthermore, the absolute value in the definition of τ can cause a loss of accuracy at points where τ changes sign, and for schemes with more subschemes the difference between the first and last indicators does not provide the desired accuracy. Yamaleev and Carpenter in [21] chose instead to set τ as the square of the highest-order undivided difference that can be defined on the combined stencil (i.e. the union of the stencils of all the subschemes). This definition prevents the order of accuracy from degenerating in the presence of any number of vanishing derivatives and generalizes to schemes with more than three subschemes.

1.5 Time Advancement

Two varieties of time-advancement schemes will be relevant to this work. First, the Runge-Kutta family of schemes which take the form:

$$\begin{aligned} u^{(s)} &= u^n + \Delta t \left(\sum_{k=1}^{S} a_{sk} F(t + c_k \Delta t, u^{(k)}) \right), \ s = 1, 2, \cdots, S \\ u^{n+1} &= u^{(S)} \end{aligned} \tag{1.40}$$

where $F(t, q) = dq/dt$. Many such methods exist, varying in the number of stages S and in their regions of stability. An important class of Runge-Kutta methods is the strong-stability preserving (SSP) class, which have the property that if a given stability condition is satisfied by a spatial discretization paired with the explicit Euler method in time, then the same stability condition is satisfied if an

SSP scheme is used in place of explicit Euler (possibly under a more restrictive CFL condition). That is, for any norm, seminorm, or convex functional $\|\cdot\|$, a scheme is SSP if:

$$\|q + \Delta t F(q)\| \leq \|q\| \; \forall \Delta t,\, 0 \leq \Delta t \leq \Delta t_E \Rightarrow \|q^{n+1}\| \leq \|q^n\| \; \forall \Delta t,\, 0 \leq \Delta t \leq c\Delta t_E \tag{1.41}$$

where $F(q) = dq/dt$ and c is some positive constant independent of $\Delta t, \Delta x$. Schemes can be designed to have this property by assembling them from convex combinations of explicit Euler steps:

$$\begin{aligned} u^{(1)} &= u^n \\ u^{(s)} &= u^n + \Delta t \left(\sum_{k=1}^{s-1} \alpha_{s,k} \left(u^{(k)} + \Delta t \frac{\beta_{s,k}}{\alpha_{s,k}} F(u^{(k)}) \right) \right), \; s = 2, \cdots, S \\ u^{n+1} &= u^{(S)} \end{aligned} \tag{1.42}$$

Since consistency requires $\sum_k \alpha_{s,k} = 1$, Eq. (1.42) is a convex combination of Euler steps (with varying time step sizes $\Delta t \beta_{s,k}/\alpha_{s,k}$) as long as all the $\alpha_{s,k}$ and $\beta_{s,k}$ are positive. The new time step restriction is found by:

$$\max_{s,k} \Delta t \frac{\beta_{s,k}}{\alpha_{s,k}} \leq \Delta t_E \Rightarrow c = \min_{i,k} \frac{\alpha_{s,k}}{\beta_{s,k}} \tag{1.43}$$

An SSP scheme is considered optimal if it has the largest possible constant c over all SSP Runge-Kutta schemes of a given order of accuracy. The most popular such scheme, which is also the one used in this work, is the third-order scheme:

$$\begin{aligned} u^{(1)} &= u^n + \Delta t F(u^n) \\ u^{(2)} &= \frac{3}{4} u^n + \frac{1}{4} \left(u^{(1)} + \Delta t F(u^{(1)}) \right) \\ u^{n+1} &= \frac{1}{3} u^n + \frac{2}{3} \left(u^{(2)} + \Delta t F(u^{(2)}) \right) \end{aligned} \tag{1.44}$$

SSP Runge-Kutta schemes were originated in [22], and further discussion can be found in e.g. [23], [24], [25]. The advantage of such schemes is that nonlinear stability properties may be available for simple time-advancement schemes such as explicit Euler, but not for more complicated high-order schemes. SSP schemes allow such properties to carry over to the more practical high-order time discretizations.

The second class of time discretizations is the natural continuous extension of Runge-Kutta schemes, originally developed by Zennaro [26] and the use of which in central schemes was originated by Bianco et al. in [27] and used in [28] and [29]. In a central scheme it is still necessary to evaluate the time-averaged fluxes Eq. (1.15), and in the absence of a semi-discrete form obtained by taking the limit $\Delta t \to 0$ then that time integral must be approximated by quadrature. Intermediate values of the solution can be obtained by a Runge-Kutta scheme applied to the differential form of the conservation law Eq. (1.1):

$$\frac{dq_j}{dt} = -\frac{\partial f(q)}{\partial x}\Big|_{x_j} \tag{1.45}$$

which after spatial discretization becomes:

$$\frac{dq_j}{dt} = F_j(t, q) \tag{1.46}$$

However the intermediate times at which Runge-Kutta stages are calculated may not coincide with convenient quadrature nodes. Thus we need to interpolate the solution within each time step using the information produced by the Runge-Kutta scheme and preferably no additional information. To enable comparisons with the schemes of [28] [30] [29] we employ the same fourth-order Runge-Kutta scheme used

there and which is given by:

$$
\begin{aligned}
u^{n+1} &= u^n + \Delta t \sum_{k=1}^{4} b_k F_k \\
F_1 &= F(u^n) \\
F_2 &= F\left(t^n + c_2 \Delta t, u^n + \frac{\Delta t}{2} F_1\right) \\
F_3 &= F\left(t^n + c_3 \Delta t, u^n + \frac{\Delta t}{2} F_2\right) \\
F_4 &= F\left(t^n + c_4 \Delta t, u^n + \Delta t F_3\right) \\
c_2 &= \frac{1}{2},\ c_3 = \frac{1}{2},\ c_4 = 1 \\
b_1 &= \frac{1}{6},\ b_2 = \frac{1}{3},\ b_3 = \frac{1}{3},\ b_4 = \frac{1}{6}
\end{aligned}
\tag{1.47}
$$

The natural continuous extension replaces the coefficients b_i with polynomials $b_i(\theta)$ to construct a polynomial $z(t)$ such that:

$$z(t^n + \theta \Delta t) = q^n + \Delta t \sum_{k=1}^{4} b_k(\theta) F_k,\ 0 \leq \theta \leq 1$$

$$z(t^n) = q^n \text{ and } z(t^{n+1}) = q^{n+1}$$

$$\max_{t^n \leq t \leq t^{n+1}} \left| q^{(r)}(t) - z^{(r)}(t) \right| = \mathcal{O}(\Delta t^{4-r}), 0 \leq r \leq 4$$

where $q(t)$ is the exact solution to Eq. (1.46). The polynomials $b_k(\theta)$ are:

$$b_1(\theta) = 2(1 - 4b_1)\theta^3 + 3(3b_1 - 1)\theta^2 + \theta$$

$$b_k(\theta) = 4(3c_k - 2)b_k\theta^3 + 3(3 - 4c_k)b_k\theta^2,\ k = 2, 3, 4$$

Since $z(t)$ uniformly approximates the solution to Eq. (1.45) within $0 \leq \theta \leq 1$ it can be evaluated at the quadrature points in order to accurately approximate the time integrals in Eq. (1.13). If the quadrature points for the time integrals are known in advance, then the values of $b_i(\theta)$ can be precalculated. In this work we

apply Simpson's rule to the continuous extension:

$$\frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} f(q(x_j,t))\,dt = \frac{1}{6}(f(q(x_j,t^n))+4f(q(x_j,t^n+\Delta t/2))+f(q(x_j,t^{n+1})))+\mathcal{O}(\Delta t^4) \tag{1.48}$$

Unfortunately, the optimal SSP schemes do not have natural continuous extensions with order of accuracy equal to that of their natural continuous extensions, which makes them unsuitable for purposes of the central schemes in this work.

1.6 Characteristic Variables

Oscillations can arise if the reconstruction itself is oscillatory or if the scheme is unstable so that small perturbations are amplified. Using a WENO algorithm prevents oscillations from the first source but achieves nothing in regard to the latter. For simple problems such as the linear advection of a scalar given by:

$$\frac{\partial q}{\partial t} + a\frac{\partial q}{\partial x} = 0 \tag{1.49}$$

(where a is a constant) oscillations of the second kind can be avoided by biasing the reconstruction stencil in the direction from which information propagates - the so-called upwind direction. If $a > 0$, so that the solution at a point $x = x_0$ depends only on the values of the solution for $x \leq x_0$, then the reconstruction scheme for $q_{j+1/2}$ should involve more points to the left of $x_{j+1/2}$ than to its right (with the directions reversed if $a < 0$). For systems of equations this condition becomes more complicated. Consider linear advection as in Eq. (1.49) but with the constant scalar a replaced by a constant square matrix A:

$$\frac{\partial q}{\partial t} + A\frac{\partial q}{\partial x} = 0 \tag{1.50}$$

We assume that this system is hyperbolic, thus A is real-diagonalizable: $A = X\Lambda X^{-1}$. We may then multiply Eq. (1.50) on the left by X^{-1} to write:

$$X^{-1}\frac{\partial q}{\partial t} + \Lambda X^{-1}\frac{\partial q}{\partial x} = 0 \tag{1.51}$$

Since X^{-1} is constant we may rewrite the original system in terms of the so-called characteristic variables $\xi = X^{-1}q$:

$$\frac{\partial \xi}{\partial t} + \Lambda\frac{\partial \xi}{\partial x} = 0 \tag{1.52}$$

Because the eigenvalue matrix Λ is diagonal, each row of Eq. (1.52) is an independent *scalar* advection equation. The reconstruction for each component should have the appropriate upwind bias, and then the physical variables q can be recovered from ξ.

This derivation does not apply in the case of a nonlinear system. We may linearize the system Eq. (1.1) to obtain:

$$\frac{\partial q}{\partial t} + J(q)\frac{\partial q}{\partial x} = 0 \tag{1.53}$$

where $J(q)$ is the flux Jacobian matrix, and then diagonalize $J(q)$ as before. However, the matrix $X(q)^{-1}$ of left eigenvectors now depends on q and therefore cannot be brought inside the derivatives to define new variables ξ. In practice, however, taking $X(q)^{-1}$ to be locally constant within each stencil usually produces non-oscillatory solutions despite the non-rigorous mathematical justification for doing so.

1.7 Drawbacks of Upwind CRWENO

With the theoretical and numerical background now in hand we can begin to see serious deficiencies in the upwind CRWENO method, all of which connect to the

use of characteristic variables. First, performing a compact reconstruction in characteristic variables leads to a block-tridiagonal matrix as opposed to the tridiagonal matrices obtained from a componentwise reconstruction. Indeed, consider compact schemes of the form Eq. (1.31) combined in a CRWENO scheme that couples the unknowns for component c in cells $j-1, j, j+1$:

$$\begin{aligned}\sum_{s=1}^{S} \omega_{cs}(\bar{q}) \left(\sum_{v=-1}^{1} L_{sv} \hat{Q}_{j+v} \right) &= \sum_{s=1}^{S} \omega_{cs}(\bar{q}) \left(\sum_{v} R_{sv} \bar{q}_{j+v} \right) \\ \sum_{v=-1}^{1} \left(\sum_{s=1}^{S} \omega_{cs}(\bar{q}) L_{sv} \right) \hat{Q}_{j+v} &= \sum_{v} \left(\sum_{s=1}^{S} \omega_{cs}(\bar{q}) R_{sv} \right) \bar{q}_{j+v}\end{aligned} \tag{1.54}$$

where S is the number of subschemes and $\hat{Q}$ is the quantity of interest. For each component c of a system of conservation laws Eq. (1.54) produces an independent tridiagonal system if different weights are used for each component. On the other hand, applying Eq. (1.54) to the characteristic variables $\xi = X^{-1} q$ gives:

$$\sum_{v=-1}^{1} \sum_{c} \left(\sum_{s=1}^{S} \omega_{fs}(\bar{\xi}) L_{sv} \right) X_{fc}^{-1} \hat{Q}_{c,j+v} = \sum_{v,c} \left(\sum_{s=1}^{S} \omega_{fs}(\bar{\xi}) R_{sv} \right) X_{fc}^{-1} \bar{q}_{c,j+v} \tag{1.55}$$

Thus each value of v corresponds to a block of the form $\sum_s \Omega^{(s)} L_{sv} X^{-1}$, where $\Omega^{(s)}$ is a diagonal matrix whose cth entry on the diagonal is the weight for subscheme s computed from the cth component of the characteristic variables $\bar{\xi}$. Because the left eigenvector matrix combines all components of the physical variables q, this system cannot be decoupled into independent systems in a componentwise fashion and the prefactor $\Omega^{(s)}$ prevents simplification by multiplying on the left by X. In comparison, for non-compact WENO the analog of Eq. (1.55) is:

$$\sum_{c} X_{fc}^{-1} \hat{Q}_{c,j+v} = \sum_{v,c} \left(\sum_{s=1}^{S} \omega_{fs}(\bar{\xi}) R_{sv} \right) X_{fc}^{-1} \bar{q}_{c,j+v} \tag{1.56}$$

which becomes after multiplying on the left by X:

$$\hat{Q}_{g,j} = \sum_{v,c,f} \left(\sum_{s=1}^{S} X_{gf} \omega_{fs}(\bar{\xi}) R_{sv} \right) X^{-1}_{fc} \bar{q}_{c,j+v} \tag{1.57}$$

For each v, the solution vector $\bar{q}_{j+v}$ in cell $j+v$ is multiplied by the matrix $X(\sum_s \Omega^{(s)} R_{sv}) X^{-1}$, which in general is not diagonal. Thus although the components of the reconstruction $\hat{Q}$ depend on all the components of $\bar{q}$, each component can be calculated independently whereas in the compact reconstruction the components of $\hat{Q}$ are coupled.

The CRWENO scheme Eq. (1.32) of [18] has a smaller leading-order truncation error coefficient than the classical WENO scheme of [15], both of which are fifth-order accurate [17]. Thus on a given grid one would expect the solution from CRWENO to have a smaller error. On the other hand, the preceding discussion strongly suggests that the computational expense for one CRWENO reconstruction is greater than that for a WENO reconstruction, especially when characteristic variables are used. It is therefore not clear whether CRWENO is superior to WENO in terms of efficiency, i.e. the time required to obtain a solution with a given level error. As Figure 1.4 shows, a simple test case of a sinusoidal density wave which advects with constant velocity under the Euler equations shows that CRWENO is in fact slightly less efficient than WENO when characteristic variables are used whereas CRWENO slightly outperforms WENO when the conserved variables are

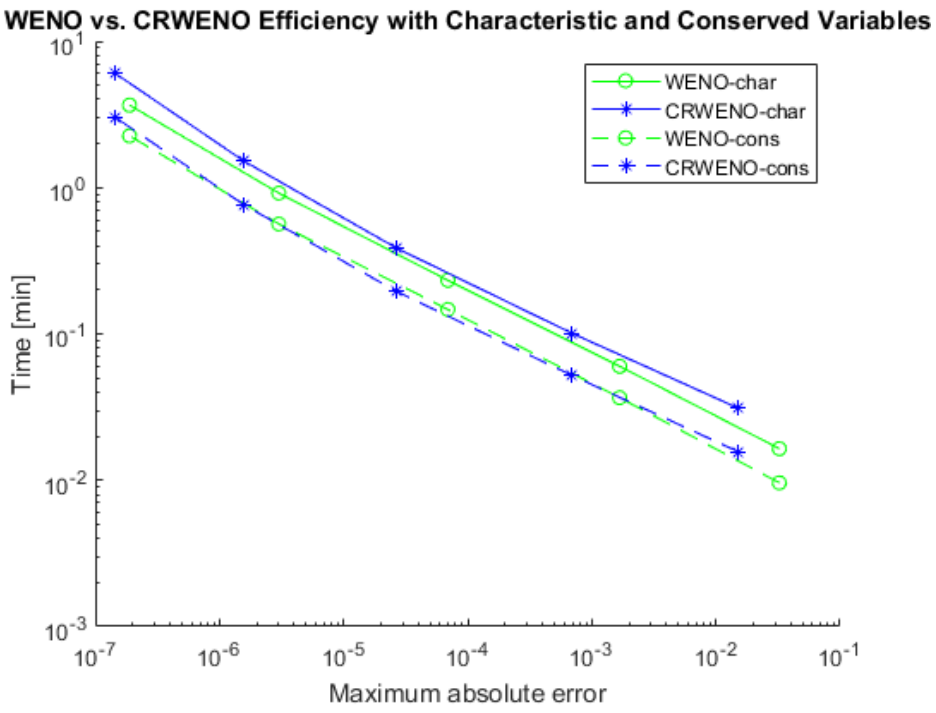

Figure 1.4: Characteristic variables increase the computational expense and WENO slightly outperforms CRWENO in efficiency when characteristic variables are used.

reconstructed directly.

$$\begin{bmatrix} \rho(x,0) \\ u(x,0) \\ P(x,0) \end{bmatrix} = \begin{bmatrix} 1+0.2\sin(4\pi x) \\ 1 \\ 1 \end{bmatrix}, \quad 0 \le x \le 2 \tag{1.58}$$

Though the efficiency difference is small enough to be potentially sensitive to implementation and hardware details, the fact that the efficiency gains obtained by reducing the error evaporate as a result of using characteristic variables indicates an opportunity for improvement. We also see that characteristic variables increase the CRWENO computation time by about 98% whereas the corresponding increase

with WENO reconstructions is only about 64%.

The mathematical justification for characteristic variables loses rigor when considering nonlinear systems of conservation laws. As previously mentioned, the standard strategy for treating nonlinear system is to use the flux jacobian evaluated at some reference state that differs between stencils but is fixed within each stencil. This strategy raises two troublesome questions. First, how should the reference state be chosen? One might choose the state of the central cell, or a simple average of the states on either side of a relevant interface, or, which seems in practice to be the most robust, the density-weighted average of those two states as done in the Roe flux difference splitting method [7]. A method may fail on some test problem with one such choice but succeed with another, and one would prefer a method that does not require such tinkering in order to function.

The second problem is that the assumption that the transformation matrix X^{-1} is locally constant does not hold when the stencil includes a shock. Near shocks this breakdown can lead to nonphysical values, such as negative density or pressure, and this problem occurs frequently in compact schemes because the transformation in one stencil influences the solution in nearby stencils. Yet some kind of characteristic transformation is still necessary in order to prevent oscillations that would arise from incorrect upwinding, so one must tinker with the reconstruction algorithm itself to produce one that functions properly with the characteristic transformation. Of course, that work could easily go to waste if the test problem changes.

Finally, although it is difficult to obtain any analytical results concerning the stability or other relevant properties of WENO methods the added complication of

the characteristic transformation amplifies that difficulty immensely. Furthermore, because the characteristic variables derive from the flux jacobian a result for one flux might not hold for general fluxes which also limits the applicability of any analytical result. One would prefer, therefore, to have a method that can be applied componentwise and, moreover, that is as simple and general as possible.

The upwind CRWENO method also inherits some drawbacks from its nature as an upwind scheme. The use of a Riemann solver to determine the numerical flux at an interface from the reconstructions on each side of the face necessarily introduces all of the problems associated with Riemann solvers discussed in Section 1.3.1. The one-dimensional upwind framework benefits from the clear division of waves into left- and right-going families which is not available in multiple dimensions. In practice, it is common to perform one-dimensional reconstructions along coordinate directions and treat each interface as a one-dimensional Riemann problem. In reality, however, a two-dimensional Riemann problem (in which four different states meet at a single point) exhibits fundamentally two-dimensional behavior examples of which can be seen in [31]. The coordinate-by-coordinate approach can nevertheless produce reasonable results at the cost of a small time step, whereas a two-dimensional Riemann solver can allow larger time steps [32]. Of course, all of these problems would be avoided by using a method that does not require a Riemann solver and that naturally extends to multiple dimensions.

1.8 Purpose and Outline of Thesis

The upwind CRWENO method has serious drawbacks due to the use of characteristic variables and the upwind framework. The purpose of this thesis is to rehabilitate the idea of compact WENO reconstructions in such a way that characteristic variables are not required but that does not sacrifice accuracy or the ability to efficiently resolve high-frequency features, and that produces a method that is simple, so that it can be fruitfully analyzed, yet robust so that it applies to a wide range of problems without problem-specific tinkering with parameters.

Upwind schemes fail when the directions of information propagation are not respected so in practice upwind schemes require use of characteristic variables, especially when the reconstruction is high-order. Therefore discarding characteristic variables requires an entirely different discretization strategy. The central schemes provide such an alternative, and have had success at dispensing with characteristic variables [10] [28]. Levy et al. constructed central WENO schemes in which the necessary quantities are reconstructed by WENO schemes of third and fourth order [28], then extended to multidimensional schemes of third [33] and fourth [30] order all without need for characteristic variables (note the different usage of the term *compact* in [33], where it is used to mean that the reconstruction stencil involves few cells rather than in the sense of [19], [18], and the present work where it is used to mean that the reconstruction is spatially implicit). Qiu and Shu in [29], however, found that when the central WENO framework of Levy et al. is used to construct fifth or ninth-order central WENO schemes the results can be oscillatory unless charac-

teristic decomposition is used in at least the subcell reconstruction. The authors of [29] give no explanation, however, as to why the characteristic decomposition is needed at all except to suggest that the high order of accuracy is at fault, let alone why it is sufficient to perform the decomposition only in the subcell reconstruction given that all the reconstructions are high-order. Furthermore, although the fourth-order central WENO scheme (which we will refer to as CWENO4 for brevity) of in [28] is developed by forming a polynomial within each cell, when translated into a finite-difference formula for the subcell average it becomes equivalent to the fifth-order subcell reconstruction in [29]. Both methods use the same fourth-order Runge-Kutta method with its natural continuous extension, so it would seem that the nature of the point value reconstruction influences the need for characteristic variables in some way.

A plausible candidate for the relevant difference is that, because it directly forms a polynomial reconstruction in each cell, CWENO4 uses the same polynomial to determine the subcell averages and the point value. The fifth-order scheme of Qiu and Shu [29] (CWENO5), although it does not explicitly involve polynomials, cannot be equivalent to any polynomial-derived scheme because the ideal weights in its subcell and point value reconstructions are different. We conjecture that this inconsistency manifests as oscillatory behavior which the characteristic decomposition serves to quell. This hypothesis would also explain why only high-order schemes appear to require characteristic variables, since only lower-order schemes can have the subcell and point value reconstructions arise from the same polynomial. Using a compact scheme prevents the direct formation of a reconstruction polynomial, so

we aim for the similar goal of designing a scheme that uses the same ideal weights for the subcell and point value reconstructions.

The remainder of this work is structured as follows. Chapter 2 describes the design of a method intended to meet the goals stated in this section and Chapter 3 discusses the quality of the solutions to various test problems obtained with the resulting method. That discussion will reveal a significant drawback to the scheme of Chapter 2, which is described and investigated in Chapter 4. Chapter 5 summarizes the project and avenues for future work.

Chapter 2: A Central CRWENO Method

In this chapter we enumerate the properties desired for a method that achieves the goals discussed in the introduction. These properties will lead to constraints on the spatial reconstruction scheme, and geometric considerations will determine the modifications required at boundaries. As central schemes have had success with avoiding characteristic transformations [28] [29] [34] and avoid Riemann solvers by construction, the development of the new method will use the framework of central schemes. We will start by deriving the one-dimensional method, which will turn out to be easily extended to multiple dimensions. Then we will conclude with numerical analysis of the method.

2.1 Design Requirements

Figure 2.1 diagrams one time step of a central scheme, which computes cell averages on the staggered grid at the next time step. Repeating the process produces cell averages on the main grid. Three quantities need to be reconstructed:

1. Subcell averages from cell averages

2. Midpoint values from cell averages

3. Derivatives from point values

A fully compact scheme would perform each of these reconstructions compactly. Because each compact reconstruction involves calculation of the nonlinear weights and solving a linear system, a central CRWENO method would appear to require three such processes whereas the upwind CRWENO requires only one. Some of this expense can be avoided, however, by designing the schemes to use the same ideal weights. Since the smoothness indicators are calculated from the same solution for each reconstruction, if the ideal weights are also identical then so will be the nonlinear weights which would only need to be calculated once. A one-parameter family of such weight-linked schemes exists, for which the subschemes for the left subcell averages are:

$$\begin{aligned}
\frac{5}{8}\bar{q}^L_{j-1} + \frac{3}{8}\bar{q}^L_j &= \frac{1}{64}\bar{q}_{j-2} + \frac{13}{32}\bar{q}_{j-1} + \frac{5}{64}\bar{q}_j \\
\frac{3}{16}\bar{q}^L_{j-1} + \frac{5}{8}\bar{q}^L_j + \frac{3}{16}\bar{q}^L_{j+1} &= \frac{5}{32}\bar{q}_{j-1} + \frac{5}{16}\bar{q}_j + \frac{1}{32}\bar{q}_{j+1} \\
\frac{3}{8}\bar{q}^L_j + \frac{5}{8}\bar{q}^L_{j+1} &= \frac{19}{64}\bar{q}_j + \frac{7}{32}\bar{q}_{j+1} - \frac{1}{64}\bar{q}_{j+2}
\end{aligned} \tag{2.1}$$

and the subschemes for point values u are:

$$\begin{aligned}
(1-s_1)q_{j-1} + s_1 q_j &= \frac{-1}{24}\bar{q}_{j-2} + \left(\frac{13}{12} - s_1\right)\bar{q}_{j-1} + \left(\frac{-1}{24} + s_1\right)\bar{q}_j \\
\left(\frac{9}{80} + s_2\right)q_{j-1} + \left(\frac{31}{40} - 2s_2\right)q_j + \left(\frac{9}{80} + s_2\right)q_{j+1} &= \\
\left(\frac{17}{240} + s_2\right)\bar{q}_{j-1} + \left(\frac{103}{120} - 2s_2\right)\bar{q}_j + \left(\frac{17}{240} + s_2\right)\bar{q}_{j+1} & \\
s_1 q_j + (1-s_1)q_{j+1} &= \left(\frac{-1}{24} + s_1\right)\bar{q}_j + \left(\frac{13}{12} - s_1\right)\bar{q}_{j+1} + \frac{-1}{24}\bar{q}_{j+2}
\end{aligned} \tag{2.2}$$

where the parameters s_1 and s_2 are related by:

$$s_2 = \frac{s_1}{10} + \frac{9}{400} \tag{2.3}$$

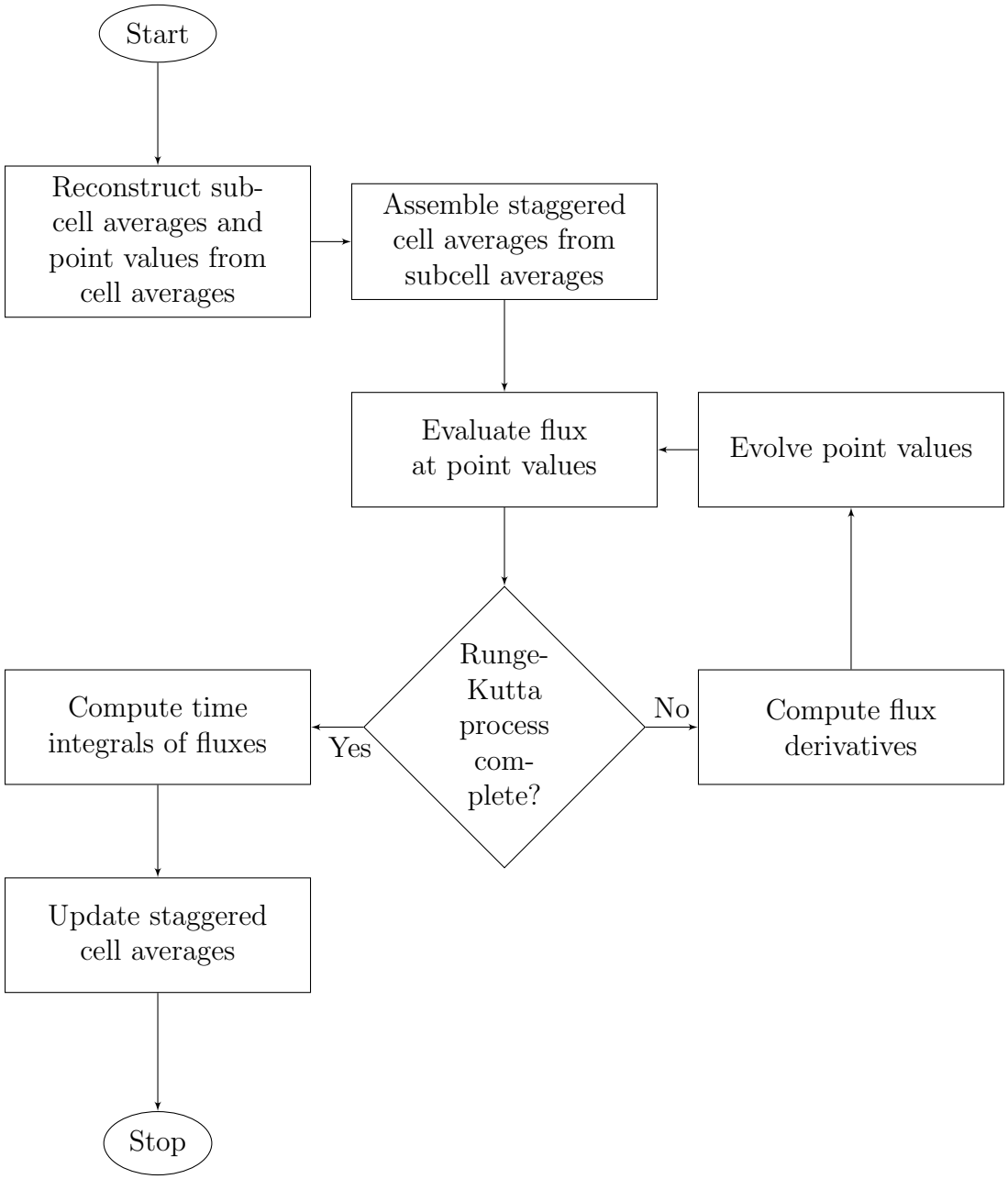

Figure 2.1: Flowchart of one time step of a central scheme.

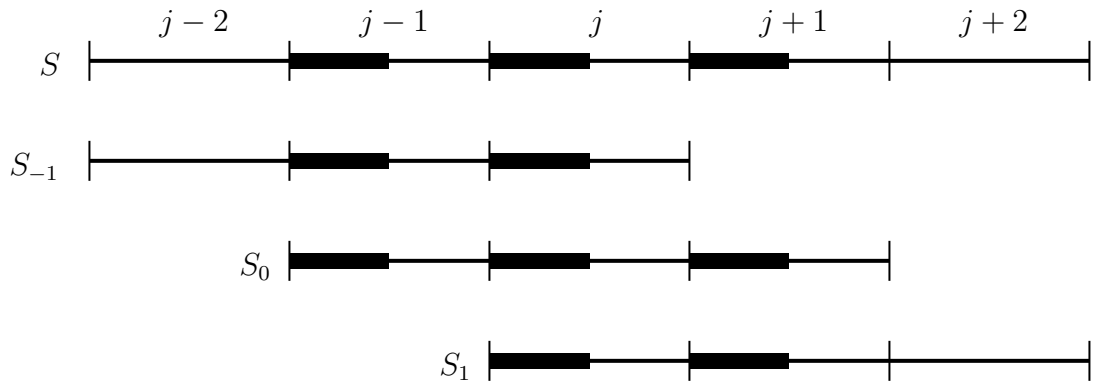

Figure 2.2: Central CRWENO stencils for the left subcell reconstruction. Solid rectangles indicate subcells.

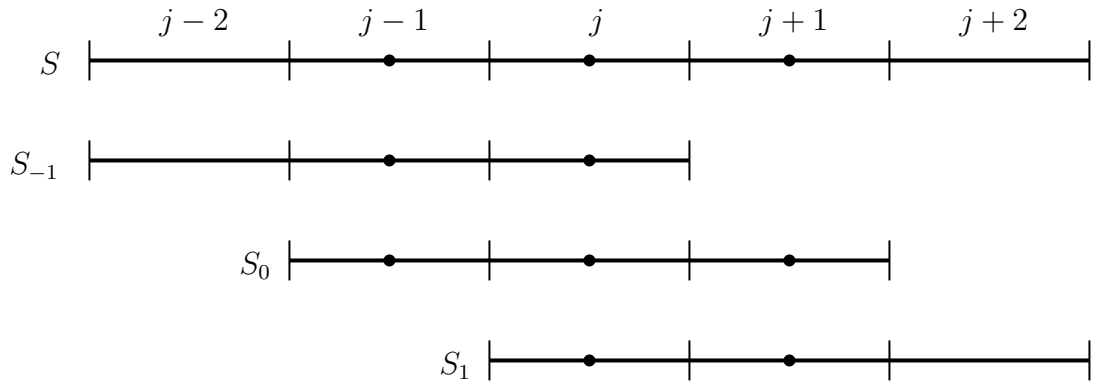

Figure 2.3: Central CRWENO stencils for the point value reconstruction.

Figure 2.2 shows the cells and subcells involved in each subscheme, and Figure 2.3 shows the cells and point values involved. Note that wherever a subscheme involves a subcell average, the corresponding subscheme for the point value involves the point value in the same cell and vice versa. S_{-1}, S_0, and S_1 are the subschemes that combine to form the high-order scheme S.

In both reconstructions the ideal weights are $\bar{\omega}_{-1} = 1/12, \bar{\omega}_0 = 5/6, \bar{\omega}_1 = 1/12$. Though the overall scheme formed according to the ideal weights has $\mathcal{O}(\Delta x^5)$ formal

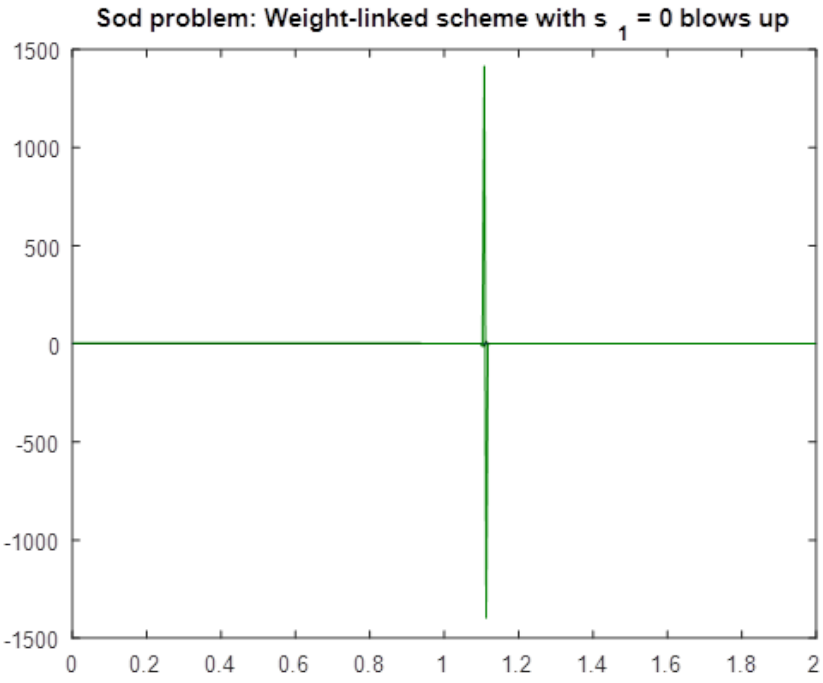

Figure 2.4: $s_1 = 0$ causes the solution to the Sod problem to blow up.

accuracy regardless of s_1, the choice of s_1 greatly affects the quality of the results when the solution is not smooth. For example, when $s_1 = 0$ the method blows up when applied to the Sod problem [35] as shown in Figure 2.4. If $s_1 \geq 0.5$, however, the solution is stable but creates spurious oscillations after the contact discontinuity shown in Figure 2.5.

The $s_1 = 0$ variant fails so spectacularly because the WENO process selects (i.e. assigns weights of $\mathcal{O}(1)$ to) only the first or only the third stencil of (2.2), which do not involve q_j at all when $s_1 = 0$. The coefficient matrix in the system for the point values therefore has a 2x2 block on the diagonal with entries that are essentially zero. While the blocks associated with the regions to either side of the

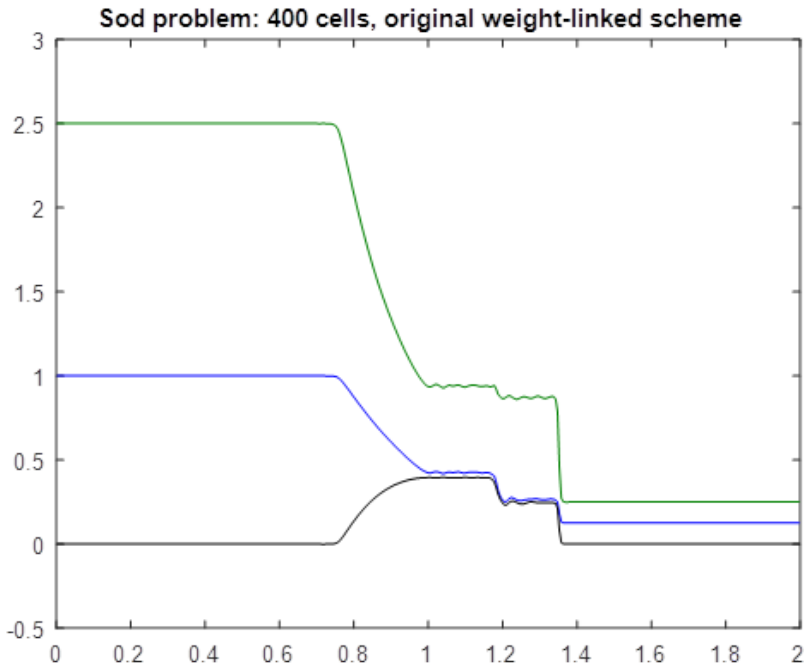

Figure 2.5: $s_1 = 0.6$ eliminates the fatal instability but does not eliminate all oscillations.

discontinuity are still essentially decoupled, they are far from diagonally dominant. When $s_1 \geq 0.5$, however, we see from (2.2) that the first and third stencils have coefficients of q_j that are greater than those of $q_{j\pm1}$, which causes the blocks of the coefficient matrix to be diagonally dominant near shocks. We note from (2.1) that the first and third subschemes in the subcell-average reconstruction are not diagonally dominant, which explains the presence of small oscillations after the instability of Figure 2.4 is addressed by imposing diagonal dominance. Furthermore, we observe that the high-frequency oscillations present in Figure 2.5 do not appear when the problem is solved by CWENO4 which is a spatially explicit method i.e. the coefficient matrix is the identity which is as diagonally dominant as a matrix can be. Lastly, diagonal dominance guarantees that the coefficient matrix is invertible. Therefore we require the designed scheme to be diagonally dominant for all possible weight combinations. Equivalently, we require each individual subscheme to be diagonally dominant in the sense that the coefficient of the unknown at cell j exceeds the sum of absolute values of all other coefficients.

Because the coefficients of the reconstruction depend on the solution itself a direct reconstruction of the staggered cell averages cannot be guaranteed to maintain conservation. Instead, we reconstruct the half-averages over the left subcells

$$\bar{q}_j^L = \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_j} u(x)\, dx \tag{2.4}$$

and then compute the half-averages over right subcells by conservation as follows:

$$\bar{q}_j^R = \frac{1}{\Delta x} \int_{x_j}^{x_{j+1/2}} u(x)\, dx = \bar{q}_j - \bar{q}_j^L \tag{2.5}$$

For convenience and to simplify the discussion of the multidimensional extension, *subcell average* will refer to these integrals that are normalized by the cell volume, which in one dimension would be more properly referred to as subcell half-averages. In effect, we reconstruct the average over a whole cell of the solution multiplied by the indicator function of the subcell. Consider a CRWENO scheme for the left subcell:

$$\sum_{s,k} \omega_s(\sigma) L_{sk} \bar{q}^L_{j+k} = \sum_{s,k} \omega_s(\sigma) R_{sk} \bar{q}_{j+k} \tag{2.6}$$

Here σ is the ordered set containing the union of all cell indices involved in the subschemes in increasing order: $\sigma = \{j_1, j_2, \cdots, j_m\}$. There are two ways to obtain the averages over right subcells. First, one may use the conservation approach in which Eq. (2.5) is substituted into Eq. (2.6), giving the following scheme for the right subcell:

$$\begin{aligned}
\sum_{s,k} \omega_s(\sigma) L_{sk} (\bar{q}_{j+k} - \bar{q}^R_{j+k}) &= \sum_{sk} \omega_s(\sigma) R_{sk} \bar{q}_{j+k} \\
-\sum_{s,k} \omega_s(\sigma) L_{sk} \bar{q}^R_{j+k} &= \sum_{s,k} \omega_s(\sigma) (R_{sk} - L_{sk}) \bar{q}_{j+k} \\
\sum_{s,k} \omega_s(\sigma) L_{sk} \bar{q}^R_{j+k} &= \sum_{s,k} \omega_s(\sigma) (L_{sk} - R_{sk}) \bar{q}_{j+k}
\end{aligned} \tag{2.7}$$

Here s indexes the subschemes. Alternatively, one may simply reflect Eq. (2.6) about the center of cell j to obtain:

$$\sum_{s,k} \omega_s(-\sigma) L_{s,-k} \bar{q}^R_{j+k} = \sum_{s,k} \omega_s(-\sigma) R_{s,-k} \bar{q}_{j+k} \tag{2.8}$$

Where if $\sigma = \{j_1, j_2, \cdots, j_m\}$ then $-\sigma$ is the reflection of σ through the point j: $-\sigma = \{j - j_m, j - j_{m-1}, \cdots, j - j_1\}$. Note the reversed order. For the subcell reconstruction to be consistent, meaning that it does not matter whether one reconstructs

the left subcell and obtains the right subcell by conservation or vice versa, we need for Eq. (2.7) and Eq. (2.8) to be identical for all possible values of the nonlinear weights ω_s. We may write the nonlinear weights ω_s in terms of the corresponding ideal weights $\bar{\omega}_s$ and functions $f_s(\sigma)$ that depend only on the solution $\bar{q}_j, j \in \sigma$.

$$\omega_s(\sigma) = \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)}, \quad \omega_s(-\sigma) = \frac{\bar{\omega}_s f_s(-\sigma)}{\sum_r \bar{\omega}_r f_r(-\sigma)} \tag{2.9}$$

We then obtain a condition on the left-hand side coefficients of the subschemes:

$$\begin{aligned} \sum_s \omega_s(\sigma) L_{sk} &= \sum_s \omega_s(-\sigma) L_{s,-k} \\ \sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)} L_{sk} &= \sum_s \frac{\bar{\omega}_s f_s(-\sigma)}{\sum_r \bar{\omega}_r f_r(-\sigma)} L_{s,-k} \end{aligned} \tag{2.10}$$

The function $f_s(\sigma)$ measures the smoothness in the stencil of subscheme s, and therefore must have the property that its value is unchanged after reflecting the stencil σ and interchanging the subscheme index: $f_s(-\sigma) = f_{-s}(\sigma)$. From this fact we can simplify Eq. (2.10):

$$\sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)} L_{sk} = \sum_s \frac{\bar{\omega}_s f_{-s}(\sigma)}{\sum_r \bar{\omega}_r f_{-r}(\sigma)} L_{s,-k} \tag{2.11}$$

Re-indexing the sums on the right-hand side:

$$\sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)} L_{sk} = \sum_s \frac{\bar{\omega}_{-s} f_s(\sigma)}{\sum_r \bar{\omega}_{-r} f_r(\sigma)} L_{-s,-k} \tag{2.12}$$

For Eq. (2.12) to hold regardless of the values of $f_s(\sigma)$ requires $\bar{\omega}_s = \bar{\omega}_{-s}$ and $L_{sk} = L_{-s,-k}$ for all s and k. That is, the ideal weights must be symmetric and the array of left-hand side coefficients must be rotationally symmetric. As a consequence, we have for the combined scheme:

$$\sum_s \bar{\omega}_s L_{sk} = \sum_s \bar{\omega}_{-s} L_{-s,-k} = \sum_s \bar{\omega}_s L_{s,-k} \tag{2.13}$$

after reindexing the sum. Therefore the left-hand side of the combined scheme must also be symmetric. From a numerical perspective it is beneficial to scale the rows so that they have equal sums so we also require the normalization condition $\sum_k L_{sk} = 1$ for each s.

The equivalence of Eq. (2.7) and Eq. (2.8) also implies a condition on the right-hand side coefficients:

$$\begin{aligned}
\sum_s \omega_s(-\sigma) R_{s,-k} &= \sum_s \omega_s(\sigma)(L_{sk} - R_{sk}) \\
\sum_s \frac{\bar{\omega}_s f_s(-\sigma)}{\sum_r \bar{\omega}_r f_r(-\sigma)} R_{s,-k} &= \sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)}(L_{sk} - R_{sk}) \\
\sum_s \frac{\bar{\omega}_{-s} f_s(\sigma)}{\sum_r \bar{\omega}_{-r} f_r(\sigma)} R_{-s,-k} &= \sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)}(L_{sk} - R_{sk})
\end{aligned} \tag{2.14}$$

We already know that $\bar{\omega}_s = \bar{\omega}_{-s}$, so:

$$\sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)} R_{-s,-k} = \sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)}(L_{sk} - R_{sk}) \tag{2.15}$$

which implies the condition:

$$R_{-s,-k} + R_{sk} = L_{sk} \tag{2.16}$$

The ideal weights $\bar{\omega}_s$ depend on the truncation error expansions of the subschemes and may be negative, so that the resulting combination of subschemes is not a convex combination. This loss of convexity can cause instability so it is desirable to have subschemes with positive ideal weights. If negative weights are unavoidable, the splitting procedure of [36] can be applied to obtain a stable method, but that procedure involves performing two sets of reconstructions with different ideal weights. Because the compact reconstructions can be expensive we prefer to minimize the number that need to be performed, so we will aim for positive ideal weights

that sum to 1. We can arrange to solve even fewer linear systems if, in addition to matching the ideal weights for the subcell and point value reconstructions, the LHS coefficients of the individual subschemes can also be matched. This would cause the coefficient matrices for the two reconstructions to be identical, with only the RHS differing to produce the reconstructed subcell averages and point values. As a result, instead of forming and solving two different coefficient matrices with different right-hand sides, we assemble only one system and apply it to the two right-hand sides.

To summarize, the conditions for the central CRWENO scheme are:

1. The subcell and point value subschemes must use the same ideal weights.
2. The ideal weights must be symmetric.
3. The ideal weights must be positive and sum to 1.
4. The left-hand side of each subscheme must be diagonally dominant.
5. The left-hand side coefficients of each subscheme for the point value must match those of the corresponding subscheme for the subcell average.
6. The left-hand side coefficients must satisfy $L_{sk} = L_{-s,-k}$.
7. The left-hand side coefficients must sum to 1.
8. The entire method must be at least fifth-order accurate (in order to be competitive with CRWENO).

2.2 Spatial Reconstructions

The foregoing discussion has not imposed any constraints on the number of subschemes or their stencils except that the stencil for subscheme $-s$ must be the reflection of that of subscheme s. In the following derivation of the central CRWENO (CCRWENO) method we will derive a three-subscheme method that uses the same stencils as the classical fifth-order WENO method of [15]. First, we constrain the left-hand side (LHS) coefficients and then use the results to constrain the right-hand side (RHS) coefficients.

2.2.1 LHS Coefficients

Let L_{sk} be the array of LHS coefficients of the subschemes $s = -1, 0, 1$:

$$L_{sk} = \begin{bmatrix} L_{-1,-1} & L_{-1,0} & L_{-1,1} \\ L_{0,-1} & L_{0,0} & L_{0,1} \\ L_{1,-1} & L_{1,0} & L_{1,1} \end{bmatrix} \tag{2.17}$$

To avoid the Gibbs phenomenon at discontinuities we need the coefficient matrix to automatically decouple at discontinuities, which means that no subscheme can involve a cell on its LHS that does not appear in the RHS stencil. Since the RHS for subscheme $s = -1$ does not include cell $j + 1$ and the RHS for subscheme for $s = 1$ does not include cell $j - 1$, we must have $L_{-1,1} = L_{1,-1} = 0$. In conjunction with the rotational symmetry constraint $L_{sk} = L_{-s,-k}$ and the normalization $\sum_k L_{sk} = 1$

this condition implies that L_{sk} must take the form:

$$L_{sk} = \begin{bmatrix} \frac{1-d_1}{2} & \frac{1+d_1}{2} & 0 \\ \frac{1-d_0}{4} & \frac{1+d_0}{2} & \frac{1-d_0}{4} \\ 0 & \frac{1+d_1}{2} & \frac{1-d_1}{2} \end{bmatrix} \tag{2.18}$$

for some parameters d_1, d_0. Diagonal dominance is achieved whenever $d_1 > 0$ and $d_0 > 0$.

2.2.2 RHS Coefficients

Let the RHS coefficients for the subcell-average subschemes be given by:

$$R_{sk}^{SA} = \begin{bmatrix} R_{-1,-2} & R_{-1,-1} & R_{-1,0} & 0 & 0 \\ 0 & R_{0,-1} & R_{0,0} & R_{0,1} & 0 \\ 0 & 0 & R_{1,0} & R_{1,1} & R_{1,2} \end{bmatrix} \tag{2.19}$$

The second subcell equivalence condition Eq. (2.16) implies that R^{SA} must have the form:

$$R_{sk}^{SA} = \begin{bmatrix} -R_{1,2} & \frac{1-d_1}{2} - R_{1,1} & \frac{1+d_1}{2} - R_{1,0} & 0 & 0 \\ 0 & \frac{1-d_0}{4} - R_{0,1} & \frac{1+d_0}{4} & R_{0,1} & 0 \\ 0 & 0 & R_{1,0} & R_{1,1} & R_{1,2} \end{bmatrix} \tag{2.20}$$

The free parameters R_{sk} can be expressed in terms of the diagonal excesses d_0, d_1 by requiring each subscheme to be at least third-order accurate, which gives:

$$R^{SA} = \begin{bmatrix} \frac{-d_1}{16} & \frac{3-d_1}{8} & \frac{3d_1+2}{16} & 0 & 0 \\ 0 & \frac{3-2d_0}{16} & \frac{1+d_0}{4} & \frac{1-2d_0}{16} & 0 \\ 0 & 0 & \frac{6+5d_1}{16} & \frac{1-3d_1}{8} & \frac{d_1}{16} \end{bmatrix} \tag{2.21}$$

Because the ideal weights are symmetric and sum to 1, we have that $\bar{\omega}_{-1} = \bar{\omega}_1 = (1 - \bar{\omega}_0)/2$. The truncation error coefficients of the combined scheme in terms of $\bar{\omega}_0, d_0, d_1$ are shown in Table 2.1

Table 2.1: Truncation error coefficients of the subcell reconstruction.

Power	Δx^3	Δx^4	Δx^5	Δx^6
Coefficient	$\frac{3\bar{\omega}_0}{16}(d_0 + d_1) - \frac{3}{16}\left(d_1 + \frac{1}{4}\right)$	0	$\frac{5\bar{\omega}_0}{64}(5d_0 + 23d_1) - \frac{5}{64}(23d_1 + 2)$	0

A similar analysis provides the RHS coefficients for third-order subschemes for the point value:

$$R^{PV} = \begin{bmatrix} \frac{-1}{24} & \frac{7-6d_1}{12} & \frac{11+12d_1}{24} & 0 & 0 \\ 0 & \frac{5-6d_0}{24} & \frac{7+6d_0}{12} & \frac{5-6d_0}{24} & 0 \\ 0 & 0 & \frac{11+12d_1}{24} & \frac{7-6d_1}{12} & \frac{-1}{24} \end{bmatrix} \tag{2.22}$$

Table 2.2 shows the truncation error coefficients for the combined point-value reconstruction:

Table 2.2: Truncation error coefficients of the point value reconstruction.

Power	Δx^3	Δx^4	Δx^5	Δx^6
Coefficient	0	$\frac{\bar{\omega}_0}{4}(d_0 - d_1 - 2) + \frac{d_1}{4} + \frac{29}{80}$	0	$\frac{\bar{\omega}_0}{32}(23d_0 - 23d_1 - 100) + \frac{23d_1}{32} + \frac{73}{28}$

To attain the desired fifth-order accuracy we need to choose $\bar{\omega}_0, d_0, d_1$ to cancel at least the Δx^3 error term in the subcell reconstruction:

$$\bar{\omega}_0(d_0 + d_1) = d_1 + \frac{1}{4} \tag{2.23}$$

and the Δx^4 term in the point value reconstruction:

$$\bar{\omega}_0(2 - d_0 + d_1) = d_1 + \frac{29}{20} \tag{2.24}$$

The system Eqs. (2.23)-(2.24) has the solution:

$$d_0 = \frac{5 + 8d_1}{17 + 20d_1}, \quad \bar{\omega}_0 = \frac{d_1 + \frac{1}{4}}{d_1 + d_0} \tag{2.25}$$

d_1 is a free parameter. We can see that $d_0 \geq 1/4$ when $d_1 \geq 0$ which then implies that the ideal weight $\bar{\omega}_0 < 1$, therefore implying that the ideal weights are all positive. Substituting Eq. (2.25) into the Δx^6 truncation error coefficient for the point value reconstruction (from Table 2.2) gives

$$T_6^{PV} = \frac{9}{4480}\left(\frac{65 - 61d_1}{1 + d_1}\right) \tag{2.26}$$

So the point value reconstruction is sixth-order accurate except when $d_1 = \frac{65}{61}$ when it becomes eighth-order accurate. On the other hand, substituting Eq. (2.25) into the Δx^5 truncation error coefficient for the subcell reconstruction (from Table 2.1) gives

$$T_5^{SA} = \frac{-3}{256}\left(\frac{5 + 23d_1}{1 + d_1}\right) \tag{2.27}$$

which unfortunately does not vanish for any positive value of d_1 so the subcell reconstruction will always be fifth-order accurate. So the subschemes for the subcell and point value reconstructions are:

$$d_0 = \frac{5 + 8d_1}{17 + 20d_1} \tag{2.28}$$

$$\bar{\omega}_{-1} = \frac{1 - \bar{\omega}_0}{2}, \quad \bar{\omega}_0 = \frac{d_1 + \frac{1}{4}}{d_1 + d_0}, \quad \bar{\omega}_1 = \frac{1 - \bar{\omega}_0}{2} \tag{2.29}$$

$$\begin{aligned}
\frac{1-d_1}{2}\bar{q}^L_{j-1} + \frac{1+d_1}{2}\bar{q}^L_j &= \frac{-d_1}{16}\bar{q}_{j-2} + \frac{3-d_1}{8}\bar{q}_{j-1} + \frac{3d_1+2}{16}\bar{q}_j \\
\frac{1-d_0}{4}\bar{q}^L_{j-1} + \frac{1+d_0}{2}\bar{q}^L_j + \frac{1-d_0}{4}\bar{q}^L_{j+1} &= \frac{3-2d_0}{16}\bar{q}_{j-1} + \frac{1+d_0}{4}\bar{q}_j + \frac{1-2d_0}{16}\bar{q}_{j+1} \\
\frac{1+d_1}{2}\bar{q}^L_j + \frac{1-d_1}{2}\bar{q}^L_{j+1} &= \frac{5d_1+6}{16}\bar{q}_j + \frac{1-3d_1}{8}\bar{q}_{j+1} + \frac{d_1}{16}\bar{q}_{j+2}
\end{aligned} \tag{2.30}$$

$$\begin{aligned}
\frac{1-d_1}{2}q_{j-1} + \frac{1+d_1}{2}q_j &= \frac{-1}{24}\bar{q}_{j-2} + \frac{7-6d_1}{12}\bar{q}_{j-1} + \frac{11+12d_1}{24}\bar{q}_j \\
\frac{1-d_0}{4}q_{j-1} + \frac{1+d_0}{2}q_j + \frac{1-d_0}{4}q_{j+1} &= \frac{5-6d_0}{24}\bar{q}_{j-1} + \frac{7+6d_0}{12}\bar{q}_j + \frac{5-6d_0}{24}\bar{q}_{j+1} \\
\frac{1+d_1}{2}q_j + \frac{1-d_1}{2}q_{j+1} &= \frac{11+12d_1}{24}\bar{q}_j + \frac{7-6d_1}{12}\bar{q}_{j+1} + \frac{-1}{24}\bar{q}_{j+2}
\end{aligned} \tag{2.31}$$

2.2.3 Derivative Reconstruction

The third spatial reconstruction is that of the flux derivatives from the values of the fluxes evaluated at cell midpoints. Suppose the reconstruction is given by

$$\sum_{s,k} \omega_s(\sigma) L_{sk} q'_{j+k} = \sum_{s,k} \omega_s(\sigma) R_{sk} q_{j+k} \tag{2.32}$$

Reflecting the scheme about point j should produce the same derivative but with opposite sign:

$$\sum_{s,k} \omega_s(-\sigma) L_{s,-k} q'_{j+k} = -\sum_{s,k} \omega_s(-\sigma) R_{s,-k} q_{j+k} \tag{2.33}$$

Therefore the weights and coefficients must satisfy:

$$\sum_{s} \omega_s(\sigma) L_{sk} = \sum_{s} \omega_s(-\sigma) L_{s,-k} \tag{2.34}$$

$$\sum_{s} \omega_s(\sigma) R_{sk} = -\sum_{s} \omega_s(-\sigma) R_{s,-k} \tag{2.35}$$

As before, expressing the nonlinear weights in terms of the ideal weights $\bar{\omega}_s$ and smoothness functions $f_s(\sigma)$ allows us to write the conditions Eq. (2.34) as:

$$\begin{aligned}\sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)} L_{sk} &= \sum_s \frac{\bar{\omega}_s f_{-s}(\sigma)}{\sum_r \bar{\omega}_r f_{-r}(\sigma)} L_{s,-k} \\ &= \sum_s \frac{\bar{\omega}_{-s} f_s(\sigma)}{\sum_r \bar{\omega}_{-r} f_r(\sigma)} L_{-s,-k}\end{aligned} \tag{2.36}$$

and Eq. (2.35) as:

$$\begin{aligned}\sum_s \frac{\bar{\omega}_s f_s(\sigma)}{\sum_r \bar{\omega}_r f_r(\sigma)} R_{sk} &= -\sum_s \frac{\bar{\omega}_s f_{-s}(\sigma)}{\sum_r \bar{\omega}_r f_{-r}(\sigma)} R_{s,-k} \\ &= -\sum_s \frac{\bar{\omega}_{-s} f_s(\sigma)}{\sum_r \bar{\omega}_{-r} f_r(\sigma)} R_{-s,-k}\end{aligned} \tag{2.37}$$

We then have the same symmetry condition on the weights:

$$\bar{\omega}_s = \bar{\omega}_{-s} \tag{2.38}$$

and on the LHS coefficients:

$$L_{sk} = L_{-s,-k} \tag{2.39}$$

and a different condition on the RHS coefficients:

$$R_{sk} = -R_{-s,-k} \tag{2.40}$$

The locations of the nonzero entries in L_{sk} must remain the same to ensure proper decoupling at discontinuities, so the form of L_{sk} is unchanged but the diagonal excesses c_0, c_1 may differ from those in the subcell and point value reconstructions:

$$L^{FD} = \begin{bmatrix} \frac{1-c_1}{2} & \frac{1+c_1}{2} & 0 \\ \frac{1-c_0}{4} & \frac{1+c_0}{2} & \frac{1-c_0}{4} \\ 0 & \frac{1+c_1}{2} & \frac{1-c_1}{2} \end{bmatrix} \tag{2.41}$$

The RHS coefficients of the subschemes with these LHS coefficients and maximum order of accuracy are:

$$R^{FD} = \begin{bmatrix} \frac{d_1}{2} & -1-d_1 & 1+\frac{d_1}{2} & 0 & 0 \\ 0 & -\frac{1}{2} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & -1-\frac{d_1}{2} & 1+d_1 & -\frac{d_1}{2} \end{bmatrix} \tag{2.42}$$

Table 2.3 shows the leading terms in the truncation error expansion.

Table 2.3: Truncation error coefficients of the derivative reconstruction.

Power	Δx^2	Δx^3	Δx^4	Δx^5
Coefficient	$\frac{1}{2} - \frac{3}{2}\bar{\omega}_0 d_0 - \frac{3}{2}\bar{\omega}_0 d_1 + \frac{3d_1}{2}$	0	$\frac{3}{2} - \frac{5}{2}\bar{\omega}_0 d_0 - \frac{25}{2}\bar{\omega}_0 d_1 + \frac{25}{2} d_1$	0

To cancel the second- and fourth-order error terms requires $\bar{\omega}_0 d_0 = \frac{4}{15}$ which then implies:

$$\frac{1}{15} = d_1(\bar{\omega}_0 - 1) > 0 \tag{2.43}$$

However, diagonal dominance requires $d_1 > 0$ and positive weights require $\bar{\omega}_0 < 1$, which together are incompatible with Eq. (2.43). The derivative reconstruction will therefore have to be fourth order at best. A fourth-order compact reconstruction would need to outperform the fourth-order non-compact scheme obtained by setting $d_1 = d_0 = 1$ and choosing $\bar{\omega}_0 = \frac{2}{3}$ to cancel the second-order error term, which is given by:

$$\bar{\omega}_{-1} = \frac{1}{3}, \bar{\omega}_0 = \frac{2}{3}, \bar{\omega}_1 = \frac{1}{3} \tag{2.44}$$

$$
\begin{aligned}
q'_j &= \frac{1}{2}q_{j-2} - 2q_{j-1} + \frac{3}{2}q_j \\
q'_j &= -\frac{1}{2}q_{j-1} + \frac{1}{2}q_{j+1} \\
q'_j &= -\frac{3}{2}q_j + 2q_{j+1} - \frac{1}{2}q_{j+2}
\end{aligned}
\tag{2.45}
$$

Either reconstruction would be performed four times per time step during the Runge-Kutta process, magnifying the cost of solving the systems arising from a compact reconstruction. In practice, the difference in accuracy between compact and non-compact derivative reconstructions is minimal whereas the former requires substantially more time. Therefore we will use the non-compact reconstruction Eqs. (2.44)-(2.45) for the flux derivatives.

2.3 Boundary Treatment

If the array of LHS coefficients $L_{s,k} \neq 0$ when $k \neq 0$, however, then at boundaries the scheme would call for subcells to be placed outside the boundaries. Rather than prescribe values for such subcells, we alter the LHS stencil at the boundaries to involve only subcells inside the domain. The symmetry condition $L_{sk} = L_{-s,-k}$ must apply at every cell so that the subcell reconstruction will be consistent, however, and implies that for the stencil to not protrude past the boundary it must also not extend into the interior; it can contain only the cell immediately adjacent to the boundary. Therefore the boundary subschemes must all be non-compact. These schemes can be found easily by setting the diagonal excess parameters $d_0 = d_1 = 1$ which prevents the error cancellation conditions Eq. (2.23) and Eq. (2.24) from holding simultaneously. Canceling the $\mathcal{O}(\Delta x^4)$ error term in the point value recon-

struction by satisfying Eq. (2.24) would lead to negative ideal weights, and more importantly would degenerate the subcell reconstruction from fifth to third order. Therefore we satisfy Eq. (2.23) at the cost of fourth-order accuracy in the point value reconstruction at boundaries. These considerations lead to the following boundary scheme:

$$\bar{\omega}_{-1} = \frac{3}{16},\ \bar{\omega}_0 = \frac{5}{8},\ \bar{\omega}_1 = \frac{3}{16} \tag{2.46}$$

$$\begin{aligned} \bar{q}_j^L &= \frac{-1}{16}\bar{q}_{j-2} + \frac{1}{4}\bar{q}_{j-1} + \frac{5}{16}\bar{q}_j \\ \bar{q}_j^L &= \frac{1}{16}\bar{q}_{j-1} + \frac{1}{2}\bar{q}_j - \frac{1}{16}\bar{q}_{j+1} \\ \bar{q}_j^L &= \frac{11}{16}\bar{q}_j - \frac{1}{4}\bar{q}_{j+1} + \frac{1}{16}\bar{q}_{j+2} \end{aligned} \tag{2.47}$$

$$\begin{aligned} q_j &= \frac{-1}{24}\bar{q}_{j-2} + \frac{1}{12}\bar{q}_{j-1} + \frac{23}{24}\bar{q}_j \\ q_j &= \frac{-1}{24}\bar{q}_{j-1} + \frac{13}{12}\bar{q}_j + \frac{-1}{24}\bar{q}_{j+1} \\ q_j &= \frac{23}{24}\bar{q}_j + \frac{1}{12}\bar{q}_{j+1} + \frac{-1}{24}\bar{q}_{j+2} \end{aligned} \tag{2.48}$$

The derivative reconstruction is already non-compact and does not need to be modified for boundary cells.

2.4 Multidimensional Extension

The staggered-grid framework extends straightforwardly to arbitrarily many space dimensions, with the staggered cells being centered at the vertices of the main cells as in Fig. 2.6 for the two-dimensional case.

The multidimensional analog of Eq. (1.20) describing the evolution of cell averages is:

$$\bar{q}_{\vec{j}+1/2}^{n+1} = \bar{q}_{\vec{j}+1/2}^{n} - \frac{1}{|\Omega_{\vec{j}+1/2}|}\int_{t^n}^{t^{n+1}}\int_{\partial\Omega_{\vec{j}+1/2}} f_d n_d \, dS \tag{2.49}$$

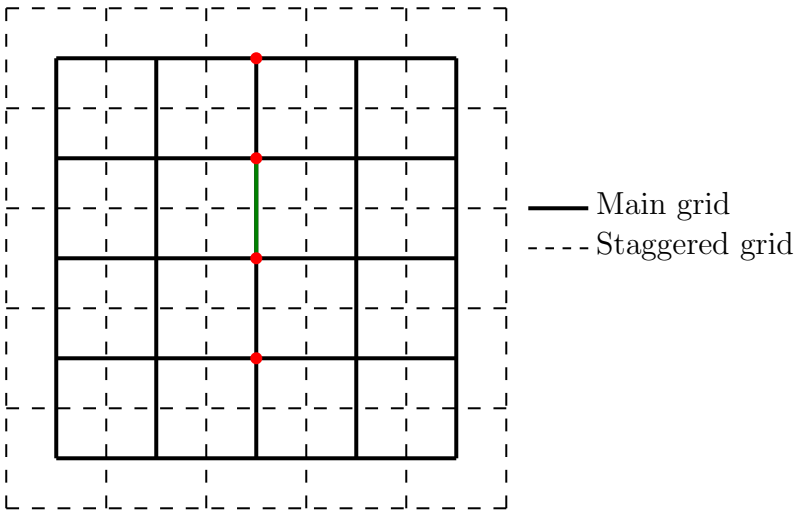

Figure 2.6: Staggering in each dimension produces a grid whose cells are centered on the vertices of the main grid.

where $\Omega_{j+1/2}$ is the cell centered at point $x_{j+1/2}$, $|\Omega_{j+1/2}|$ is its volume, n_d is the dth component of the unit normal vector to the boundary $\partial\Omega_{j+1/2}$, and f_d is the physical flux function in the dth dimension. Here $\vec{j}$ is a multi-index. The procedure of obtaining the initial staggered average $\bar{q}^n_{j+1/2}$ by reconstructing averages over subcells then evolving point values to obtain the time-integrated fluxes does not change; however, the multidimensional case alters the nature of the quantities being reconstructed in ways that would appear to greatly complicate the method. First, the subcells are orthants of their respective cells which would appear to necessitate construction of $2^D - 1$ different subschemes for all the subcell averages not computed by conservation. Second, the point values of interest are those located at the cell centers. Finally, the boundary integral in Eq. (2.49) requires quadrature of surface integrals, as opposed to straightforward evaluation at a point. All of these challenges, however, can be easily addressed by considering tensor products of the

one-dimensional schemes which we now describe.

A multi-index notation will simplify the coming discussion. For the D-dimensional case, an arrow above a symbol denotes an ordered tuple of D elements. Then we define:

Table 2.4: Notation conventions for multidimensional schemes.

Object or operation	Multi-index notation	Equivalent
Tuple	$\vec{i}$	$(i_1, i_2, \cdots, i_D)$
Tuple of identical numerical constants c	$\vec{c}$	$(c, c, \cdots, c)$
Coordinate tuple in direction d	$\vec{e}_d$	$(\delta_{1d}, \delta_{2d}, \cdots, \delta_{Dd})$
Tuple subscript of sequence ϕ_n	$\phi_{\vec{i}}$	$(\phi_{i_1}, \phi_{i_2}, \cdots, \phi_{i_D})$
Tuple subscript of array $\boldsymbol{B}$	$\boldsymbol{B}_{\vec{i}}$	$\boldsymbol{B}_{i_1,i_2,\cdots,i_D}$
Generic binary operation * on tuples	$\vec{u} * \vec{v}$	$(u_1 * v_1, u_2 * v_2, \cdots, u_D * v_D)$
Scalar multiplication	$a\vec{u}$	$(au_1, au_2, \cdots, au_D)$
Product of tuple elements	$\Pi\vec{u}$	$u_1 u_2 \cdots u_D$
Tuple interval	$[\vec{a}, \vec{b}]$	$[a_1, b_1] \times [a_2, b_2] \times \cdots \times [a_D, b_D]$
Tuple differential	$d\vec{z}$	$dz_D dz_{D-1} \cdots dz_2 dz_1$
Integral with vector bounds	$\int_{\vec{a}}^{\vec{b}} Q\, d\vec{z}$	$\int_{a_1}^{b_1} \cdots \int_{a_D}^{b_D} Q\, dz_D \cdots dz_1$

2.4.1 Tensor Product Extensions

Consider a D-dimensional region $\Omega = [a_0, b_0] \times [a_1, b_1] \cdots \times [a_{D-1}, b_{D-1}]$ and let $\vec{\Delta x} = (\Delta x_0, \Delta x_1, \cdots, \Delta x_{D-1})$ be a tuple of cell widths in each dimension. The integral (normalized by the cell volume $\Pi\vec{\Delta x}$) of an integrable function u over Ω can be written as an iterated integral:

$$\frac{1}{\Pi\vec{\Delta x}} \int_\Omega u(x)\, dx = \frac{1}{\Delta x_0} \int_{a_0}^{b_0} \left(\frac{1}{\Delta x_1} \int_{a_1}^{b_1} \left(\cdots \left(\frac{1}{\Delta x_{D-1}} \int_{a_{D-1}}^{b_{D-1}} u\, dx_{D-1} \right) \cdots \right) dx_2 \right) dx_1 \tag{2.50}$$

Therefore averaging over rectangular Ω is equivalent to taking successive averages over each coordinate direction within the cell, each of which is a one-dimensional process. Suppose $\Omega^L_{\vec{j}}$ is the subcell of cell $\vec{j}$ given by $[x_{\vec{j}-1/2}, x_{\vec{j}}]$ and that we have a one-dimensional subcell reconstruction scheme given by:

$$\sum_k L_k \bar{q}^L_{j+k} = \sum_k R_k \bar{q}_{j+k} + \mathcal{O}(\Delta x^p) \tag{2.51}$$

Then we may rewrite the outermost integral in Eq. (2.50) using the reconstruction Eq. (2.51):

$$\begin{aligned}\sum_{k_0} L_{k_0} \frac{1}{\Pi\vec{\Delta x}} \int_{\Omega^L_{\vec{j}+k_0\vec{e}_0}} &= \sum_{k_0} L_{k_0} \frac{1}{\Delta x_0} \int_{x_{j_0-1/2+k_0}}^{x_{j_0+k_0}} \left(\frac{1}{\Delta x_1} \int_{x_{j_1-1/2}}^{x_{j_1}} (\cdots)\, dx_1 \right) dx_0 \\ &= \sum_{k_0} R_{k_0} \frac{1}{\Delta x_0} \int_{x_{j_0-1/2+k_0}}^{x_{j_0+1/2+k_0}} \left(\frac{1}{\Delta x_1} \int_{x_{j_1-1/2}}^{x_{j_1}} (\cdots)\, dx_1 \right) dx_0 + \mathcal{O}(\Delta x_0^p)\end{aligned} \tag{2.52}$$

Then we may apply the reconstruction to Eq. (2.52) along the x_1 direction:

$$\begin{aligned}&\sum_{k_1} L_{k_1} \left(\sum_{k_0} L_{k_0} \frac{1}{\Pi\vec{\Delta x}} \int_{\Omega^L_{\vec{j}+k_0\vec{e}_0+k_1\vec{e}_1}} \right) = \\ &\sum_{k_1} L_{k_1} \left(\sum_{k_0} R_{k_0} \frac{1}{\Delta x_0} \int_{x_{j_0-1/2+k_0}}^{x_{j_0+1/2+k_0}} \left(\frac{1}{\Delta x_1} \int_{x_{j_1-1/2+k_1}}^{x_{j_1+k_1}} (\cdots)\, dx_1 \right) dx_0 + \mathcal{O}(\Delta x_0^p) \right) \\ &= \sum_{k_0,k_1} R_{k_1} R_{k_0} \left(\frac{1}{\Delta x_0} \int_{x_{j_0-1/2+k_0}}^{x_{j_0+1/2+k_0}} \frac{1}{\Delta x_1} \int_{x_{j_1-1/2+k_1}}^{x_{j_1+1/2+k_1}} \left(\frac{1}{\Delta x_2} \int_{x_{j_2-1/2+k_2}}^{x_{j_2+k_2}} (\cdots)\, dx_2 \right) dx_1 \right) dx_0 \\ &+ \mathcal{O}(\Delta x_0^p) + \mathcal{O}(\Delta x_1^p)\end{aligned} \tag{2.53}$$

Repeating this process for each dimension eventually produces the complete tensor-product extension:

$$\sum_{\vec{k}} \left(\Pi L_{\vec{k}}\right) \bar{q}^L_{\vec{j}+\vec{k}} = \sum_{\vec{k}} \left(\Pi R_{\vec{k}}\right) \bar{q}_{\vec{j}+\vec{k}} + \mathcal{O}(\Delta x_0^p) + \cdots + \mathcal{O}(\Delta x_{D-1}^p) \tag{2.54}$$

Note that the order of accuracy is unchanged. Eq. (2.54) defines the average for the subcell that occupies the left half along each dimension of the full cell which contains it. The averages for other subcells can be obtained by replacing Eq. (2.51) with its one-dimensional equivalent for the right subcell at the appropriate steps of the process Eqs. (2.52)-(2.54) producing the general tensor-product extension:

$$\sum_{\vec{k}} \left(L_{k_0}^{\alpha_0} L_{k_1}^{\alpha_1} \cdots L_{k_{D-1}}^{\alpha_{D-1}} \right) \bar{q}_{\vec{j}+\vec{k}}^{\alpha} = \sum_{\vec{k}} \left(R_{k_0}^{\alpha_0} R_{k_1}^{\alpha_1} \cdots R_{k_{D-1}}^{\alpha_{D-1}} \right) \bar{q}_{\vec{j}+\vec{k}} + \mathcal{O}(\Delta x_0^p) + \cdots + \mathcal{O}(\Delta x_{D-1}^p) \tag{2.55}$$

where α is a multi-index identifying the orthant occupied by the subcell in question and the superscripts on L and R indicate whether the coefficient comes from the left- or right-subcell scheme. Note that unlike in the one-dimensional case, in multiple dimensions the right-subcell scheme is actually used because more than one subcell average must be computed before obtaining the last one by conservation.

The point value may be considered as the limiting case of a function average as the averaging region shrinks to the cell center, therefore the same argument leads to a tensor-product extension of the point value scheme. If the one-dimensional point value scheme is given by:

$$\sum_k L_k^{PV} q_{j+k} = \sum_k R_k^{PV} \bar{q}_{j+k} + \mathcal{O}(\Delta x^p) \tag{2.56}$$

then the tensor-product extension is:

$$\sum_{\vec{k}} (\Pi L_{\vec{k}}) q_{\vec{j}+\vec{k}} = \sum_{\vec{k}} (\Pi R_{\vec{k}}) \bar{q}_{\vec{j}+\vec{k}} + \mathcal{O}(\Delta x_0^p) + \cdots + \mathcal{O}(\Delta x_{D-1}^p) \tag{2.57}$$

The foregoing discussion applies directly to tensor-product extensions of sub-schemes of WENO-type schemes and to tensor-product extensions of the combined

schemes with ideal weights. Let c_{sk} be the array of (left- or right-hand side) coefficients of the one-dimensional subschemes and C_k the coefficients of the one-dimensional combined scheme. Then the coefficients of the multidimensional extension of the combined scheme are:

$$\begin{aligned}\sum_{\vec{k}} \left(\Pi C_{\vec{k}}\right) &= \sum_{\vec{k}} \prod_{d=1}^{D} \left(\sum_{s_d} \bar{\omega}_{s_d} c_{s_d,k_d} \right) \\ &= \sum_{\vec{k}} \sum_{\vec{s}} \prod_{d=1}^{D} \bar{\omega}_{s_d} c_{s_d,k_d} \\ &= \sum_{\vec{s}} \left(\prod_{d=1}^{D} \bar{\omega}_{s_d} \right) \sum_{\vec{k}} \left(\prod_{d=1}^{D} c_{s_d,k_d} \right)\end{aligned} \tag{2.58}$$

The product in the sum over offsets $\vec{k}$ is the coefficient array of the tensor-product extension of the subschemes given by the multi-index $\vec{s}$, and therefore the multi-dimensional ideal weight corresponding to that combination of subschemes must be:

$$\bar{\omega}_{\vec{s}} = \prod_{d=1}^{D} \bar{\omega}_{s_d} \tag{2.59}$$

Figure 2.7 shows an example of this construction in two dimensions. A one-dimensional subscheme for the left subcell in the x direction (index r, shown in red) is combined with another one-dimensional subscheme for the left subcell in the y direction (index s, shown in blue) to produce a two-dimensional subscheme for the subcell in the lower left quadrant. For clarity we show only the left-hand side coefficients, but the same process applies to the right-hand side coefficients.

Figure 2.8 shows the relationship between the main and staggered grids in the two-dimensional case.

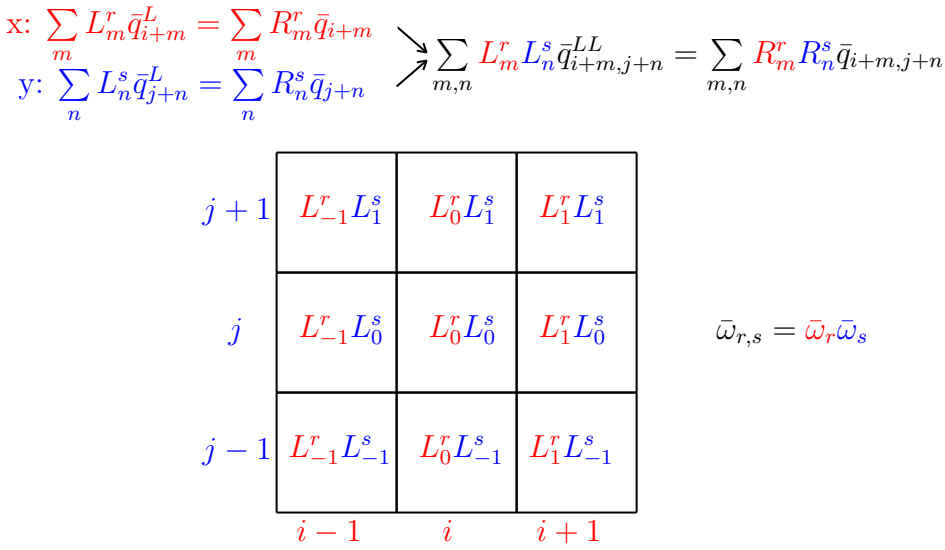

Figure 2.7: A subscheme (red) in the x direction is combined with a subscheme (blue) in the y direction to produce a two-dimensional subscheme.

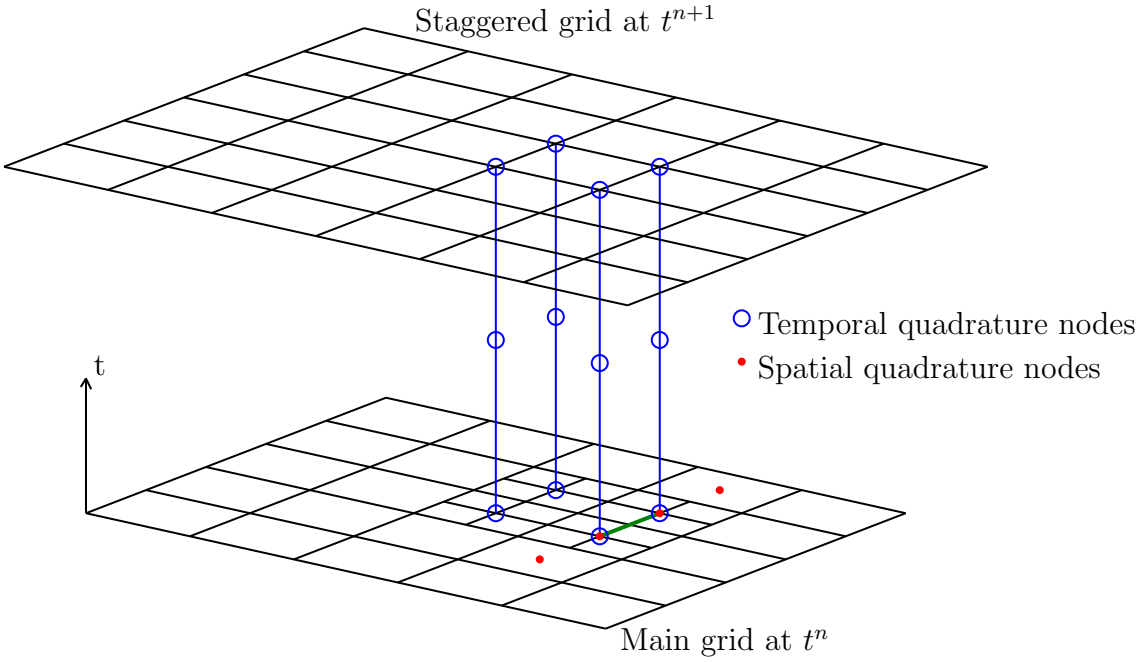

Figure 2.8: Evolution from main to staggered grids in two dimensions.

2.4.2 Surface Integral Quadrature

In multiple dimensions the flux integrals become surface integrals which must be approximated. In two dimensions, for example Eq. (2.49) becomes:

$$\begin{aligned}\bar{q}^{n+1}_{i+1/2,j+1/2} = \bar{q}^{n}_{i+1/2,j+1/2} \\ - \frac{1}{\Delta x \Delta y}\int_{t^n}^{t^{n+1}} \left(\int_{x_i}^{x_{i+1}} f_1(u(x, y_{j+1})) - f_1(u(x, y_j))\, dx \right) dt \\ - \frac{1}{\Delta x \Delta y}\int_{t^n}^{t^{n+1}} \left(\int_{y_j}^{y_{j+1}} f_0(u(x_{i+1}, y)) - f_0(u(x_i, y))\, dy \right) dt\end{aligned} \tag{2.60}$$

Though it is possible to evaluate these integrals by a WENO-like process of combining candidate values depending on local smoothness, as in [30] we find that it is enough to approximate them by a simple quadrature rule which for the face shown in green in Fig. 2.6 involves the points shown there in red:

$$\frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} u(x)\, dx = \frac{-1}{12}u_{i-1} + \frac{13}{12}u_i + \frac{13}{12}u_{i+1} - \frac{1}{12}u_{i+2} + \mathcal{O}(\Delta x^4) \tag{2.61}$$

The desired fifth-order spatial accuracy of the overall method requires, however, that a more accurate quadrature be used such as the sixth-order rule:

$$\frac{1}{\Delta x}\int_{x_i}^{x_{i+1}} u(x)\, dx = \frac{11}{1440}u_{i-2} - \frac{31}{480}u_{i-1} + \frac{401}{720}u_i + \frac{401}{720}u_{i+1} - \frac{31}{480}u_{i+2} + \frac{11}{1440}u_{i+3} + \mathcal{O}(\Delta x^6) \tag{2.62}$$

In either case, the integral is approximated by a linear combination of values of the integrand at midpoints, which will be computed by a Runge-Kutta method with its natural continuous extension as in the one-dimensional case. In implementation it is convenient to exchange the order of integration in (2.60), calculating first the time integrals of the fluxes at each cell center and then using Eq. (2.62) to approximate the spatial integrals. If more than two dimensions are involved, then the

surface integrals themselves are multidimensional and can be approximated by the tensor-product extension of a one-dimensional rule. That is, if the one-dimensional approximation is given by:

$$\frac{1}{\Delta x}\int_{x_j}^{x_{j+1}} q(x)\,dx = \sum_k c_k q_{j+k} + \mathcal{O}(\Delta x^p) \tag{2.63}$$

then the integral over the surface $S_d = [x_{j_0}, x_{j_0} + \Delta x_0] \times [x_{j_1}, x_{j_1} + \Delta x_1] \times \cdots \times x_{j_d} \times \cdots \times [x_{j_{D-1}}, x_{j_{D-1}} + \Delta x_{D-1}]$ (i.e. a surface normal to the x_d direction) is approximated by:

$$\frac{\Delta x_d}{\Pi\vec{\Delta x}}\int_{S_d} q\,ds = \sum_{\vec{k},k_d=0} \Pi_{s\neq d} c_{k_s} q_{\vec{j}+\vec{k}} \tag{2.64}$$

2.4.3 Smoothness Indicators

Jiang and Shu [15] define the one-dimensional indicator as a sum of scaled L^2 norms of derivatives of the reconstruction polynomial $\tilde{P}(x)$ over the cell $[x^* - \Delta x/2, x^* + \Delta x/2]$ common to all stencils:

$$\beta = \int_{x^*-\Delta x/2}^{x^*+\Delta x/2} \sum_{r=1}^{r_d} \frac{1}{\Delta x}\left(\Delta x^r \frac{d^r\tilde{P}}{dx^r}\right)^2 dx \tag{2.65}$$

Here r_d is the number of derivatives used (usually $r_d = 2$). We generalize this definition to higher dimensions by considering derivatives of the multidimensional reconstruction polynomial $P(\vec{x})$ for the cell centered at $\vec{x}^*$ in each of the coordinate directions:

$$\beta = \int_{\vec{x}^*-\vec{\Delta x}/2}^{\vec{x}^*+\vec{\Delta x}/2} \sum_{d=1}^{D}\sum_{r=1}^{r_d} \frac{1}{\Pi\vec{\Delta x}}\left(\Delta x_d^r \frac{\partial^r\tilde{P}}{\partial x_d^r}\right)^2 d\vec{x} \tag{2.66}$$

Note that $\Pi\vec{\Delta}x = \Delta x_1 \Delta x_2 \cdots \Delta x_D$ is the volume of a cell. Define the array B of coefficients of $\tilde{P}(\vec{x})$ according to:

$$\tilde{P}(\vec{x}) = \sum_{\{\vec{i}|0\le i_s\le r_s\}} \boldsymbol{B}_{\vec{i}} \Pi \left(\frac{\vec{x} - \vec{x}^*}{\vec{\Delta x}} \right)^{\vec{i}} \frac{1}{\vec{i}!} \tag{2.67}$$

Define the nondimensional displacement $\vec{z}$ by:

$$\vec{z} = \frac{\vec{x} - \vec{x}^*}{\vec{\Delta x}} \tag{2.68}$$

Then the reconstruction polynomial becomes:

$$P(\vec{z}) = \sum_{\{\vec{i}|0\le i_s\le r_s\}} \boldsymbol{B}_{\vec{i}} \Pi \left(\vec{z}^{\vec{i}} \right) \frac{1}{\vec{i}!} \tag{2.69}$$

and the definition of the indicator becomes:

$$\beta = \int_{-\vec{1}/2}^{\vec{1}/2} \sum_{d=1}^{D} \sum_{r=1}^{r_d} \left(\frac{\partial^r P}{\partial z_d^r} \right)^2 d\vec{z} \tag{2.70}$$

The necessary derivatives of P can be found from (2.69):

$$\frac{\partial P}{\partial z_d} = \sum_{0\le i_s\le r_s} \boldsymbol{B}_{\vec{i}} i_d \Pi \left(\vec{z}^{\vec{i}-\vec{e}_d} \right) \frac{1}{\vec{i}!} \tag{2.71}$$

where the sum runs over all D-tuples $(i_1, i_2, \cdots, i_D)$ such that $0 \le i_s \le r_s$ for each $s = 1, 2, \cdots, D$. Since the summand with $i_d = 0$ vanishes, we may skip that term in the sum to avoid unneeded computations:

$$\frac{\partial P}{\partial z_d} = \sum_{\delta_{sd}\le i_s\le r_s} \boldsymbol{B}_{\vec{i}} \Pi \left(\frac{\vec{z}^{\vec{i}-\vec{e}_d}}{(\vec{i} - \vec{e}_d)!} \right) \tag{2.72}$$

where δ_{sd} is the Kronecker delta. A similar equation holds for the rth derivative:

$$\frac{\partial^r P}{\partial z_d^r} = \sum_{r\delta_{sd}\le i_s\le r_s} \boldsymbol{B}_{\vec{i}} \Pi \left(\frac{\vec{z}^{\vec{i}-r\vec{e}_d}}{(\vec{i} - r\vec{e}_d)!} \right) \tag{2.73}$$

$$\beta = \sum_{d=1}^{D} \sum_{r=1}^{r_d} \int_{-\vec{1}/2}^{\vec{1}/2} \left(\sum_{r\delta_{sd} \le i_s \le r_s} \boldsymbol{B}_{\vec{i}} \Pi \left(\frac{\vec{z}^{\vec{i}-r\vec{e}_d}}{(\vec{i}-r\vec{e}_d)!} \right) \right)^2 d\vec{z} \tag{2.74}$$

Rather than reduce this definition to a formula giving the indicator directly from cell averages, we evaluate it by Gaussian quadrature. This approach is more general and circumvents the need to store such formulas once they are computed. Denote the one-dimensional Gaussian quadrature nodes in $[-1/2, 1/2]$ by ξ_q with weights w_q, for $q = 0, 1, ..., Q$. Then converting the integrals in (2.74) gives:

$$\beta = \sum_{\vec{0} \le \vec{q} \le \vec{Q}} \Pi w_{\vec{q}} \sum_{d=1}^{D} \sum_{r=1}^{r_d} \left(\sum_{r\delta_{sd} \le i_s \le r_s} \boldsymbol{B}_{\vec{i}} \Pi \left(\frac{\xi_{\vec{q}}^{\vec{i}-r\vec{e}_d}}{(\vec{i}-r\vec{e}_d)!} \right) \right)^2 \tag{2.75}$$

Written in more traditional notation:

$$\beta = \sum_{\vec{0} \le \vec{q} \le \vec{Q}} \left(\prod_{j=1}^{D} w_{q_j} \right) \sum_{d=1}^{D} \sum_{r=1}^{r_d} \left[\sum_{\substack{\vec{i}, \\ r\delta_{sd} \le i_s \le r_s}} \boldsymbol{B}_{\vec{i}} \prod_{j=1}^{D} \left(\frac{\xi_{q_j}^{i_j - r\delta_{jd}}}{(i_j - r\delta_{jd})!} \right) \right]^2 \tag{2.76}$$

To complete the calculation of β we need to obtain the coefficients $\boldsymbol{B}_{\vec{i}}$ from the cell averages. The reconstruction polynomial is defined by requiring that it match the cell averages on neighboring cells $\vec{j}$:

$$\begin{aligned} \bar{u}_{\vec{j}} &= \int_{-\vec{1}/2+\vec{j}}^{\vec{1}/2+\vec{j}} \sum_{\vec{0} \le \vec{i} \le \vec{r}_s} \boldsymbol{B}_{\vec{i}} \frac{\vec{z}^{\vec{i}}}{\vec{i}!} \, d\vec{z} \\ &= \sum_{\vec{0} \le \vec{i} \le \vec{r}_s} \boldsymbol{B}_{\vec{i}} \prod_{d=1}^{D} \left(\int_{-1/2+j_d}^{1/2+j_d} \frac{z_d^{i_d}}{i_d!} \, dz_d \right) \\ &= \sum_{\vec{0} \le \vec{i} \le \vec{r}_s} \boldsymbol{B}_{\vec{i}} \prod_{d=1}^{D} \left(\frac{(j_d + 1/2)^{i_d+1} - (j_d - 1/2)^{i_d+1}}{(i_d+1)!} \right) \\ &= \sum_{\vec{0} \le \vec{i} \le \vec{r}_s} \boldsymbol{B}_{\vec{i}} \prod_{d=1}^{D} V_{j_d, i_d}^{(d)} \end{aligned} \tag{2.77}$$

Each $V^{(d)}$ is a matrix that produces the average over the interval from $(0, \cdots, 0, j_d - 1/2, 0, \cdots, 0)$ to $(0, \cdots, 0, j_d + 1/2, 0, \cdots, 0)$ from the coefficient array $\boldsymbol{B}$. Therefore

the entries of $\boldsymbol{B}$ can be recovered using the inverses of the $V^{(d)}$:

$$\boldsymbol{B}_{i_1,\cdots,i_D} = \sum_{(j_1,\cdots,j_D),j_s\in J_s} \bar{u}_{j_1,\cdots,j_D} \prod_{d=1}^{D} \left[\left(V^{(d)}\right)^{-1}\right]_{i_d,j_d} \tag{2.78}$$

The matrices $\left(V^{(d)}\right)^{-1}$ can be precomputed and depend only on the set of cells over which $P(\vec{z})$ is defined.

2.5 Numerical Analysis

The CCRWENO method consists of four processes:

1. Reconstruction of subcell averages from cell averages
2. Reconstruction of point values from cell averages
3. Reconstruction of flux derivatives from flux point values
4. Time advancement

Heuristically, one expects that the CCRWENO method would become unstable if any of these steps is individually unstable in some appropriate sense. We will consider the subcell and point value reconstructions individually and require that when the input is a single Fourier mode the output is a Fourier mode of lesser or equal amplitude. This analysis will also suggest values for the diagonal dominance parameter d_1.

2.5.1 Subcell Reconstruction

Let L_k, R_k be respectively the left- and right-hand side coefficients for cell $j+k$ of the combined scheme using the ideal weights. The amplitude condition is (see

Appendix A):

$$\left| \frac{\sum_k R_k e^{ik\theta}}{\sum_k L_k e^{ik\theta}} \frac{2\sin(k\theta/2)}{1 - e^{-ik\theta/2}} \right| \leq 1 \tag{2.79}$$

for all $0 \leq \theta \leq \pi$ where $\theta = m\Delta x$ is the wavenumber m normalized by the grid spacing. After some simplifications Eq. (2.79) becomes

$$G(d_1, \theta) = \left| \frac{\sum_k R_k e^{ik\theta}}{\sum_k L_k e^{ik\theta}} \frac{2\sin(k\theta/2)}{1 - e^{-ik\theta/2}} \right| \leq 1 \tag{2.80}$$

Note that the coefficients depend on d_0, d_1, and $\bar{\omega}_0$ the first and third of which depend on d_1 due to the accuracy constraint and the compatibility with the point value reconstruction Eq. (2.25). Thus the amplification factor G depends on the diagonal dominance parameter d_1 and the normalized wavenumber θ. Figure 2.9 shows the G as a function of those two parameters.

We see that not only is the amplification factor less than unity for all wavenumbers and all $d_1 \in [0, 2]$, but that $G(d_1, \theta)$ becomes less sensitive to d_1 as d_1 increases. In practice, at the higher wavenumbers $\theta > \pi/2$ the nonlinear effects of the WENO weight adaptation intervene to stabilize the reconstruction so the variation of the amplification factor in that region is less relevant.

2.5.2 Point Value Reconstruction

Let L_k, R_k be respectively the left- and right-hand side coefficients for cell $j+k$ of the combined scheme using the ideal weights. The amplitude condition is (see Appendix A):

$$G(d_1, \theta) = \left| \frac{\sum_k R_k e^{ik\theta}}{\sum_k L_k e^{ik\theta}} \frac{2\sin(k\theta/2)}{k\theta} \right| \leq 1 \tag{2.81}$$

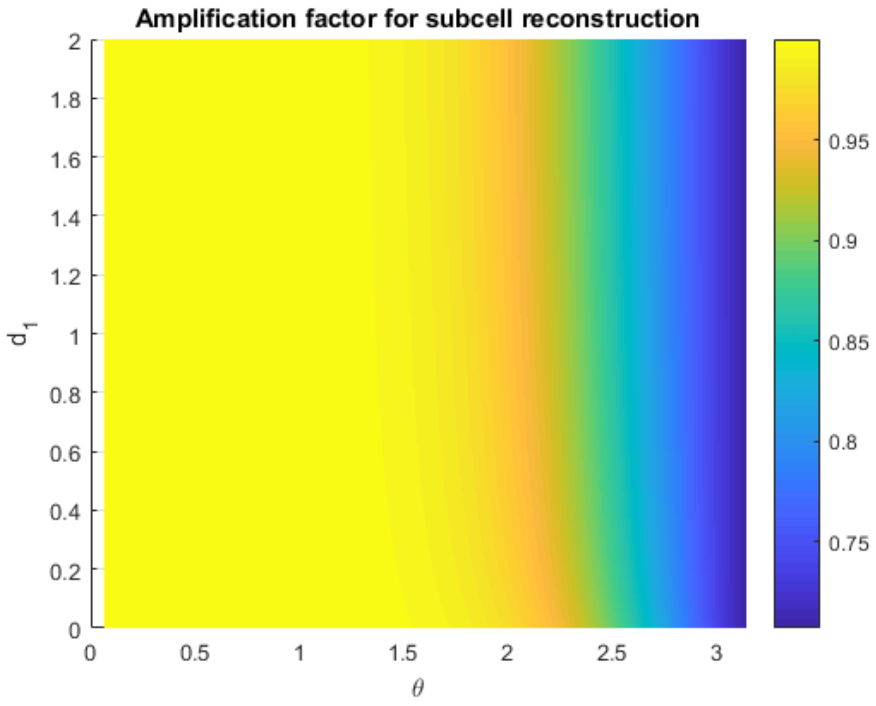

Figure 2.9: The subcell reconstruction is stable for all $d_1 \geq 0$ and depends weakly on d_1.

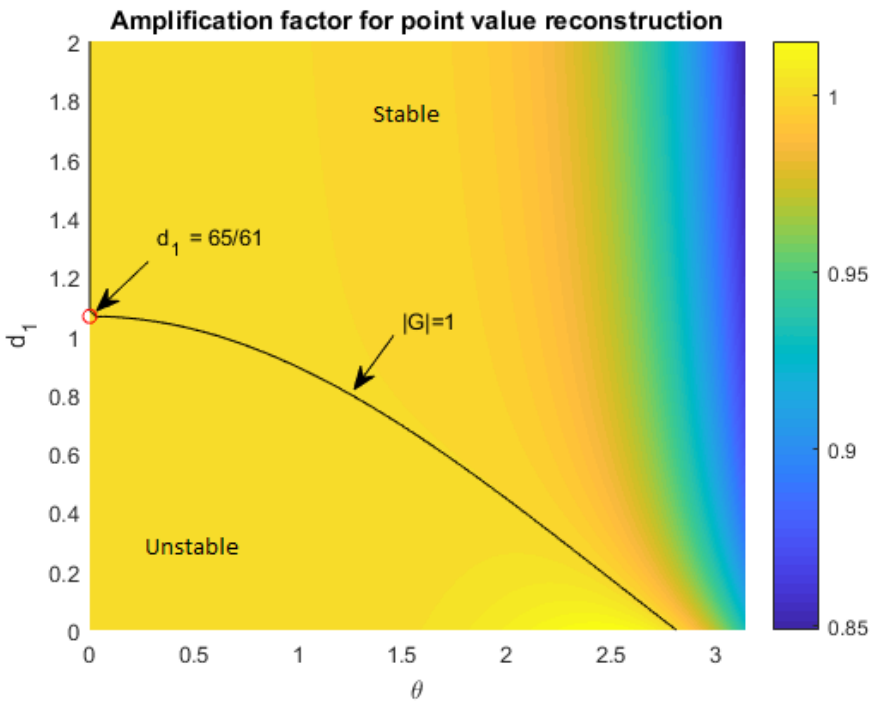

Figure 2.10: The point value reconstruction is unstable for $d_1 < 65/61$.

Figure 2.10 shows the amplification factor as a function of d_1 and θ. Also shown is the level curve $|G| = 1$, which divides the region of instability on the lower left from the region of stability. We see, therefore, that the critical value of d_1 is the value where this level curve intersects $\theta = 0$. Numerical evidence supports the conjecture that this intersection occurs at the circled point where $d_1 = 65/61$ which also gives the 8th-order point-value reconstruction. The instability in the region $0.6 < d_1 < 1$ is so mild that in some cases it may be unnoticeable. Experimentally, using $d_1 = 1.3$ gives stable results for all the test cases that will be considered in Chapter 3.

2.5.3 Conditioning of the System

Let c_i, $i = -1, 0, 1$ be the left-hand side coefficients of the combined scheme Eqs. (2.30)-(2.31) using ideal weights. Symmetry and normalization imply that $2c_{-1} = 2c_1 = c_0$, and Eqs. (2.28)-(2.30) can be used to express c_0 as:

$$c_0 = \frac{25 + 31d_1}{40 + 40d_1} \tag{2.82}$$

Note that for nonnegative d_1, $5/8 \leq c_0 < 31/40$. In particular, $c_0 < 1$. Now consider the fully periodic case, so that a non-compact boundary treatment is not required and the coefficient matrix M_p is tridiagonal except for nonzero entries in the lower-left and upper-right corners. The diagonal entries are c_0 and the off-diagonal entries are $c_1 = c_{-1}$, therefore M_p is symmetric. Since $c_0 > 0$, then by the Gershgorin circle theorem, it is also positive definite as long as it is strictly diagonally dominant which it is by design. Thus the L^2 condition number, equal to the ratio of the largest to the smallest singular value of M_p, is also equal to the ratio of the largest to smallest eigenvalues of M_p since its eigenvalues and singular values coincide. The symmetry of M_p ensures that these eigenvalues are all real, so the Gershgorin circle theorem implies that they all reside in the interval $[c_0 - 2|c_1|, c_0 + 2|c1|]$. But $2c_1 = 1 - c_0$ and positive d_1 implies $0 < c_0 < 1$, so we have the following upper bound on the condition number:

$$\kappa(M_p) \leq \frac{1}{2c_0 - 1} = \frac{20 + 20d_1}{5 + 11d_1} \tag{2.83}$$

(In fact, this inequality becomes an equality if the number of cells is even). M_p is strictly diagonally dominant when $d_1 > 0$, in which case $\kappa < 4$. In D dimensions,

the tensor-product construction of the scheme allows the coefficient matrix $M_p^{(D)}$ to be expressed as a Kronecker product of the one-dimensional coefficient matrix $M_p = M_p^{(1)}$:

$$M_p^{(D)} = \underbrace{M_p \otimes M_p \cdots \otimes M_p}_{D \text{ copies}} \tag{2.84}$$

Consider the singular value decomposition of M_p: $M_p = U\Sigma V^T$. Then we have also the following decomposition of $M_p^{(D)}$:

$$\begin{aligned} M_p^{(D)} &= (U\Sigma V^T) \otimes (U\Sigma V^T) \cdots \otimes (U\Sigma V^T) \\ &= (U \otimes \cdots \otimes U)(\Sigma \otimes \cdots \otimes \Sigma)(V^T \otimes \cdots \otimes V^T) \end{aligned} \tag{2.85}$$

where each group of Kronecker products includes D copies of the corresponding matrix. It then follows that the largest and smallest singular values of $M_p^{(D)}$ are respectively the largest and smallest singular values of M_p raised to the power D, from which we obtain the following bound on the condition number:

$$\kappa(M_p^{(D)}) \leq \left(\frac{20 + 20d_1}{5 + 11d_1}\right)^D < 4^D \tag{2.86}$$

Thus in the ideal periodic case the coefficient matrix is well-conditioned for problems of low- to moderate dimensionality and the conditioning improves as d_1 increases.

Accounting for non-periodic boundary conditions in the one-dimensional case leads to a M that is not symmetric:

$$M = \begin{bmatrix} 1 & 0 & & & \\ c_1 & c_0 & c_1 & & \\ & \ddots & \ddots & \ddots & \\ & & c_1 & c_0 & c_1 \\ & & & 0 & 1 \end{bmatrix} \tag{2.87}$$

Since M is not symmetric its singular values differ from its eigenvalues so we must consider the eigenvalues of MM^T to obtain the singular values of M. MM^T is symmetric and pentadiagonal with the following form:

$$MM^T = \begin{bmatrix} 1 & c_1 & 0 & & & & & & \\ c_1 & c_0^2+2c_1^2 & 2c_0c_1 & c_1^2 & & & & & \\ 0 & 2c_0c_1 & c_0^2+2c_1^2 & 2c_0c_1 & c_1^2 & & & & \\ 0 & c_1^2 & 2c_0c_1 & c_0^2+2c_1^2 & 2c_0c_1 & c_1^2 & & & \\ & & \ddots & \ddots & \ddots & \ddots & \ddots & & \\ & & & c_1^2 & 2c_0c_1 & c_0^2+2c_1^2 & 2c_0c_1 & c_1^2 & \\ & & & & c_1^2 & 2c_0c_1 & c_0^2+2c_1^2 & 2c_0c_1 & 0 \\ & & & & & c_1^2 & 2c_0c_1 & c_0^2+2c_1^2 & c_1 \\ & & & & & & 0 & c_1 & 1 \end{bmatrix} \tag{2.88}$$

MM^T is not diagonally dominant for arbitrary c_0. The intervals on the real axis that can contain the eigenvalues of MM^T are, by the Gershgorin circle theorem:

$$\begin{aligned} &\text{(A)} \quad [1-c_1, 1+c_1] = \left[\frac{1+c_0}{2}, \frac{3-c_0}{2}\right] \\ &\text{(B)} \quad [c_0^2+c_1^2-c_1-2c_0c_1, c_0^2+3c_1^2+2c_0c_1+c_1] \\ &\qquad = \left[\frac{1}{4}(3c_0-1)^2+\frac{1}{2}(c_0-1), \frac{1}{4}(c_0+1)^2+\frac{1}{2}(1-c_0)(2-c_0)\right] \\ &\text{(C)} \quad [c_0^2-4c_0c_1, c_0^2+4c_0c_1+4c_1^2] = [c_0(3c_0-2), 1] \end{aligned} \tag{2.89}$$

The lower bound on the eigenvalues needs to be positive which, given that $c_0 > 0$,

requires:

$$\begin{aligned} &\text{(A)} \quad \frac{1}{4}(c_0+1)^2 + \frac{1}{2}(1-c_0)(2-c_0) > 0 \Leftrightarrow c_0 \notin \left[\frac{1}{9}\left(2-\sqrt{13}\right), \frac{1}{9}\left(2+\sqrt{13}\right)\right] \\ &\text{(B)} \quad c_0(3c_0-2) > 0 \Leftrightarrow c_0 > \frac{2}{3} \end{aligned} \tag{2.90}$$

Eq. (2.82) ensures that $c_0 > (2+\sqrt{13})/9 = 0.622839\ldots$ so Eq. (2.90)A is automatically satisfied. The second condition Eq. (2.90)B, however, requires that $d_1 > 5/13$ in view of Eq. (2.82). We have already seen that choosing $d_1 > 5/13$ is required for the point-value reconstruction to be stable so this condition is not prohibitive. The relevant range of c_0 is therefore $2/3 < c_0 < 31/40$ and over this range the functions defining the endpoints in Eq. (2.89) do not intersect. The largest upper bound comes from Eq. (2.89)A and the smallest lower bound comes from Eq. (2.89)C. The upper bound on the condition number is therefore:

$$\kappa(M) \leq \sqrt{\frac{3-c_0}{2c_0(3c_0-2)}}, \quad \frac{2}{3} < c_0 < \frac{31}{40} \tag{2.91}$$

In terms of the diagonal excess d_1:

$$\kappa(M) \leq \sqrt{\frac{1780d_1^2 + 3680d_1 + 1900}{403d_1^2 + 170d_1 - 125}}, \quad d_1 > \frac{5}{13} \tag{2.92}$$

This bound approaches $\sqrt{1780/403} = 2.10163\ldots$ as $d_1 \to \infty$. Using the recommended diagonal excess $d_1 = 1.3$ gives the bound as $\kappa(M) \leq 3.53167\ldots$. The same Kronecker-product procedure used in the periodic case gives the corresponding condition number bound for multidimensional reconstructions with boundary conditions.

The lower bound obtained with the Gershgorin theorem is substantially lower than the smallest eigenvalues actually computed, particularly when d_1 is small.

Numerical experiments strongly suggest that the eigenvalues of MM^T are bounded below by $(2c_0 - 1)^{-2}$ and bounded above by a number approximately equal to 1.05 regardless of d_1, which gives a bound on the condition number that is only about 5% larger than the corresponding bound in the periodic case. Qualitatively, this is the behavior one would expect since M is almost symmetric and MM^T is almost a Toeplitz matrix. Unfortunately the structure of MM^T is not amenable to deriving these bounds analytically.

A similar analysis cannot be performed for the case where the weights are not fixed at their ideal values because, although the ideal weights are computed in the tensor-product fashion, the non-oscillatory weights depend on the multidimensional indicators which are not calculated as tensor-products of one-dimensional indicators. In practice, however, with non-oscillatory weights the condition number stays well within an order of magnitude of its value when ideal weights are used.

2.6 Dual-Grid Formulation

It was mentioned in Section 1.3.2 that central schemes with fixed-size staggered cells incur a numerical dissipation that scales as $\mathcal{O}(\Delta t^{-1})$, leading to excessive smearing of sharp gradients and precluding the possibility of a semi-discrete form to be used with Runge-Kutta methods for time advancement. The present central CRWENO scheme has this deficiency as well. Whereas the modification suggested by Kurganov and Tadmor in [13] solves this problem if the scheme produces a functional representation of the solution over the whole cell, in the present case only the

subcell averages and the midpoint value are ever computed. Furthermore, using the variable and asymmetric cell sizes that depend on local wave speeds described in [13] may destroy the precarious situation in which the coefficients and ideal weights for the two reconstructions can be made to coincide.

Fortunately, an alternative discretization is available that leads to a dissipation independent of Δt and therefore a semi-discrete form. Y. Liu in [37] introduced the idea of reconstructing the same solution on the main and staggered grids simultaneously as follows. Let $\bar{u}_j$ denote, as usual, the average of the solution over the main-grid cell $[x_{j-1/2}, x_{j+1/2}]$ and u_j the value of the solution $u(x)$ at point x_j, where $u(x)$ reproduces the cell averages $\bar{u}_j$. Now denote also by $\bar{v}_{j-1/2}$ the average of the auxiliary solution $v(x)$ over the staggered cell $[x_{j-1}, x_j]$ and let $v_{j-1/2}$ be its point value at $x_{j-1/2}$. Then for each grid, a forward Euler approximation to the solution at the next time step involves the fluxes at midpoint values from the other grid and a convex combination of the two available representations of the average over the cell in question.

$$\begin{aligned} \bar{u}_j^{n+1} &= \theta \left(\frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_{j+1/2}} v(x)\, dx \right) + (1-\theta)\bar{u}_j^n - \frac{1}{\Delta x}(f(v_{j+1/2}^n) - f(v_{j-1/2}^n)) \\ \bar{v}_{j-1/2}^{n+1} &= \theta \left(\frac{1}{\Delta x} \int_{x_{j-1}}^{x_j} u(x)\, dx \right) + (1-\theta)\bar{v}_{j-1/2}^n - \frac{1}{\Delta x}(f(u_j^n) - f(u_{j-1}^n)) \end{aligned} \tag{2.93}$$

Liu's approach is to let the combination parameter θ depend on the time step: $\theta = \Delta t/\Delta\tau$ where $\Delta\tau$ is an upper bound on the permissible time step. With this

choice Eq. (2.93) leads directly to the semi-discrete form:

$$\begin{aligned}\frac{d\bar{u}_j}{dt} &= \frac{1}{\Delta\tau}\left(\frac{1}{\Delta x}\int_{x_{j-1/2}}^{x_{j+1/2}} v(x)\,dx - \bar{u}_j\right) - \frac{\Delta t}{\Delta x}(f(v^n_{j+1/2}) - f(v^n_{j-1/2})) \\ \frac{d\bar{v}_{j-1/2}}{dt} &= \frac{1}{\Delta\tau}\left(\frac{1}{\Delta x}\int_{x_{j-1}}^{x_j} u(x)\,dx - \bar{v}_{j-1/2}\right) - \frac{\Delta t}{\Delta x}(f(u^n_j) - f(u^n_{j-1}))\end{aligned} \tag{2.94}$$

from which one sees that the discrepancy between the two representations of the solution provides dissipation. More importantly, the dissipative terms can be calculated from the subcell averages and the flux terms from the point values, so the CCRWENO scheme already developed can be directly transplanted into this framework to obtain a semi-discrete method. This would allow a smaller time step thus less numerical dissipation, an expectation which will be tested in Chapter 3. Figure 2.11 diagrams one time step of a dual-grid central scheme. Computational expense can be avoided if the non-oscillatory weights are calculated once per time step and not once per Runge-Kutta stage.

One might wonder whether the dissipative term in Eq. (2.94) is truly necessary, or if the point value reconstruction alone provides enough dissipation on its own. If so, then the subcell reconstruction could be entirely avoided which would not only avoid computational expense, but also allow more freedom in designing point-value schemes since matching weights and coefficients would be unnecessary. Indeed, the truncation error coefficients in Table 2.2 imply that an eighth-order scheme with positive ideal weights would be available. In the semi-discrete case this is possible, but such a scheme cannot be stable in the fully-discrete case.

Proposition 2.1. *Let $f(q) = aq$, $a > 0$, be the flux for linear advection and consider a point-value reconstruction $u_j = \sum_k c_k \bar{u}_{j+k}$ with c_k constant. Then the semi-*

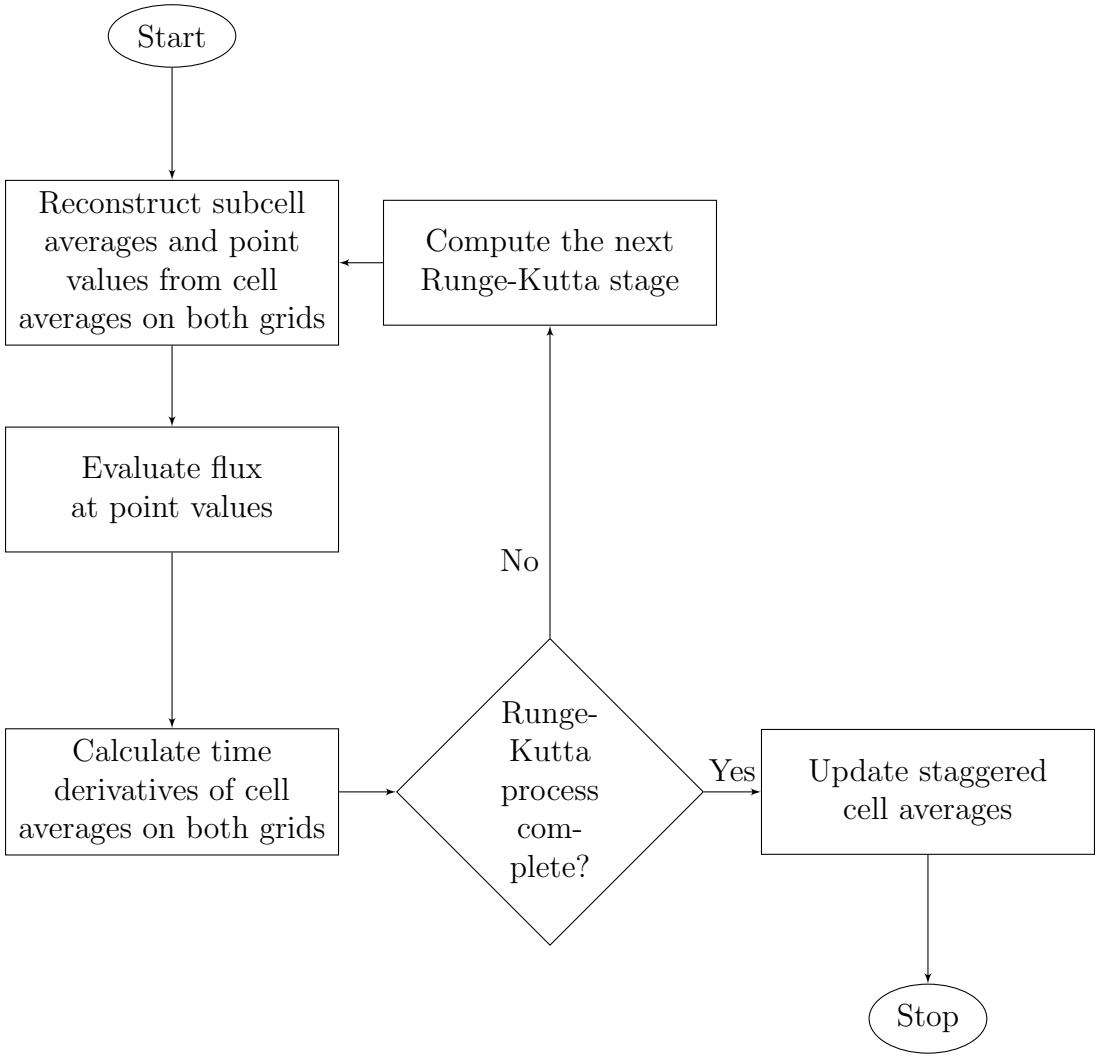

Figure 2.11: Flowchart of one time step of a dual-grid central scheme.

discrete scheme given by

$$\begin{aligned} \frac{d\bar{u_j}}{dt} &= -\frac{\Delta t}{\Delta x}(f(v^n_{j+1/2}) - f(v^n_{j-1/2})) \\ \frac{dv_{j-1/2}^{-}}{dt} &= -\frac{\Delta t}{\Delta x}(f(u^n_j) - f(u^n_{j-1})) \end{aligned} \tag{2.95}$$

with periodic boundaries is L^2-stable if the circulant matrix R given by $R_{i,j} = c_{j-i \mod N}$ (N being the number of cells on the main grid) is symmetric, in which case it is neutrally stable.

Proof. Let U and V denote the vectors of cell averages on the main and staggered grids, respectively. Both are of length N since the periodic boundary ensures that the staggered cells centered on the first and last interfaces have the same solution, so we need only track one. Then the scheme Eq. (2.95) can be written as:

$$\frac{d}{dt}\begin{bmatrix} U \\ V \end{bmatrix} = \frac{-a}{\Delta x}\begin{bmatrix} 0 & RD^+ \\ RD^- & 0 \end{bmatrix}\begin{bmatrix} U \\ V \end{bmatrix} = -\frac{a}{\Delta x}A\begin{bmatrix} U \\ V \end{bmatrix} \tag{2.96}$$

where $D^{\pm}$ are the forward and backward difference matrices:

$$D^+ = \begin{bmatrix} -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & & \vdots \\ \vdots & & \ddots & \ddots & 0 \\ 0 & 0 & \cdots & -1 & 1 \\ 1 & 0 & \cdots & 0 & -1 \end{bmatrix}, \; D^- = \begin{bmatrix} 1 & 0 & 0 & \cdots & -1 \\ -1 & 1 & 0 & & \vdots \\ \vdots & \ddots & \ddots & & 0 \\ 0 & \cdots & -1 & 1 & 0 \\ 0 & 0 & \cdots & -1 & 1 \end{bmatrix} \tag{2.97}$$

Note that $D^+ = -(D^-)^T$. For stability we ask that the L^2 norm of $[U^T, V^T]$ not increase:

$$0 \geq \frac{1}{2}\frac{d}{dt}\left([U^T, V^T]\begin{bmatrix} U \\ V \end{bmatrix}\right) = \frac{-a}{\Delta x}[U^T, V^T]A\begin{bmatrix} U \\ V \end{bmatrix} \tag{2.98}$$

So positive semidefiniteness of A implies stability. This is equivalent to positive semidefiniteness of the symmetric part $\bar{A}$ of A:

$$\begin{aligned}\bar{A} &= \frac{1}{2}(A + A^T) = \frac{1}{2}\begin{bmatrix} 0 & RD^+ + (RD^-)^T \\ RD^- + (RD^+)^T & 0 \end{bmatrix} \\ &= \frac{1}{2}\begin{bmatrix} 0 & RD^+ - D^+R^T \\ RD^- - D^-R^T & 0 \end{bmatrix}\end{aligned} \tag{2.99}$$

Since $D^{\pm}$ and the reconstruction matrix R are all circulant they commute with each other, so we have:

$$\bar{A} = \frac{1}{2}\begin{bmatrix} 0 & D^+(R - R^T) \\ D^-(R - R^T) & 0 \end{bmatrix} \tag{2.100}$$

For this matrix to be positive semidefinite requires:

$$\begin{aligned} 0 &\leq [U^T, V^T]\begin{bmatrix} 0 & D^+(R - R^T) \\ (D^+(R - R^T))^T & 0 \end{bmatrix}\begin{bmatrix} U \\ V \end{bmatrix} \\ &= U^T(D^+(R - R^T))V + V^T(D^+(R - R^T))^T U \\ &= 2U^T(D^+(R - R^T))V \end{aligned} \tag{2.101}$$

for any vectors U and V whose elements have the same sum (because both U and V represent the same solution). This is only possible if every entry of $(D^+(R - R^T))$ is non-negative, since otherwise one may choose U and V to have all elements zero except one which is 1 to isolate a negative entry of $D^+(R - R^T)$. $(D^+(R - R^T)) \geq 0$ if and only if the entries along each column of $R - R^T$ are non-decreasing and the last entry is less than or equal to the first. This is only possible if all entries along a column of $R - R^T$ are equal, but its diagonal entries are necessarily all zero due to its being antisymmetric so it must be that $R - R^T = 0 \iff R = R^T$.

In this case, the matrix $\bar{A} = 0$ so A is antisymmetric, from which it follows that $\frac{d}{dt}(U^T U + V^T V) = 0$. □

The simplest reconstruction matrix arises from the second-order approximation of point values by cell averages, $u_j = \bar{u}_j + \mathcal{O}(\Delta x^2)$, in which case $R = I$. In view of this result one expects the time discretization to ruin the precarious neutral stability, which is indeed the case as we now show in the case of forward Euler time-stepping.

Proposition 2.2. *Let $f(q) = aq$, $a > 0$, be the flux for linear advection and consider a point-value reconstruction $u_j = \sum_k c_k \bar{u}_{j+k}$ with c_k constant. Then the fully-discrete scheme given by*

$$\begin{aligned} u_j^{n+1} &= u_j^n - \frac{\Delta t}{\Delta x}(f(v_{j+1/2}^n) - f(v_{j-1/2}^n)) \\ v_{j-1/2}^{n+1} &= v_{j-1/2}^n - \frac{\Delta t}{\Delta x}(f(u_j^n) - f(u_{j-1}^n)) \end{aligned} \tag{2.102}$$

with periodic boundaries is unconditionally unstable.

Proof. Keeping the same notation as in Proposition 2.1, the evolution over one time step can be written:

$$\begin{bmatrix} U^{n+1} \\ V^{n+1} \end{bmatrix} = \begin{bmatrix} U^n \\ V^n \end{bmatrix} - \sigma \begin{bmatrix} 0 & RD^+ \\ RD^- & 0 \end{bmatrix} \begin{bmatrix} U^n \\ V^n \end{bmatrix} = \begin{bmatrix} I & -\sigma RD^+ \\ -\sigma RD^- & I \end{bmatrix} \begin{bmatrix} U^n \\ V^n \end{bmatrix} = M \begin{bmatrix} U^n \\ V^n \end{bmatrix} \tag{2.103}$$

where $\sigma = a\Delta t/\Delta x$ is the CFL number. For convenience we write $A = RD^+, B =$

RD^-. Then for stability we want the L^2 norm to be non-increasing:

$$\begin{bmatrix} U^n \\ V^n \end{bmatrix}^T \begin{bmatrix} U^n \\ V^n \end{bmatrix} \geq \begin{bmatrix} U^{n+1} \\ V^{n+1} \end{bmatrix}^T \begin{bmatrix} U^{n+1} \\ V^{n+1} \end{bmatrix} = \begin{bmatrix} U^n \\ V^n \end{bmatrix}^T M^T M \begin{bmatrix} U^n \\ V^n \end{bmatrix}$$
$$= \begin{bmatrix} U^n \\ V^n \end{bmatrix}^T \begin{bmatrix} U^n \\ V^n \end{bmatrix} + \begin{bmatrix} U^n \\ V^n \end{bmatrix}^T (M^T M - I) \begin{bmatrix} U^n \\ V^n \end{bmatrix} \tag{2.104}$$

Therefore we need the matrix $M^T M - I$ to be negative semidefinite. We have:

$$M^T M - I = \sigma \begin{bmatrix} \sigma B^T B & -A - B^T \\ -A^T - B & \sigma A^T A \end{bmatrix} \tag{2.105}$$

And we desire:

$$0 \geq \begin{bmatrix} U \\ V \end{bmatrix} (M^T M - I) \begin{bmatrix} U \\ V \end{bmatrix} = \sigma^2 (U^T B^T B U + V^T A^T A V) - 2\sigma U^T (A + B^T) V \tag{2.106}$$

$$2\sigma U^T (A + B^T) V \geq \sigma^2 (U^T B^T B U + V^T A^T A V) \tag{2.107}$$

for any U, V whose elements have the same sum. Clearly the right-hand side of (2.107) is always non-negative. Because U and V are arbitrary, however, they can always be chosen so that the left-hand side is strictly negative unless every entry of $A + B^T$ is non-negative. But $A + B^T = RD^+ - D^+ R^T = D^+(R - R^T)$, since the matrices are all circulant thus commute with each other. Stability therefore requires that every entry of $D^+(R - R^T)$ be non-negative, which by the reasoning in Proposition 2.1 occurs if and only if $R = R^T$. In that case, however, $A + B^T = 0$ so the left-hand side of Eq. (2.107) would vanish for all U, V whereas the right-hand side can be made strictly positive. Therefore any linear dual-grid scheme is unconditionally unstable with explicit Euler time advancement. □

In practice, using a WENO-type reconstruction appears to provide some dissipation due to the weight adaptation process which quells the instability for short times, but it eventually appears even for smooth solutions. If a discontinuity is present the instability immediately appears and becomes catastrophic. The schemes Eq. (2.95) and Eq. (2.102) can be viewed as the limiting case of Eq. (2.94) as $\Delta\tau \to \infty$, which suggests that better stability will be obtained when $\Delta\tau$ is the smallest feasible upper bound on the time step Δt.

Chapter 3: CCRWENO Numerical Results

In this chapter we apply the CCRWENO method developed in Chapter 2 to a suite of test problems. Unless otherwise indicated, all results are obtained using the Jiang-Shu formulation [15] of the non-oscillatory weights.

3.1 1-Dimensional Tests

3.1.1 Convergence

First we establish that CCRWENO converges at fifth order for a simple case, linear advection of a low-frequency sinusoid through a periodic domain. The domain is $[0, 1]$ and the initial condition is:

$$q(x, t = 0) = \sin(2\pi x) \tag{3.1}$$

which evolves according to Eq. (1.49) with propagation speed $a = 1$. Table 3.1 lists the errors obtained on successively refined grids after one period of advection. The maximum error decreases more quickly than fifth-order whereas the L^1 error decreases at fifth-order.

Repeating the same test with a nonlinear flux and a higher-frequency sinusoid

Table 3.1: CCRWENO errors in 1D linear advection of a sinusoid.

N_1	L^∞ error	L^∞ error order	L^1 error	L^1 error order
50	4.5315×10^{-5}	-	1.8589×10^{-5}	-
100	1.7596×10^{-6}	4.69	5.6671×10^{-7}	5.04
200	6.7411×10^{-8}	4.71	1.7717×10^{-8}	5.00
400	2.0849×10^{-9}	5.01	5.4400×10^{-10}	5.03
800	4.3678×10^{-11}	5.58	1.6575×10^{-11}	5.04

also gives fifth-order convergence. The initial condition is:

$$q(x, t = 0) = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} = \begin{bmatrix} 1 + 0.2\sin(8\pi x) \\ \rho \\ \frac{P}{\gamma - 1} + \frac{1}{2}\rho \end{bmatrix} \tag{3.2}$$

where the pressure $P = 1$ is constant, the ratio of specific heats $\gamma = 1.4$ is constant, and the velocity $u = 1$ is also constant. The solution evolves according to the one-dimensional Euler equations Eq. (1.10) for one period. The domain is again $[0, 1]$ with periodic boundaries. Table 3.2 confirms that CCRWENO converges at fifth order.

Table 3.2: CCRWENO errors in 1D density wave advection.

N_1	L^∞ error	L^∞ error order	L^1 error	L^1 error order
50	1.3987×10^{-2}	-	9.7308×10^{-3}	-
100	6.6739×10^{-4}	4.39	4.3670×10^{-4}	4.48
200	2.4283×10^{-5}	4.78	1.3382×10^{-5}	5.03
400	7.9087×10^{-7}	4.94	4.1199×10^{-7}	5.02
800	2.3266×10^{-8}	5.09	1.2512×10^{-8}	5.04

Applied to Euler advection of a density wave, CCRWENO is more efficient than upwind WENO and CRWENO with characteristic variables. Despite using a four-stage Runge-Kutta scheme compared to the three-stage SSP Runge-Kutta scheme [23] used with the upwind schemes, the cost per cell of CCRWENO is lower due to

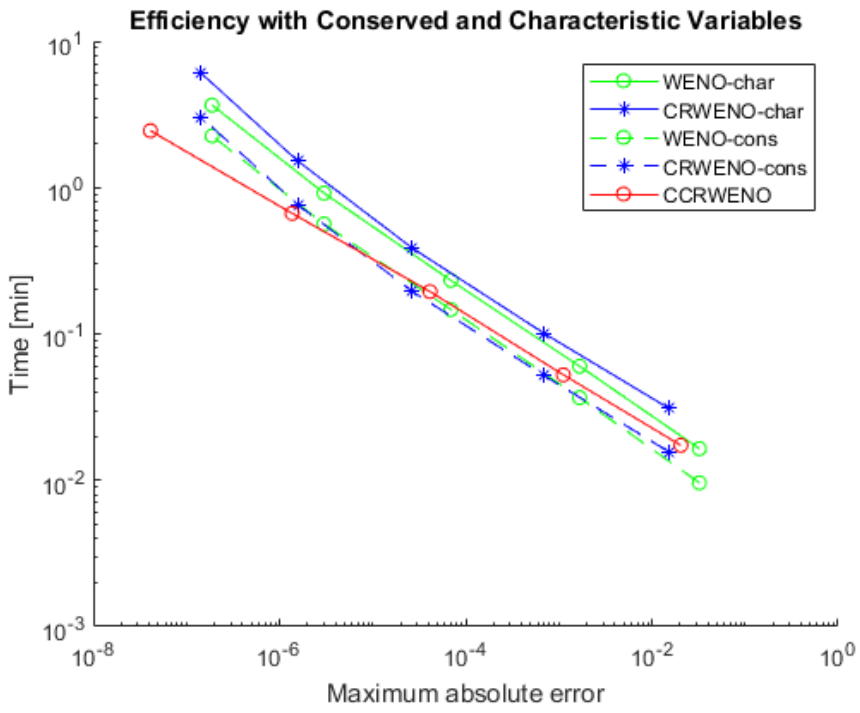

Figure 3.1: CCRWENO outperforms the characteristic upwind schemes and on sufficiently fine grids also outperforms the upwind schemes without characteristic variables.

the absence of any Riemann solver. As a result, for sufficiently fine grids CCRWENO also outperforms the two upwind schemes without characteristic variables. Figure 3.1 demonstrates this behavior.

3.1.2 Riemann Problems

Several classical test cases take the form of Riemann problems, which serve to demonstrate the ability to solve Riemann problems without Riemann solvers and to do so without spurious oscillations. We begin with the Riemann problem of Sod [35], in which the initial discontinuity produces a contact discontinuity flanked

by a rarefaction and a shock. The initial conditions are:

$$[\rho, u, P] = \begin{cases} [1, 0, 1] & x \leq 1 \\ [0.125, 0, 0.1] & x > 1 \end{cases} \tag{3.3}$$

The contact discontinuity provides most of the challenge in this problem, since its development out of the initial discontinuity can be accompanied by spurious oscillations. Figure 3.2 shows part of the solutions obtained with $\lambda = 0.1$ at time $t = 0.2$. Compared to CWENO4, the CCRWENO solution is steeper near the discontinuities at the cost of larger oscillations there, though those oscillations do decrease in amplitude as the grid is refined. Observing the solution as it evolves shows that the initial oscillations are larger with CCRWENO and diminish less rapidly than in the solution from CWENO4. The initial larger size likely results from improper coupling in the first few steps. As the contact discontinuity and the shock separate, there is a time step in which each subscheme contains a discontinuity in its stencil, causing the weights to have similar magnitudes even though the solution is discontinuous. Once the oscillations are established, they are propagated as smooth solutions which experience less damping due to the lower numerical dissipation of CCRWENO.

A second Riemann problem is that of Lax. Qiu and Shu in [29] use the Lax problem to test the non-oscillatory behavior of their fifth-order central WENO scheme, and it was the results of the tests with this problem that led them to suggest incorporating a characteristic decomposition into the subcell reconstruction. We

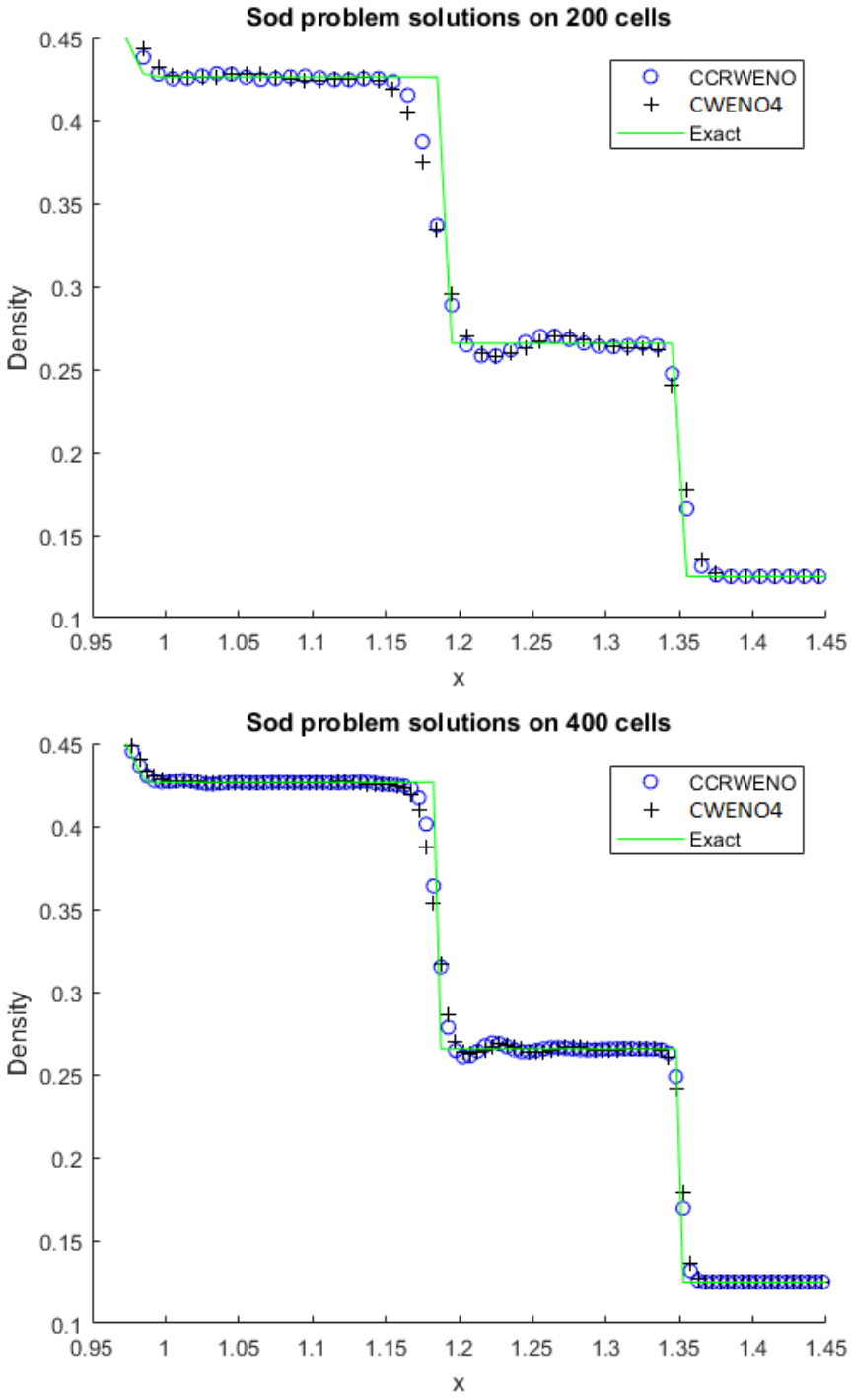

Figure 3.2: Solutions to the Sod problem near discontinuities. Top: 200 cells. Bottom: 400 cells. Circles: CCRWENO solution. Crosses: Solution by CWENO4. Solid line: Exact solution.

repeat their tests here with the CCRWENO scheme. The initial conditions are:

$$[\rho, u, P] = \begin{cases} [0.445, 0.698, 3.528] & x \leq 0 \\ [0.5, 0, 0.571] & x > 0 \end{cases} \tag{3.4}$$

As in the Sod problem, the challenge is to resolve the discontinuity (a shock this time) without incurring oscillations. Figure 3.3 shows part of the solutions obtained with $\lambda = 0.038$ at time $t = 0.16$. Both the CWENO4 and CCRWENO results show oscillations of similar magnitude near the contact discontinuity, with the CCRWENO scheme producing steeper discontinuities. The oscillations decrease in amplitude as the grid is refined. Compared with Figure 3 in [29], CCRWENO produces oscillations of similar magnitude and frequency to CWENO5 as well.

3.1.3 Shu-Osher Problem

A high-order scheme should resolve small-scale features in addition to being non-oscillatory. In the Shu-Osher problem, a shock encounters a density wave producing a wave train behind the shock. Thus the problem contains both a discontinuity and high-frequency waves in close proximity. The initial conditions are:

$$[\rho, u, P] = \begin{cases} \left[\frac{27}{7}, \frac{4\sqrt{35}}{9}, \frac{31}{3}\right] & x \leq 1 \\ [1 + 0.2\sin(5x), 0, 1] & x > 1 \end{cases} \tag{3.5}$$

Figure 3.4 shows the solutions obtained with $\lambda = 0.038$ at time $t = 1.8$ in the region containing the small-scale features. Both CWENO4 and CCRWENO fail to resolve the oscillations on 200 cells, but detect them on 400 cells with CCRWENO

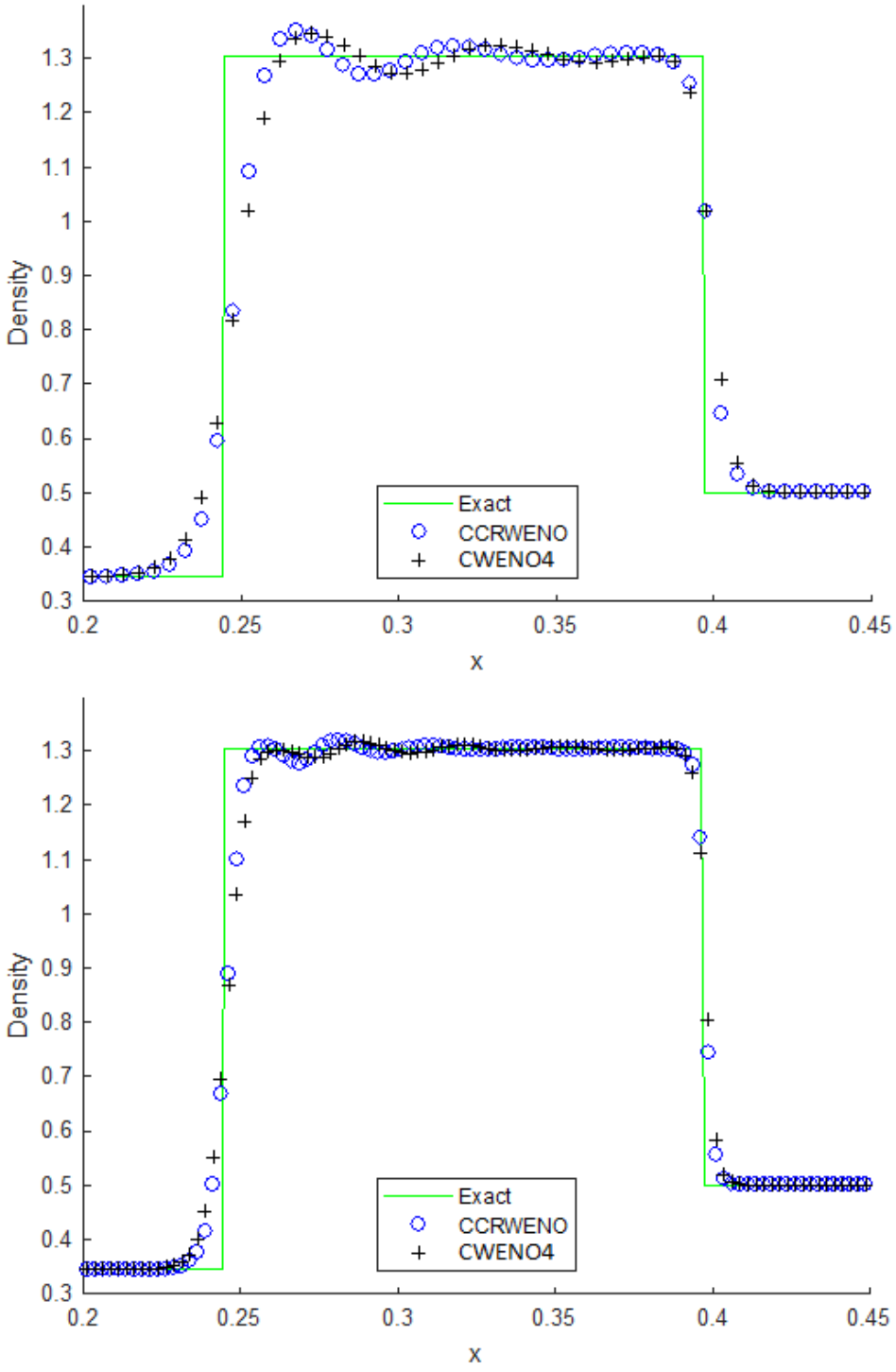

Figure 3.3: Solutions to the Lax problem near discontinuities. Top: 200 cells. Bottom: 400 cells. Circles: CCRWENO solution. Crosses: Solution by CWENO4. Solid line: Exact solution.

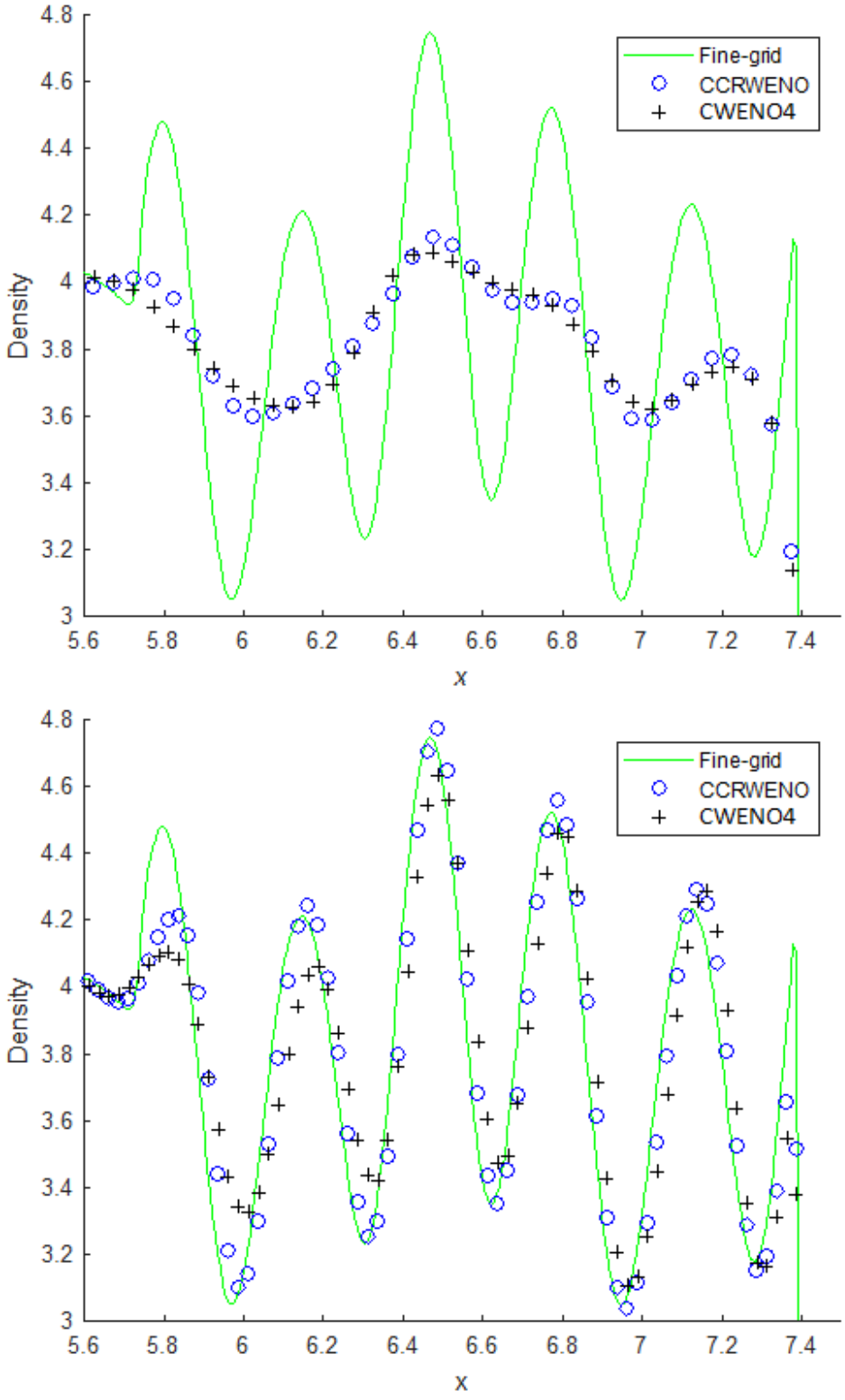

Figure 3.4: Solutions to the Shu-Osher problem near discontinuities. Top: 200 cells. Bottom: 400 cells. Circles: CCRWENO solution. Crosses: Solution by CWENO4. Solid line: Exact solution.

matching the amplitudes much more closely. No spurious oscillations appear due to the shock in either case.

3.2 2-Dimensional Tests

3.2.1 Convergence

First we verify fifth-order convergence in the case of a linear flux. Consider a sinusoidal solution advecting through a unit square with periodic boundary, with constant velocity oriented along a diagonal of the square. The initial condition is:

$$u(x, y, 0) = \sin^2(\pi x) \sin^2(\pi y) \tag{3.6}$$

The governing equation is:

$$\frac{\partial u}{\partial t} + \frac{\partial u}{\partial x} + \frac{\partial u}{\partial y} = 0 \tag{3.7}$$

The CCRWENO stencil parameter $d_1 = 1.3$ and the mesh ratio $\lambda = \Delta t/\Delta x = 0.35$. Time steps are taken until $t = 1$ and the solution is resolved on the main grid. Table 3.3 shows the error behavior for CCRWENO applied to this problem.

Table 3.3: CCRWENO errors in 2D linear advection of a sinusoid.

N_1	N_2	L^∞ error	L^∞ error order	L^1 error	L^1 error order
20	20	3.26×10^{-4}	-	7.87×10^{-5}	-
40	40	1.04×10^{-5}	4.96	2.27×10^{-6}	5.12
80	80	3.27×10^{-7}	5.00	6.92×10^{-8}	5.03
160	160	9.48×10^{-9}	5.11	2.13×10^{-9}	5.02
320	320	2.62×10^{-10}	5.18	6.63×10^{-11}	5.00

Since the solution is smooth and an exact solution is available, we can compare the efficiency of CCRWENO to those of CWENO4 and the tensor-product extension of CWENO5 (see [29]) by considering the computation time required for each to achieve a given error, as in Figure 3.5. As expected, CCRWENO requires more time on a given grid but produces less error. This speed advantage disappears as the grid

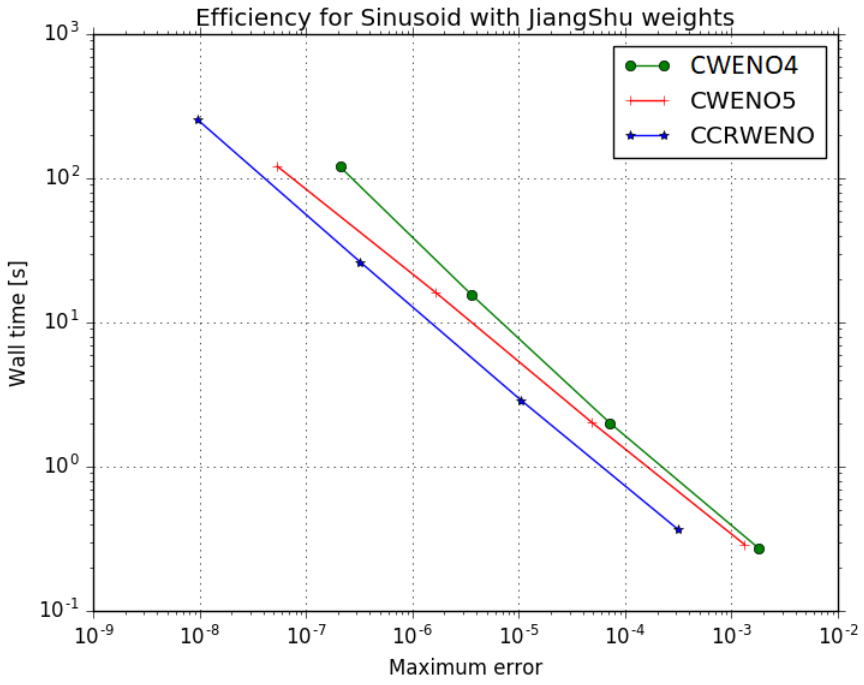

Figure 3.5: Efficiency of CCRWENO and CWENO4 for linear wave advection. Markers correspond to the grids in Table 3.3.

is refined due to the expense of solving a linear system, but the error incurred by CCRWENO is sufficiently smaller than those produced by the other schemes that it remains the most efficient choice.

To verify fifth-order convergence with a nonlinear flux we consider a moving isentropic vortex. The isentropic vortex is a smooth exact solution of the two-dimensional Euler equations for an ideal gas. The domain is a square of side length 10, with the vortex initially centered. The freestream velocity at which the vortex travels is $(u_\infty, v_\infty) = (1, 5.5)$. Note that the vortex does not travel in a coordinate

direction.

$$\begin{bmatrix} \rho \\ u \\ v \\ P \end{bmatrix} = \begin{bmatrix} \rho_\infty \left(1 - \frac{(\gamma-1)w^2}{2\gamma}\right)^{\frac{1}{\gamma-1}} \\ u_\infty - (y-5)w \\ v_\infty + (x-5)w \\ \rho^\gamma \end{bmatrix} \tag{3.8}$$

$$w = \frac{b}{2\pi}\exp\left(\frac{1-r^2}{2}\right),\ r = \sqrt{(x-5)^2 + (y-5)^2}$$

The ratio of specific heats is set to its value for air, $\gamma = 1.4$ and the freestream density is $\rho_\infty = 1$. The vortex strength is $b = 5$. The mesh ratio is $\lambda = 0.05$. Table 3.4 shows the error achieved at time $t = 10$ and confirms fifth-order convergence. Using the exact solution, we can again compare the computational efficiency yielding the results in Figure 3.6. CCRWENO is again more efficient than CWENO4 and CWENO5 apart from on one of the test grids, which appears to be a fluke owing to cache effects.

Table 3.4: CCRWENO errors in isentropic vortex advection.

N_1	N_2	L^∞ error	L^∞ error order	L^1 error	L^1 error order
40	40	3.77×10^{-2}	-	1.94×10^{-1}	-
80	80	7.92×10^{-4}	5.57	5.61×10^{-3}	5.11
120	120	9.90×10^{-5}	5.13	8.31×10^{-4}	4.71
160	160	2.27×10^{-5}	5.12	2.10×10^{-4}	4.79
200	200	7.23×10^{-6}	5.12	7.06×10^{-5}	4.87

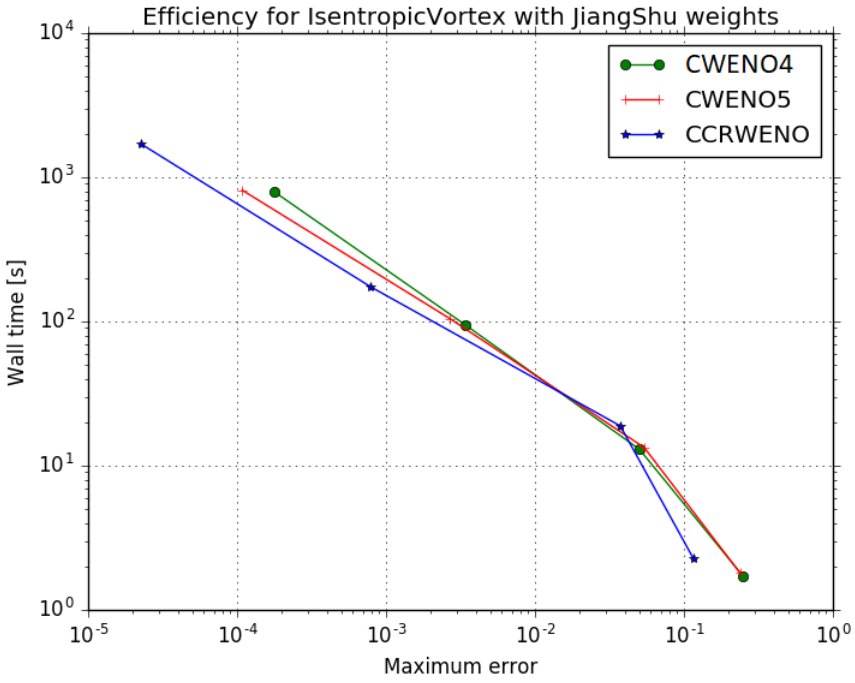

Figure 3.6: CCRWENO outperforms other central schemes in isentropic vortex advection. Markers correspond to the grids in Table 3.4.

3.2.2 Riemann Problems

Two-dimensional Riemann problems test the ability of the tensor-product extension of the one-dimensional CCRWENO to reproduce phenomena that are inherently two-dimensional, as well as the ability to resolve Riemann fans without any kind of Riemann solver. Lax and Liu [38] present several Riemann problems based on [39]. Figures 3.7-3.9 show the results obtained with CCRWENO for a selection of the configurations in [38] each on a grid of 400×400 cells. In some cases the solution is resolved without visible oscillations as desired, but in others a discontinuity produces a train of oscillations. In every case the key features of each solution are present. In particular, the roll-ups of contact discontinuities are clearly resolved and shock interactions emerge correctly. The oscillations, however, pose a serious drawback and will be addressed in Chapter 4. The results for Configurations 5 and 16 improve on the results in [40] because the flux derivative reconstruction in the current implementation calculates new one-dimensional smoothness indicators instead of reusing the two-dimensional indicators computed for the subcell and point value reconstructions, as was done for the tests in [40].

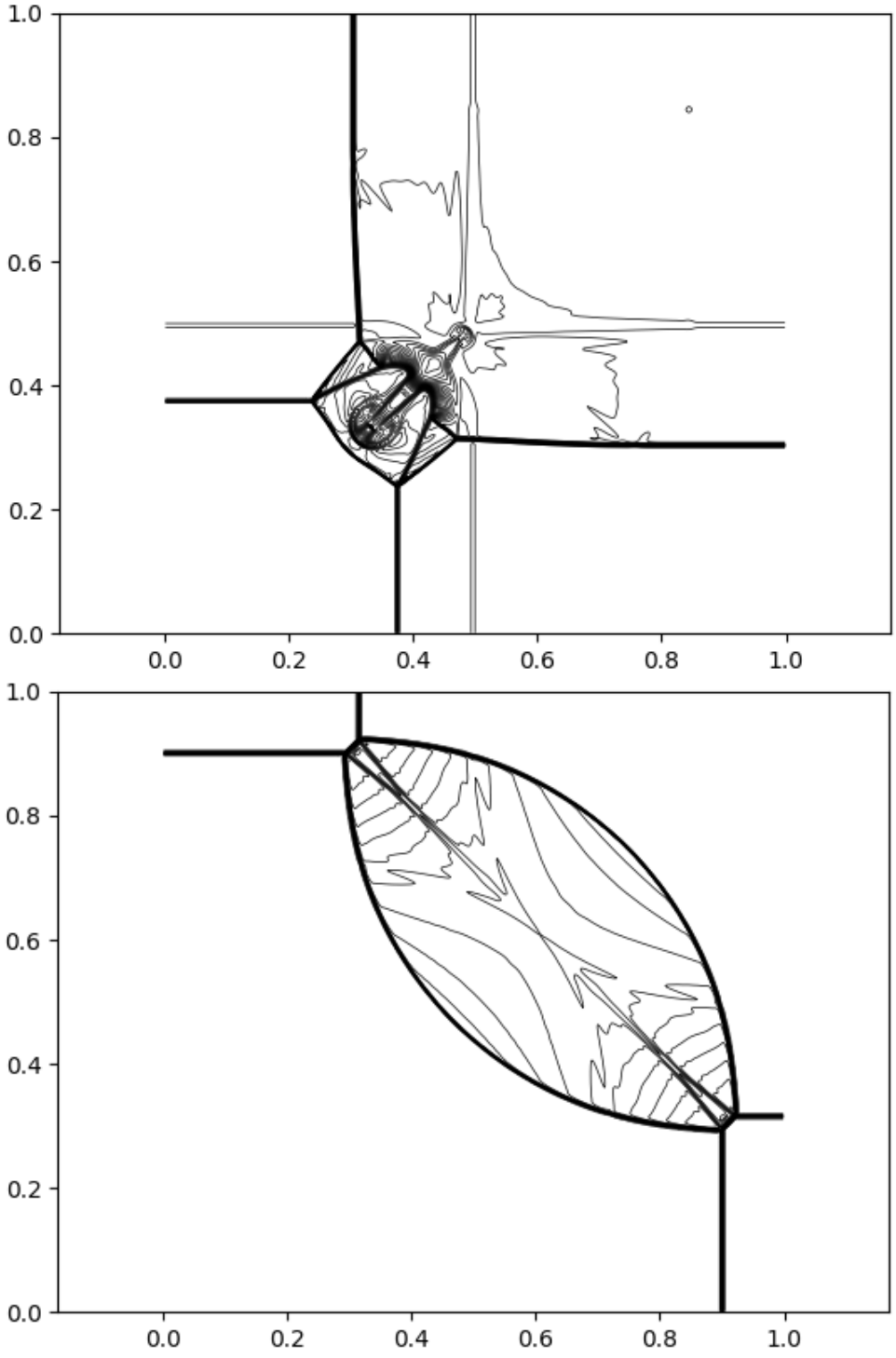

Figure 3.7: Density contours of the CCRWENO solutions of Configurations 3 (top) and 4 (bottom) from [38].

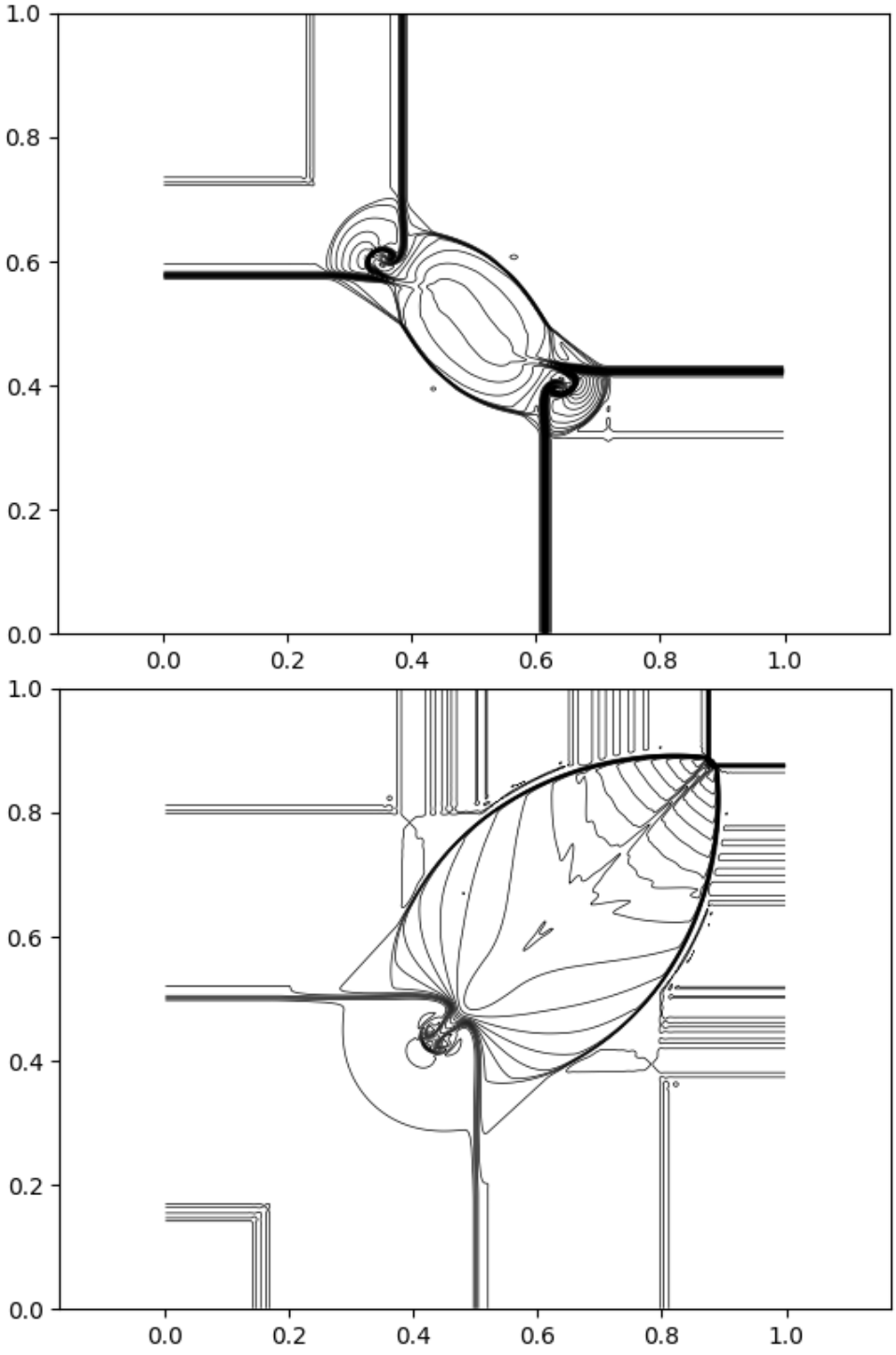

Figure 3.8: Density contours of the CCRWENO solutions of Configurations 5 (top) and 12 (bottom) from [38].

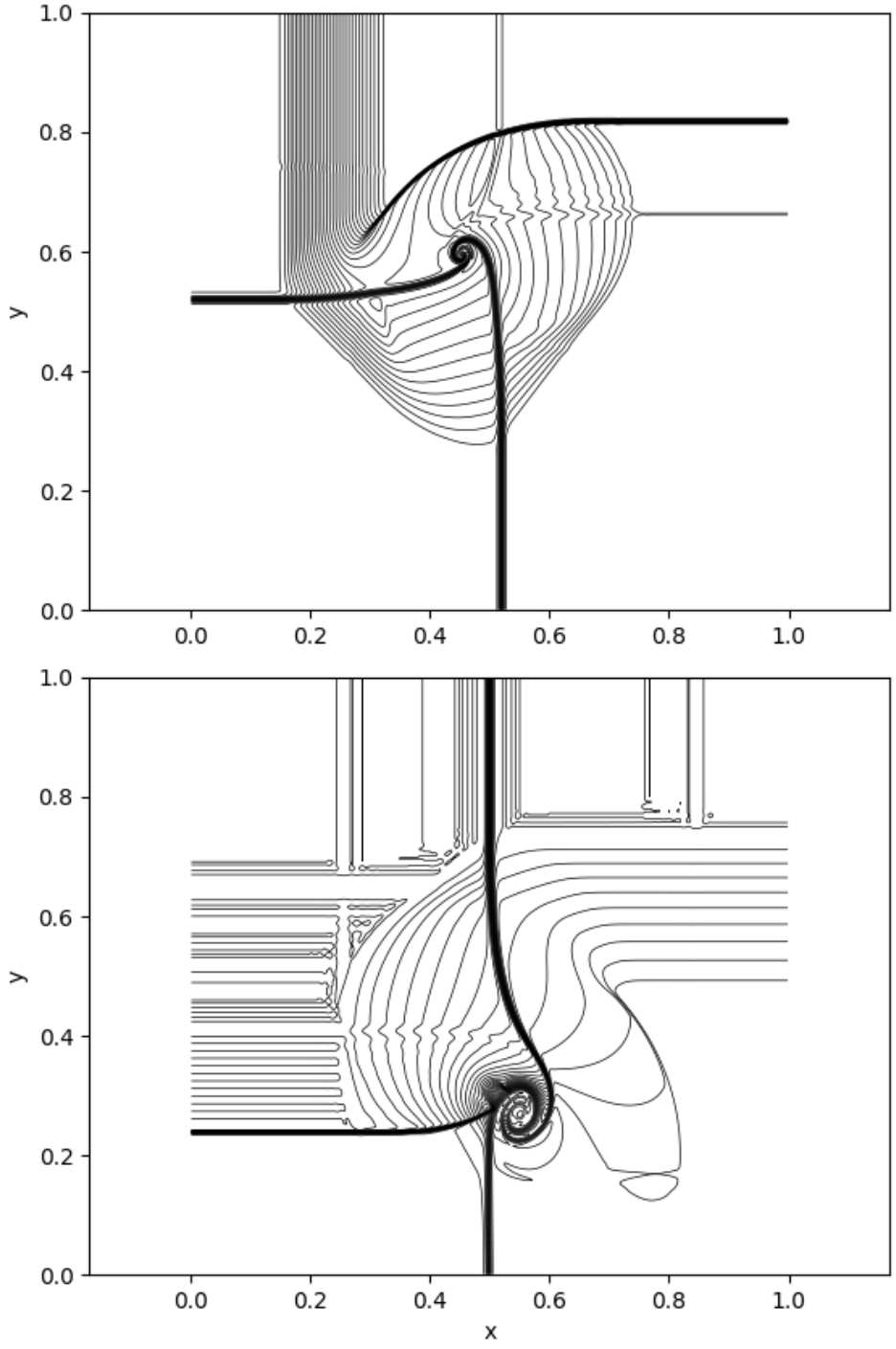

Figure 3.9: Density contours of the CCRWENO solutions of Configurations 16 (top) and 17 (bottom) from [38].

3.2.3 Rayleigh-Taylor Instability

This test case is taken from [41]. Gravity acts in the $+y$ direction on two fluids of densities 1 and 2 that are initially at rest but for a small perturbation in the vertical velocity. Because the Euler equations include no viscous effects, the apparent consequences of viscosity reflect solely the numerical viscosity. The domain is $0 < x < 0.25, 0 < y < 1$ and the initial condition is:

$$[\rho, u, v, P] = \begin{cases} [2, 0, -0.025\sqrt{\gamma P/\rho}\cos(8\pi x), 1 + 2y] & y \leq 0.5 \\ [1, 0, -0.025\sqrt{\gamma P/\rho}\cos(8\pi x), 1.5 + y] & y > 0.5 \end{cases} \tag{3.9}$$

where $\gamma = 5/3$ (the Atwood number is $1/3$). The simulation ends at time $t = 1.95$. The left and right boundaries are slip walls while solution values at the top and bottom boundaries are kept constant. Because the solution behavior depends strongly on the numerical viscosity, different formulations of the non-oscillatory weights can give markedly different results. Figure 3.10 shows the solutions obtained on a 64×256 grid and a 128×512 grid using the Jiang-Shu weights [15] and the solutions obtained on the same grids using the weighting strategy described in [42], which are defined as follows:

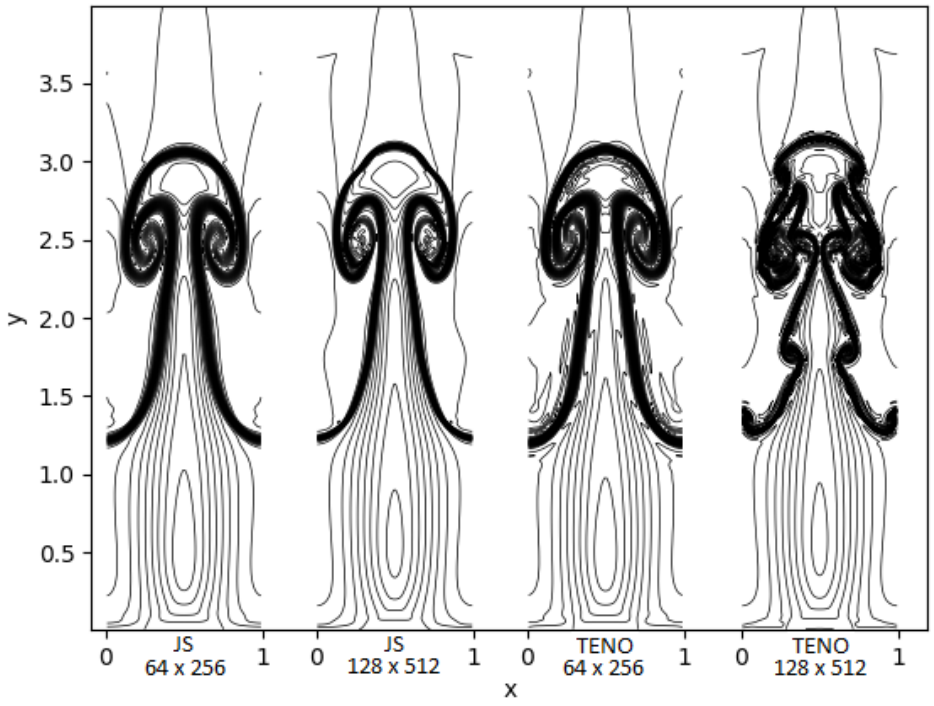

Figure 3.10: Density contours of the CCRWENO solutions of the inviscid Rayleigh-Taylor instability problem using Jiang-Shu and TENO weights.

$$
\begin{aligned}
\omega_s &= \frac{\bar{\omega}_s \delta_s}{\sum_r \bar{\omega}_r \delta_r} \\
\delta_s &= \begin{cases} 0 & \chi_s < C_T \\ 1 & \text{otherwise} \end{cases} \\
\chi_s &= \frac{\gamma_s}{\sum_r \gamma_r} \\
\gamma_s &= \left(C + \frac{\tau}{\epsilon + \beta_s} \right)^q
\end{aligned}
\tag{3.10}
$$

where β_s are the Jiang-Shu smoothness indicators [15]. The parameters are set as $q = 6$, $C_T = 10^{-4}$, and $C = 1$.

We see that on the coarser grid the choice of weights has little effect on the

solution although some nascent small features appear when the TENO weights are used. On the finer grid, however, the TENO weights clearly incur much less artificial dissipation giving rise to secondary vortices and a completely different plume shape. Compared to the cases using TENO weights in [42], which exhibit more fine features and asymmetric solutions, we conclude that even with TENO weights the CCRWENO method has more artifical dissipation than the upwind schemes of the same order used in [42].

3.2.4 Double Mach Reflection

Woodward and Colella [31] originated the double Mach reflection problem as a test case for non-oscillatory methods. A Mach 10 shock initially makes a 60° angle with the horizontal at $x = 1/6$ and moves rightward, forming two Mach stems and two contact discontinuities. The physical domain is $0 < x < 4, 0 < y < 1$ and the initial condition is:

$$[\rho, u, v, P] = \begin{cases} [1.4, 0, 0, 1] & y < (x - 1/6)\sqrt{3} \\ [8, 7.145, -4.125, 116.8333] & \text{otherwise} \end{cases} \tag{3.11}$$

The simulation ends at time $t = 0.2$. The exact post-shock conditions are imposed at the left boundary and along the portion of the lower boundary from $x = 0$ to $x = 1/6$, while the remainder of the lower boundary is a reflecting wall. Boundary values on the top edge correspond to the exact motion of the shock and the right boundary is an outflow boundary. This case also benefits from use of the TENO weighting strategy Eq. (3.10). Figure 3.11 shows the results on 512×128 and

1024×256 grids using the Jiang-Shu weights [15] and the TENO [42] weights using Eq. (3.10). The TENO weights cause less numerical dissipation thereby revealing the instability in the contact discontinuity as it approaches the wall.

3.2.5 High-Frequency Wave Propagation

We expect the compact reconstructions used in CCRWENO to improve the resolution of small-scale features. As a test we consider advection of sinusoidal density waves of increasingly high frequency on a fixed 150×150 grid. The initial condition is:

$$\begin{aligned} \rho(x,y) &= 2 + \sin(2\pi k x)\sin(2\pi k y) \\ u &= 1 \\ v &= 0 \\ P &= 1 \end{aligned} \tag{3.12}$$

which evolves according to the Euler equations Eq. (1.10). The domain is a square of side length 1 and the mesh ratio is $\lambda = \Delta t/\Delta x = 0.35$. Figure 3.12 shows the errors incurred for each wavenumber k with CCRWENO, CWENO4, and CWENO5 [29]. Sudden dips in the error occur at wavenumbers of $k = 30, 50,$ and 60 with each method, corresponding to cases where either the five-point stencil of the combined scheme or a three-point substencil contain an integer number of wavelengths. In such cases the combined point-value reconstruction becomes exact for a sinusoidal solution so less error accumulates per time step. CCRWENO either matches or considerably outperforms the other two schemes at each wavenumber.

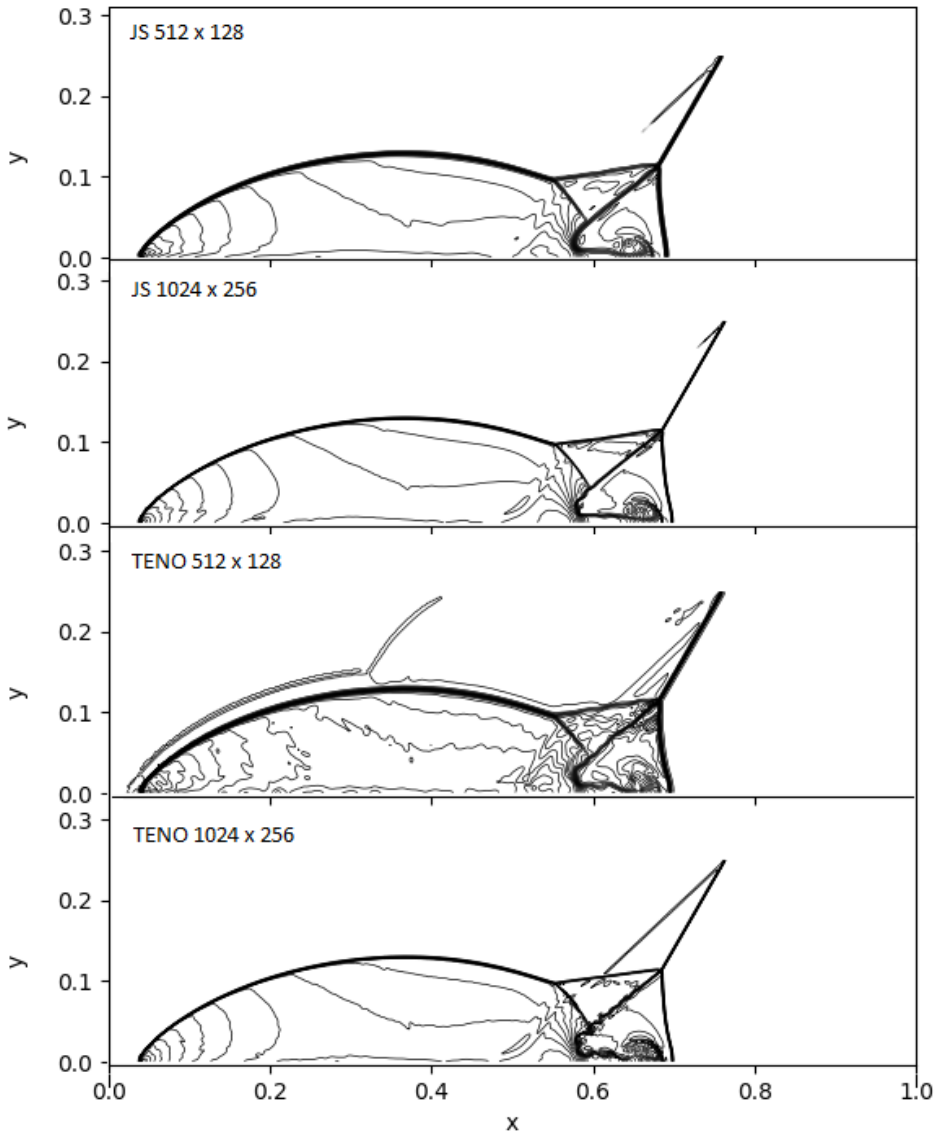

Figure 3.11: Density contours of the CCRWENO solutions of the double Mach reflection problem using Jiang-Shu and TENO weights.

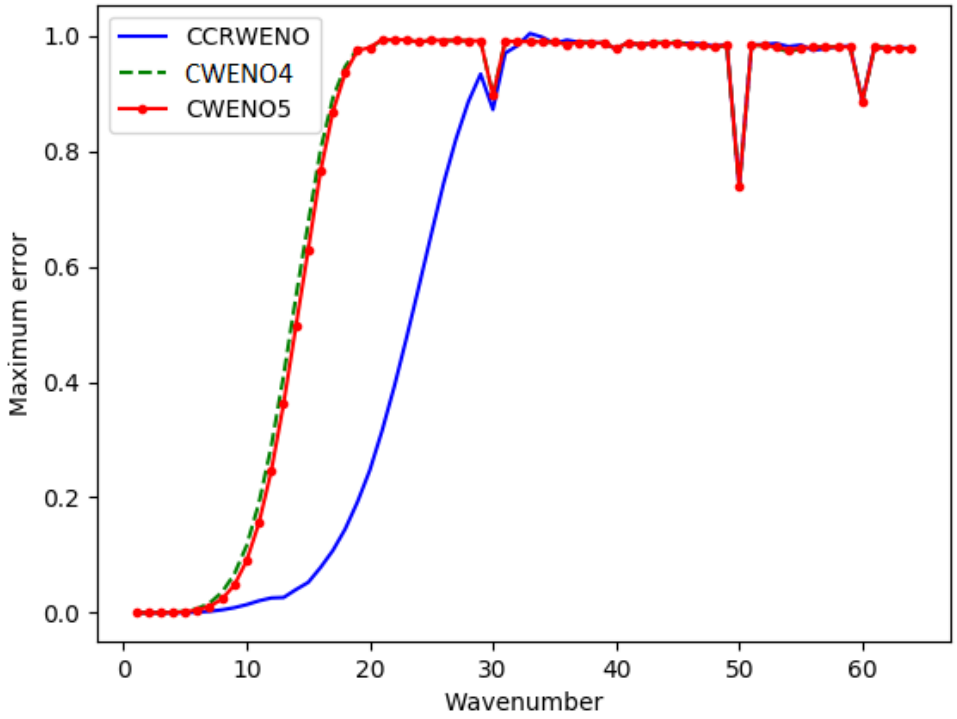

Figure 3.12: Maximum error vs. wavenumber for three central schemes. Solid line: error incurred with CCRWENO. Dashed line: CWENO4. Dotted line: CWENO5.

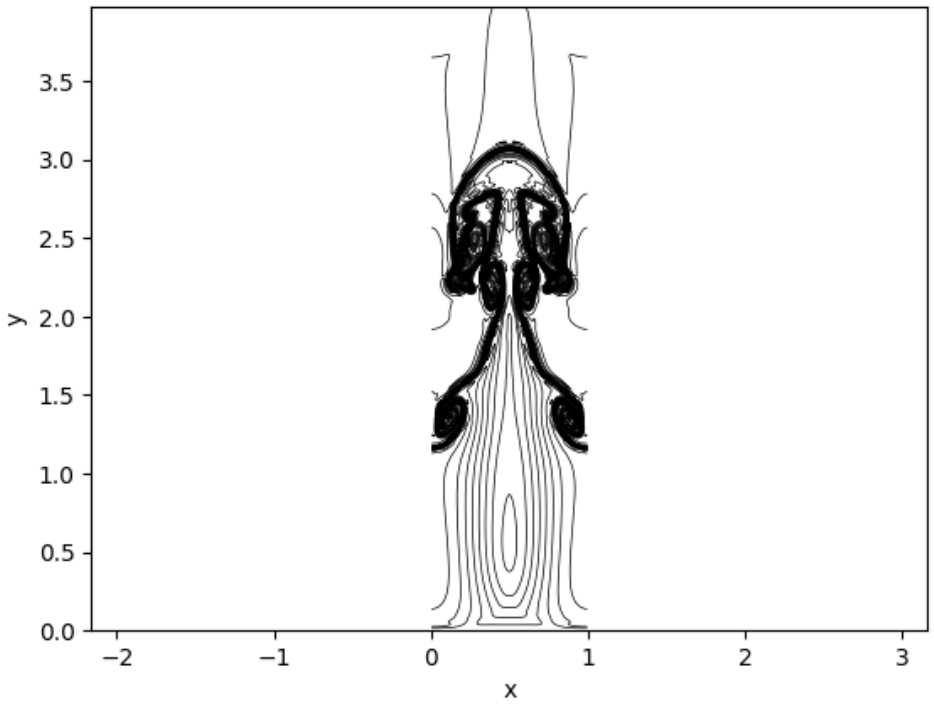

Figure 3.13: Density contours of the dual-grid CCRWENO solution to the Rayleigh-Taylor instability problem with TENO weights.

3.2.6 Dual-Grid Formulation

By construction, we expect the dual-grid CCRWENO to exhibit less numerical dissipation at small time steps. To test this expectation we apply the dual-grid variant to test cases where the time step is small: the Rayleigh-Taylor instability and, for a representative case among the two-dimensional Riemann problems, Configuration 3. Figure 3.13 shows the solution obtained for the Rayleigh-Taylor instability using the TENO weights with the same parameters and grid size as the rightmost result in Figure 3.10. Smaller numerical dissipation would give rise to finer features. Neither solution contains features that are clearly finer than those of the other.

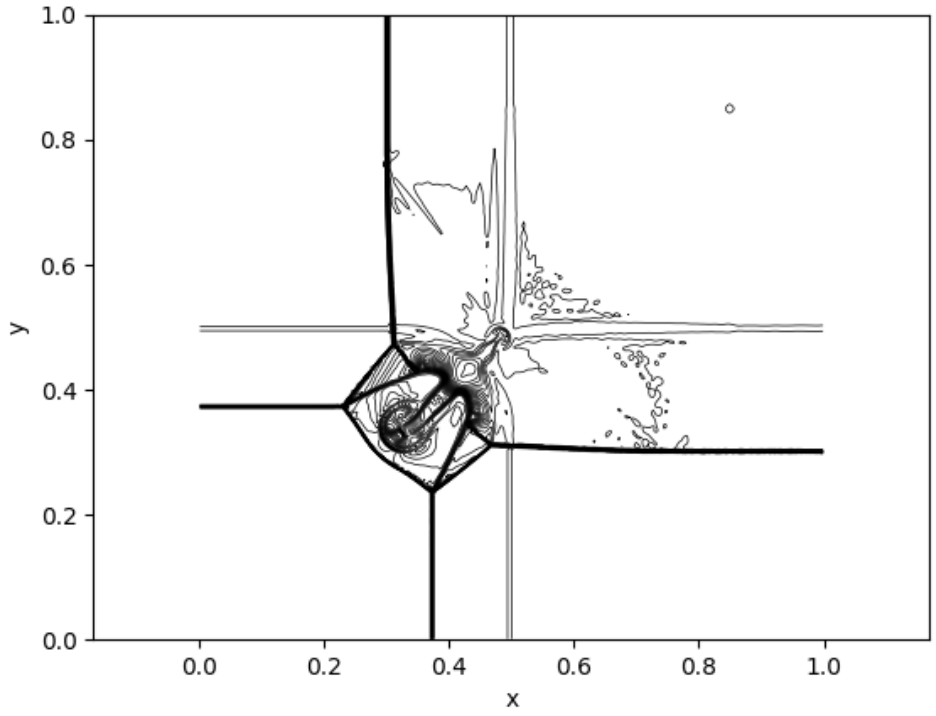

Figure 3.14: Density contours of the dual-grid CCRWENO solution to Configuration 3.

Figure 3.14 shows the solution to Configuration 3. Compared to Figure 3.7, the dual-grid variant produces more erratic oscillations and shows asymmetry in the central plume. The asymmetry results from the smaller numerical diffusion providing less stabilizing influence. These results clearly indicate that the dual-grid formulation has similar or less numerical dissipation when the time step is small. The cost of this improvement is drastically slower computation. Since a reconstruction is performed on each grid each Runge-Kutta stage, using the third-order SSP Runge-Kutta method [22] the computation time is increased by a factor of almost 6 over the original CCRWENO. Therefore the dual-grid variant should only be used when a small time step is absolutely necessary.

Chapter 4: Investigating Oscillations

The numerical results of the CCRWENO method applied to two-dimensional Riemann problems include spurious oscillations in some cases, indicating that the need for characteristic variables still remains. The original conjecture that using the same weights for both the subcell and point value reconstructions evidently is false. To uncover the reason for the oscillations we begin by reexamining the reasoning on which the original conjecture was based.

4.1 Characteristic Variables in CWENO4

Nowhere in the formulation of a central scheme does a need for characteristic variables appear. The two-dimensional CWENO4 scheme [30] produces solutions to Configurations 5 and 16 with no spurious oscillations. When that scheme is applied to other two-dimensional Riemann problems oscillations do appear. Figure 4.1 shows the results obtained with CWENO4 and the tensor-product extension of CWENO5 from [29] applied to Configuration 17. Qualitatively, these solutions hardly differ from the CCRWENO result in Figure 3.9, though the CCRWENO result has fewer oscillations. Comparing the CCRWENO results on Configurations 5 and 16 to the CWENO4 results on those cases presented in [30] show similar results, suggesting

that the need for characteristic variables depends in some way on the problem. It is not the case, however, that CWENO4 never requires characteristic variables as was first believed.

4.2 The Role of Characteristic Variables

It is now clear that inconsistency in the ideal weights between the subcell and point-value reconstructions does not necessitate characteristic variables as was first thought. To modify the CCRWENO method to avoid characteristic variables now requires a fuller understanding of their function, so that it may be fulfilled by some other means. The literature presents several explanations as to why characteristic variables are needed. Quoting directly some representative examples in the context of central schemes:

- "Due to the staggering, the approximation of the evolved fluxes is done in smooth regions (up to an appropriate CFL condition). Hence, no characteristic decomposition is required and the upwinding is replaced by a straightforward centered computation of the quantities involved." [28]

- "Namely, no Riemann problems are solved and consequently characteristic decompositions – required in order to distinguish between the left and right-going waves inside the Riemann fan, are avoided."' [10]

- "When the reconstruction order becomes higher, characteristic decomposition is usually necessary to reduce spurious oscillations for systems of conservation

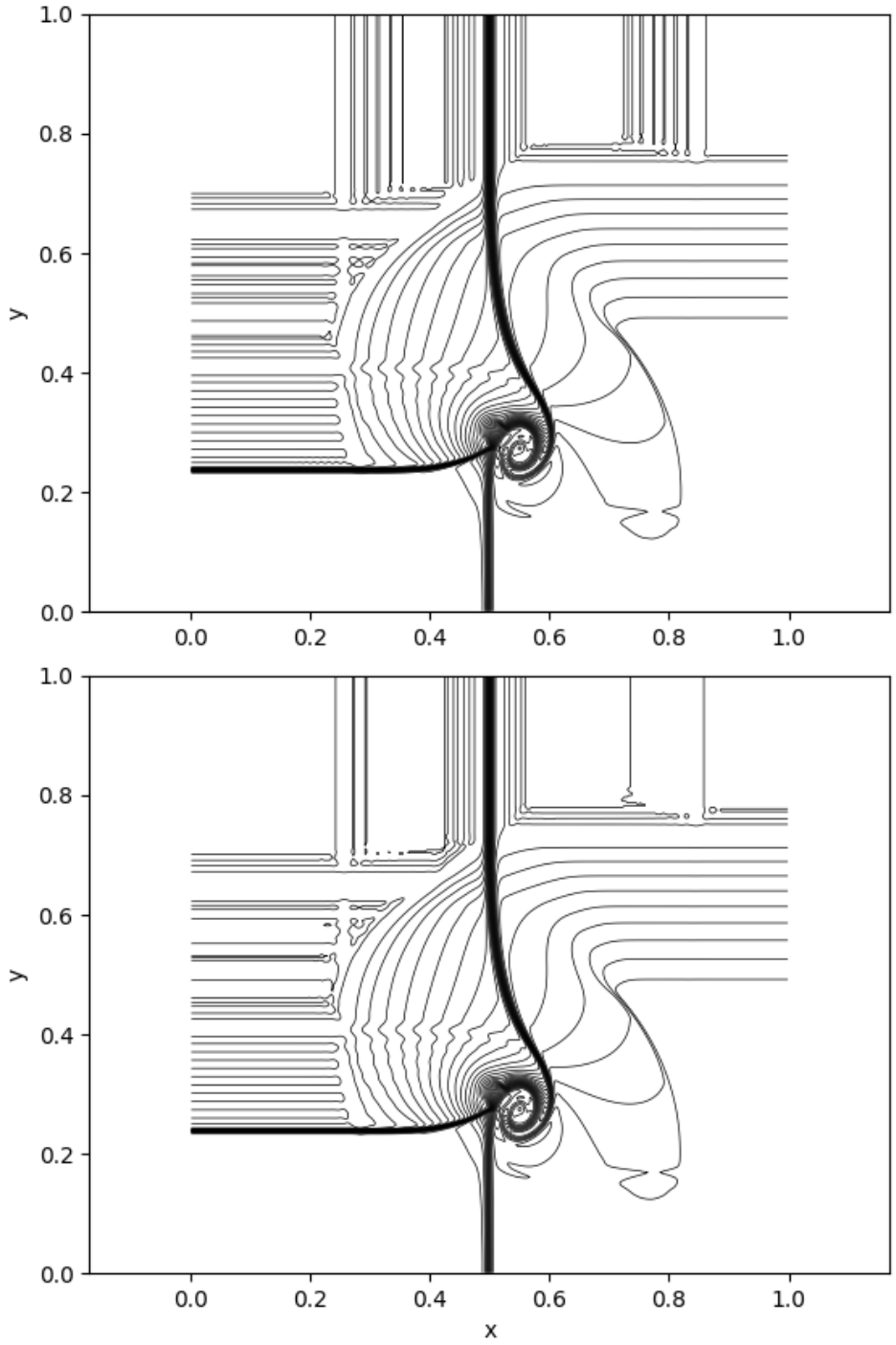

Figure 4.1: The CWENO4 and CWENO5 schemes produce spurious oscillations on Configuration 17.

laws. Characteristic decomposition locally creates larger smooth area for polynomial reconstruction by separating discontinuities into different characteristic fields."' [43]

And in the context of upwind schemes:

- "However, for more demanding test problems, or when the order of accuracy is high, we would need the following more costly, but much more robust characteristic decompositions."' [44]

- "It is also known that while the characteristic-based reconstruction is necessary for inviscid flow problems with strong discontinuities, reconstruction of conserved or primitive variables suffices when physical viscosity is present."' [18]

None of these explanations survive close scrutiny. Regarding the central schemes, the explanations from [28] or [10] would imply that characteristic variables would never be necessary for any central scheme since every central scheme evaluates fluxes in smooth regions and avoids solving Riemann problems. Yet in Chapter 3 and Figure 4.1 we clearly see oscillations in some cases. The assertion in [43] that high order of accuracy usually requires characteristic variables is somewhat undermined by that paper itself, which presents a process (the hierarchical reconstruction) that eliminates the oscillations but preserves the order of accuracy without requiring characteristic variables (though often producing over- and undershoots at discontinuities [43]). Evidently the high-order is not inherently at fault, since the fifth-order accuracy preserved by the hierarchical reconstruction does not

lead to oscillations, though it is an experimental fact that without special treatment high-order schemes without characteristic decompositions produce oscillations.

The second part of the explanation from [43] correctly states that after the characteristic decomposition it may be that some characteristic fields are smooth, but the relevance of this fact is dubious. First, it can happen that a discontinuity in one conserved variable causes discontinuities in more than one characteristic variable, which if this explanation were true would suggest that under some circumstances the characteristic decomposition can actually exacerbate oscillations. This appears never to occur and a literature search for documentation of such a phenomenon has been fruitless. Second, if redistributing the discontinuity among characteristic fields explains the need for the decomposition, then one would expect the scalar case to be the most difficult since no such redistribution is possible. In reality, however, the scalar case is the most well-behaved. Finally, this explanation has no connection to the order of accuracy and we know experimentally that the order of accuracy is important.

Another common explanation, found in [44] for example, is that the characteristic decomposition decouples the solution into independent scalar equations. Disregarding the fact that this is true only in an approximate sense when the Jacobian is non-constant, this explanation is only a reduction to the scalar case. One still needs to explain why the scalar case is so well-behaved, and why the order of accuracy has any importance. It could be true that even an approximate diagonalization is sufficient.

The explanations from the literature on upwind schemes make more sense though they are less directly applicable to mitigating the oscillations from CCR-WENO. The explanation from [44] is consistent with experiments but is hardly a precise mathematical statement. What makes a problem demanding? Similarly the explanation in [18] accurately reflects practical knowledge but does not lead directly to a mathematical formulation. Intuitively, one expects a shock to be "strong"' if its magnitude is large. In the scalar case, however, even large jumps do not lead to oscillations.

To design a high-order method that avoids the characteristic decomposition we need to properly understand its function, and this understanding must be consistent with the following experimental facts:

1. Oscillations do not appear in the scalar case.

2. Even without characteristic variables, some discontinuities produce small or no oscillations.

3. Central schemes do not produce oscillations when the flux Jacobian is constant, even without the characteristic decomposition.

4. Characteristic variables are needed when the order of accuracy is 4 or greater (and possibly for lower-order schemes).

5. Oscillations only appear near discontinuities.

Assuming to be true the explanation in [44], that the characteristic decomposition (approximately) diagonalizes the system Eq. (1.1), we can explain some

of the salient facts. The scalar case behaves well because the Jacobian is already diagonal, being simply a scalar. The discontinuities that produce small or no oscillations might be those for which the Jacobians (of the flux in the direction normal to the discontinuity) of the left and right states are already close to diagonal. This prediction is confirmed by Figure 4.2, in which oscillations emanate from the upper discontinuity while the lower discontinuity produces none. We see that the Jacobians for the left and right states in the top half have zeros on their diagonals whereas those in the bottom half, though not diagonally dominant, are closer to being so than their upper counterparts. On the other hand, Figure 4.3 shows that in Configuration 4 no oscillations are visible (though some appear at intermediate time steps before being obscured by the contour level choice), yet the Jacobians are not at all close to being diagonally dominant.

On the other hand, the facts are also explained by the following hypothesis: the characteristic decomposition produces a system for which the eigenvectors of the flux Jacobians are approximately constant over a stencil. In particular, the eigenvectors on either side of a discontinuity become closer to equal. Therefore the characteristic decomposition is needed when those eigenvectors in the original system are too different. This explanation addresses all the shortcomings of those previously discussed:

- The scalar case is well-behaved because the eigenvectors are simply 1, which is already constant.

- The discontinuities that produce small or no oscillations are those for which

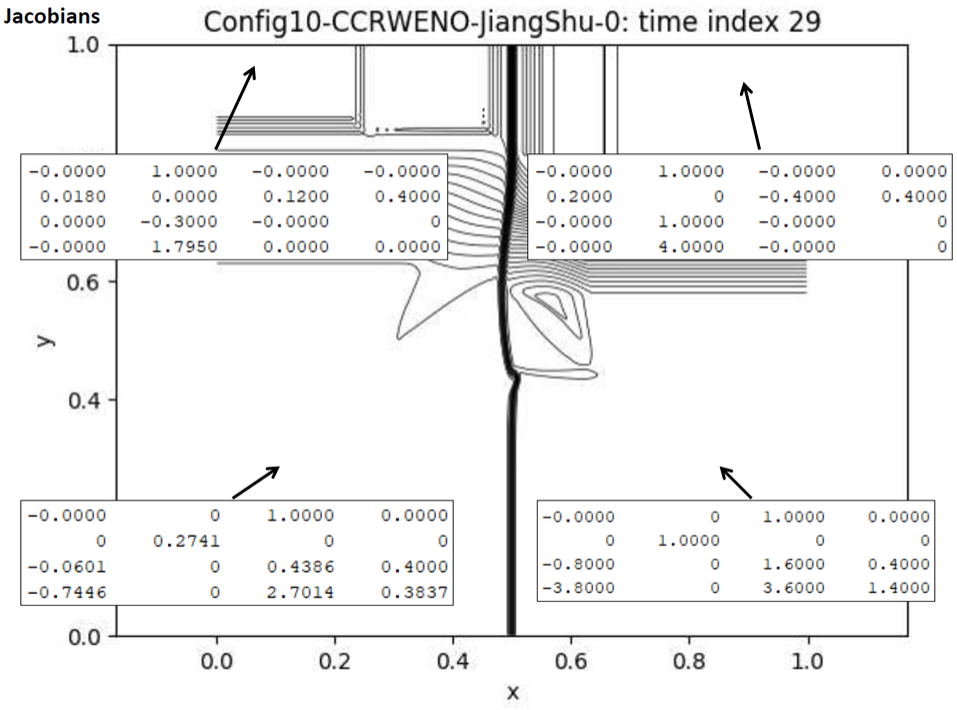

Figure 4.2: The Jacobians for the lower discontinuity are closer to diagonally dominant than are the Jacobians for the upper discontinuity.

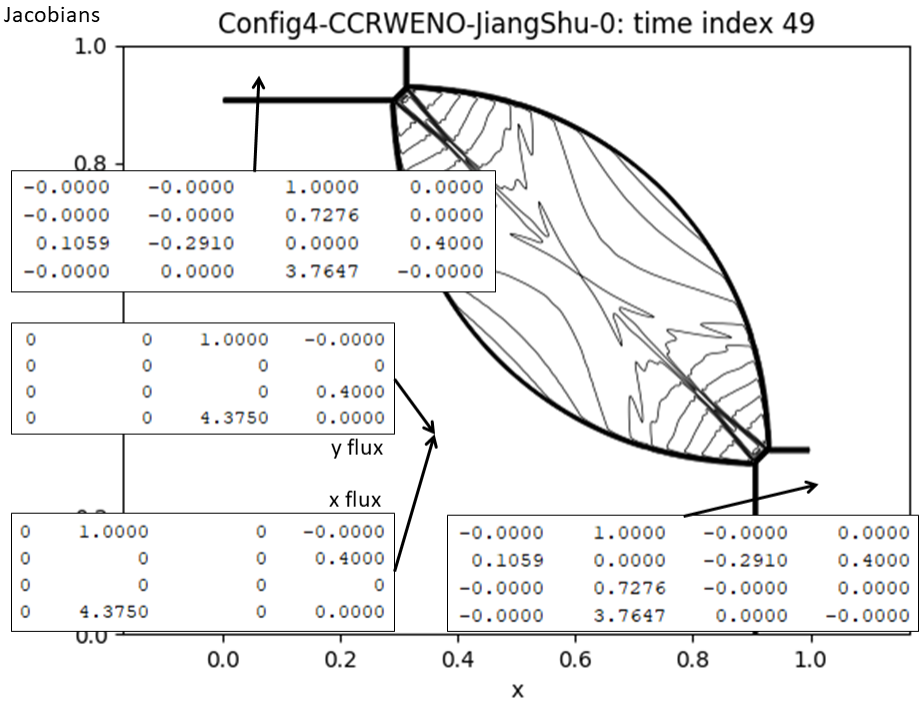

Figure 4.3: In Configuration 4 the oscillations are minuscule despite non-diagonally-dominant Jacobians.

the eigenvectors are already similar.

- When the flux Jacobian is constant so are the eigenvectors. The central framework does not involve flux Jacobian eigenvectors at all, so it is not surprising that problems whose origin lies with those eigenvectors would affect central schemes. Central schemes do not require upwinding, but the eigenvectors may still change.

- High-order schemes require characteristic variables while low-order schemes do not because high-order schemes better preserve large gradients near the shock location. These large gradients in the solution lead to dissimilar eigenvectors within a stencil necessitating a change to characteristic variables. In contrast, low-order schemes diffuse discontinuities rapidly enough that these gradients are smaller, and the low-order schemes use smaller stencils anyway. In viscous flows the physical viscosity performs the same function, leading to the observed lesser need for characteristic variables [18].

- Oscillations appear near discontinuities because the eigenvector matrix is discontinuous there.

- A physics-independent measure of shock strength is the dissimilarity between the eigenvectors on either side.

Furthermore, when the Jacobian is diagonalized the eigenvector matrix of the diagonalized system is the identity which is constant, explaining why diagonalization appears to be important. And finally, recall that the justification for the charac-

teristic decomposition involved treating the eigenvector matrix as constant within a stencil anyway.

For all its elegance, the eigenvector hypothesis has conceptual obstacles. First, if an eigenvalue is repeated (as occurs in the multidimensional Euler equations) then its eigenvectors are not unique. In this case the comparison should be the angle between the eigenspaces corresponding to the repeated eigenvalues [45]. This comparison causes no trouble with the Euler equations which always have a repeated eigenvalue, but for a general conservation law it may be that an eigenvalue is repeated on one side of a shock but on the other side the eigenvalues are all distinct. A natural comparison is not clear in such a case. This possibility raises the broader question of which eigenvectors should be compared to each other in the first place. For the Euler equations where it is known that the eigenvalue set of each flux Jacobian has the form $\{u-a, u, u, u+a\}$ the comparison can be between corresponding eigenvectors after the eigenvalues are sorted in ascending order (provided that u does not change sign). This might not be possible for general fluxes.

After accounting for these details, the agreement between eigenvectors for the states in Configuration 10 of [38], shown in Figure 4.4, is fairly close but not dramatically so. The eigenvectors for the left and right states of the lower shock are somewhat similar whereas those for the upper shock are much less so simply by noting the signs of individual components. In the well-behaved Configuration 4, shown in Figure 4.5 with the relevant eigenvectors, the eigenvectors are mostly close to equal (after possibly changing signs of columns). The eigenvector hypothesis led to a more correct prediction about this configuration than did the diagonalization

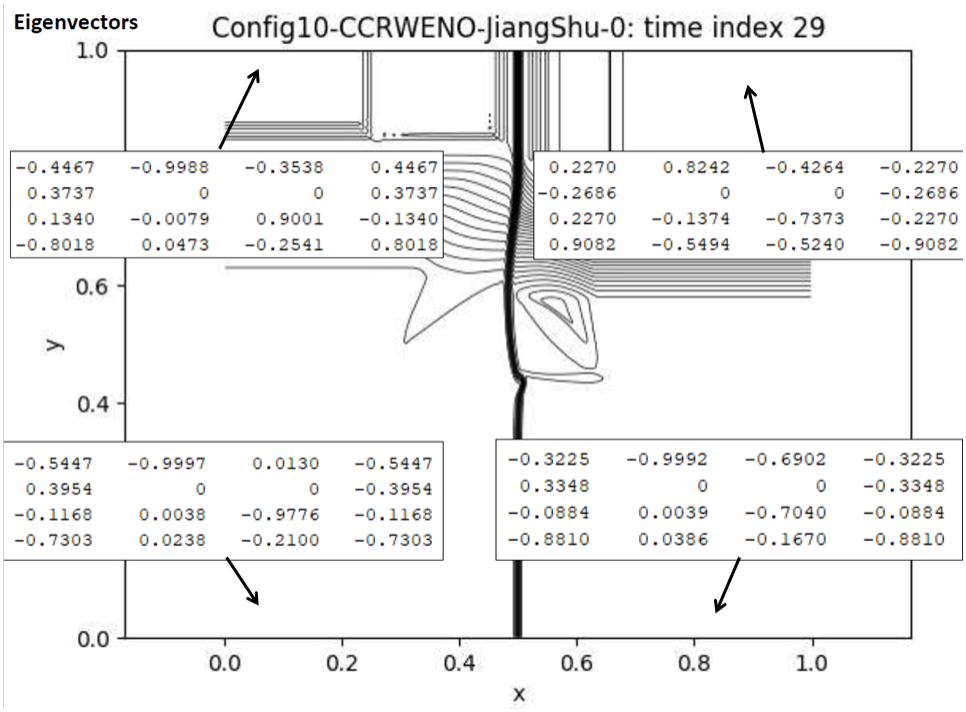

Figure 4.4: The eigenvector matrices for the lower discontinuity are closer to equal than are those for the upper discontinuity.

hypothesis.

The eigenvectors shown in Figures 4.4 and 4.5 demonstrate another troublesome feature of the eigenvector hypothesis: the approximate equality can be rough. Considering the lower discontinuity in Figure 4.4, for example, the first columns of the two eigenvector matrices have the same sign pattern but the individual components differ from their respective averages by as much as 25%. Calling them approximately equal is justifiable only in comparison to the eigenvectors for the upper discontinuity which differ even in signs. The agreement in Figure 4.5 is much closer but the fourth columns each have a component that clearly does not approximate its counterpart. However, the fact that discrepancies of this size observably

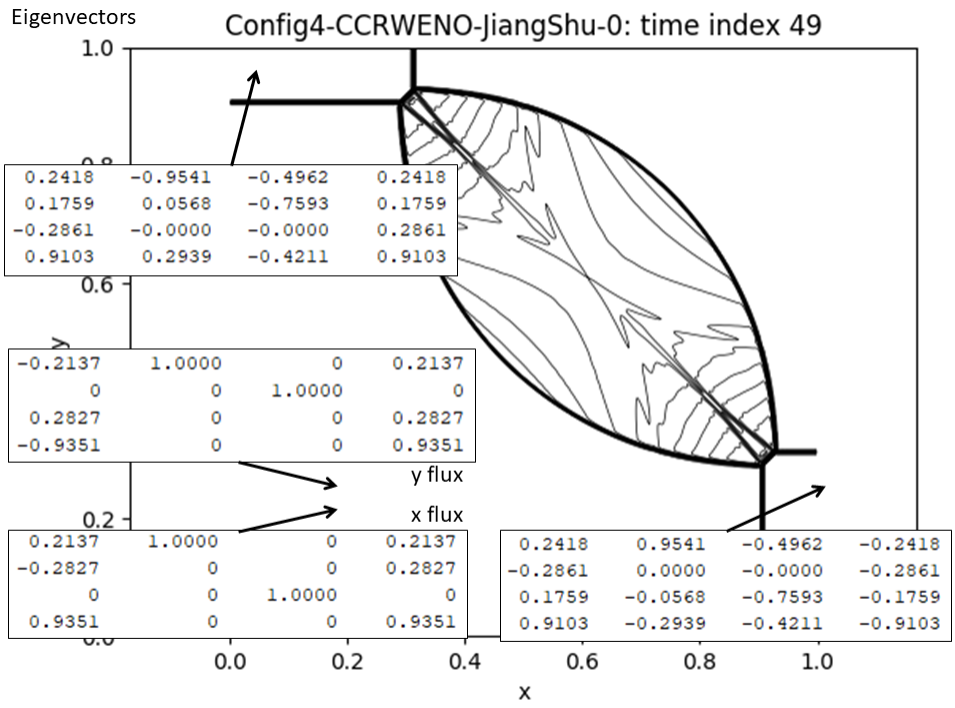

Figure 4.5: The eigenvector matrices for two of the discontinuities are numerically close.

do not necessitate a characteristic decomposition demonstrates that such a transformation need not align the eigenvectors perfectly (as one would expect since it is known to work despite not being fully rigorous). It may even be enough to simply match the signs of the components.

4.2.1 Numerical Characteristic Transformations

We need to check now whether applying a characteristic transformation results in Jacobians that are more diagonally dominant and/or have more similar eigenvectors in the case of the two-dimensional Euler equations. The transformation matrix will be the inverse of the eigenvector matrix for the Jacobian of the flux in the appropriate direction, corresponding to eigenvalues sorted in increasing order, evaluated at the Roe-averaged state [7] computed from two state vectors (i.e., the left and right states across a discontinuity). The Roe average q_A is obtained from the

left state q_L and right state q_R by:

$$q_A = \begin{bmatrix} \rho_A \\ \rho_A u_A \\ \rho_A V \\ E_A \end{bmatrix}, r = \frac{\sqrt{\rho_R}}{\sqrt{\rho_L}}$$

$$\begin{aligned} \rho_A &= r\rho_L \\ u_A &= \frac{u_L + r u_R}{1+r} \\ V &= \frac{v_L + r v_R}{1+r} \\ H_A &= \frac{H_L + r H_R}{1+r} \\ E_A &= \frac{1}{\gamma}\left(\rho_A H_A + \frac{\gamma - 1}{2}\rho_A(u_A^2 + v_A^2)\right) \end{aligned} \tag{4.1}$$

where H is the enthalpy, related to the energy E by:

$$H = \frac{1}{\rho}\left(\gamma E - \frac{\gamma - 1}{2}\rho(u^2 + v^2)\right) \tag{4.2}$$

Testing the eigenvector hypothesis with the two-dimensional Euler equations is complicated by the repeated eigenvalue leading to a two-dimensional eigenspace. Therefore the numerical tests will examine eigenvectors with the one-dimensional Euler equations and the Jacobian diagonals for the two-dimensional Euler equations. To quantify the improvement in diagonal dominance, we introduce a diagonal dominance measure of an $N \times N$ matrix A defined by:

$$\theta(A) = \frac{1}{N}\sum_i \frac{|A_{ii}|}{\sum_j |A_{ij}|} \tag{4.3}$$

where if the denominator in the sum is zero then the whole summand is set to zero. $\theta(A) = 0$ if and only if A has zeros on the diagonal and $\theta(A) = 1$ if and only if A is

diagonal.

We now examine the diagonals of Jacobians before and after a characteristic transformation, taking examples from discontinuities in the two-dimensional Riemann problems of [38].

1. The discontinuity between the first and second quadrants in Configuration 11.

$$q_L = \begin{bmatrix} 2 \\ 0 \\ -0.6 \\ 2.59 \end{bmatrix}, q_R = \begin{bmatrix} 1 \\ 0 \\ 0.3 \\ 2.545 \end{bmatrix}, q_A = \begin{bmatrix} 1.4142 \\ 0 \\ -0.0728 \\ 2.5460 \end{bmatrix} \tag{4.4}$$

The Jacobian at the states q_L, q_R, and q_A are:

$$J_L = \begin{bmatrix} 0 & 1.0000 & 0 & 0 \\ 0.0180 & 0 & 0.1200 & 0.4 \\ 0 & -0.3000 & 0 & 0 \\ 0 & 1.7950 & 0 & 0 \end{bmatrix}, J_R = \begin{bmatrix} 0 & 1.0000 & 0 & 0 \\ 0.0180 & 0 & -0.1200 & 0.4 \\ 0 & 0.3000 & 0 & 0 \\ 0 & 3.5450 & 0 & 0 \end{bmatrix}$$

$$J_A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0.0005 & 0 & 0.0206 & 0.4 \\ 0 & -0.0515 & 0 & 0 \\ 0 & 2.5199 & 0 & 0 \end{bmatrix} \tag{4.5}$$

The eigendecomposition of J_A is:

$$J_A = V\Lambda_A V^{-1}$$

$$V = \begin{bmatrix} -0.3459 & 1 & 0.9407 & 0.3459 \\ 0.3471 & 0 & 0 & 0.3471 \\ 0.0178 & 0 & 0.3386 & -0.0178 \\ -0.8715 & -0.0013 & -0.0187 & 0.8715 \end{bmatrix}, \Lambda_A = \begin{bmatrix} -1.0037 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.0037 \end{bmatrix} \tag{4.6}$$

Note that the J_L and J_R have zeros on their diagonals. The characteristic decomposition procedure transforms the quasilinear form

$$\frac{dq}{dt} + J(q)\frac{dq}{dx} = 0 \tag{4.7}$$

by premultiplying it with V^{-1}, which is equivalent to the following quasilinear system:

$$\frac{d}{dt}(V^{-1}q) + V^{-1}J(q)V\frac{d}{dx}(V^{-1}q) = 0 \tag{4.8}$$

We are therefore interested in the diagonals of $V^{-1}J_LV$ and $V^{-1}J_RV$ in comparison to those of J_L and J_R. We have:

Table 4.1: Diagonal dominance of Jacobians with and without characteristic decomposition for the upper discontinuity in Configuration 11.

θ	J	$V^{-1}JV$
Left	0	0.3875
Right	0	0.4108

2. The discontinuity between quadrants 2 and 3 in Configuration 18.

Table 4.2: Diagonal dominance of Jacobians with and without characteristic decomposition for the left discontinuity in Configuration 18.

θ	J	$V^{-1}JV$
Left	0.3739	0.7265
Right	0.1921	0.6581

Table 4.3: Diagonal dominance of Jacobians with and without characteristic decomposition for the lower discontinuity in Configuration 16.

θ	J	$V^{-1}JV$
Left	0.2859	0.5986
Right	0.3245	0.5353

3. The discontinuity between quadrants 3 and 4 in Configuration 16

It is clear enough that the characteristic decomposition serves to transform the original system into one whose Jacobian is more diagonally dominant, which is not surprising given the motivation. Liu and Osher in [46] made a similar observation: "...near discontinuities, the flux must be close to a flux which does not mix the fields" (which is closer to the eigenvector hypothesis). A full examination would consider the case where the transition from the left to the right state is not abrupt, corresponding to a smeared shock within a stencil. Because such an analysis would depend on the specific method being employed it is outside the scope of this chapter. However, we now have a plausible explanation for why the best choice of the average state can depend on the problem: in some cases, it might not provide enough diagonal dominance or even reduce it.

We now turn our attention to the one-dimensional Euler equations to examine

the eigenvectors. To quantify the similarity between two sets P and Q of eigenvectors, in the order of increasing eigenvalues and normalized to have L^2 norm 1, we introduce the quantity $\eta(P, Q)$:

$$\eta(P, Q) = \sum_i \max \|P_i - Q_i \operatorname{sign}(P_i \cdot Q_i)\|_{L^\infty} \tag{4.9}$$

The dot product accounts for eigenvectors being equivalent under sign change; it automatically replaces Q_i with $-Q_i$ if $-Q_i$ would make a smaller angle with P_i. A value of $\eta = 0$ indicates perfect agreement, and for consistency all eigenvectors are normalized to have length 1. We consider three cases:

1. A steady shock with left and right states:

$$q_L = \begin{bmatrix} 1 \\ 1 \\ 3.7857 \end{bmatrix}, q_R = \begin{bmatrix} 2.6667 \\ 2 \\ 8.7857 \end{bmatrix} \tag{4.10}$$

2. The initial discontinuity in the Lax shock tube problem

3. The initial discontinuity in the Shu-Osher problem

Table 4.4 shows the similarity measure for the eigenvectors before and after characterstic decomposition for these cases. In each case the eigenvectors of the transformed system are less similar than those of the original system. In view of this fact and the conceptual difficulties with comparing sets of eigenvectors we discard the eigenvector hypothesis in favor of the diagonal dominance hypothesis, with the caveat that diagonal non-dominance does not necessarily lead to oscillations if characteristic variables are not used.

Table 4.4: Similarity of Jacobian eigenvectors with and without characteristic decomposition for several test cases.

Case	J_L and J_R	$V^{-1}J_LV$ and $V^{-1}J_RV$
1D shock	0.1950	0.5968
Lax problem	0.9539	1.9995
Shu-Osher problem	1.3975	2.7462

4.3 Designing a Non-Oscillatory CCRWENO Method

4.3.1 Approximate Characteristic Decomposition

The characteristic decomposition procedure can be formulated as the solution to a maximization problem.

Definition 4.1 (Generalized Characteristic Decomposition - One-dimensional Case)**.** *Find a $D \times D$ matrix M satisfying:*

$$M = argmax \sum_j \theta(MJ(q_j)M^{-1}) \tag{4.11}$$

where the sum runs over all cells j in a stencil and $J(q_j)$ is the Jacobian of the physical flux function evaluated at q_j.

The characteristic decomposition is usually presented as a procedure, instead of a solution of a specific problem, and a procedure that only applies naturally in one dimension at that. Definition 4.1, however, extends naturally to multiple dimensions as follows:

Definition 4.2 (Generalized Characteristic Decomposition - Multidimensional Case)**.**

Find a $D \times D$ matrix M satisfying:

$$M = argmax \sum_{d=1}^{D} \sum_{\vec{j}} \theta(M J_d(q_{\vec{j}}) M^{-1}) \tag{4.12}$$

where J_d is the Jacobian of the physical flux in the dth direction and the interior sum runs over all cells in the multidimensional stencil, indexed by the multi-index $\vec{j}$ (see Table 2.4).

To keep the componentwise reconstructions in CCRWENO requires that the matrix M be diagonal in every stencil. However, such a matrix cannot improve the diagonal dominance of an arbitrary matrix.

Proposition 4.1. *Let $D \in \mathbb{R}^{N \times N}$ be a diagonal matrix. Then $\theta(DAD^{-1}) \geq \theta(A)$ for all $A \in \mathbb{R}^{N \times N}$ if and only if $D_{ii} = \pm c$ for some constant $c \in \mathbb{R}$, in which case $\theta(DAD^{-1}) = \theta(A)$.*

Proof. Let A_{ij} be the i, j entry of A and d_i be the i, i entry of D. Rewriting the inequality using the definition of θ:

$$\begin{aligned} \theta(DAD^{-1}) &= \frac{1}{N} \sum_i \frac{|d_i A_{ii} d_i^{-1}|}{\sum_j |d_i A_{ij} d_j^{-1}|} \\ &= \frac{1}{N} \sum_i \frac{|A_{ii}|}{|d_i| \sum_j |A_{ij}||d_j^{-1}|} = \theta(AD^{-1}) \\ &\geq \frac{1}{N} \sum_i \frac{|A_{ii}|}{\sum_j |A_{ij}|} \end{aligned} \tag{4.13}$$

which is true for arbitrary A_{ij} if and only if:

$$\begin{aligned} & |d_i| \sum_j |A_{ij}||d_j^{-1}| \leq \sum_j |A_{ij}| \ \forall i \\ \iff & |A_{ii}| + \sum_{j \neq i} |A_{ij}| \frac{|d_i|}{|d_j|} \leq |A_{ii}| + \sum_{j \neq i} |A_{ij}| \ \forall i \\ \iff & \left| \frac{d_i}{d_j} \right| \leq 1 \ \forall i, j \text{ with } j \neq i \end{aligned} \tag{4.14}$$

Of course, if the last inequality is strict for some i and j then interchanging i and j produces a violation. Therefore the only possibility is that $|d_i| = |d_j|$ for all i, j. Clearly this situation would make the original inequality become an equality. $\square$

Allowing non-diagonal M would mean sacrificing the advantages of componentwise reconstruction, so another strategy is needed.

4.3.2 Limiters

Rather than altering the innards of the algorithm we now consider modifying the solutions obtained with the CCRWENO method of Chapter 2. The oscillations form from initial overshoots or undershoots that are imperfectly corrected in later steps, so the most obvious strategy is to blend the CCRWENO solution with a solution that is closer to the original cell averages. This blending should be performed only near jumps since it is only there where oscillations are generated. Thus some form of discontinuity detection is necessary. Fortunately this is already accomplished by the calculation of the non-oscillatory weights; a jump exists wherever one of those weights is small. This reasoning leads to the following limiting procedure for the subcell averages $\bar{u}^S$ in terms of the CCRWENO solutions $\tilde{u}^S$:

$$\bar{u}_j^S = \begin{cases} (1-r)\tilde{u}_j^S + \frac{r}{2^D}\bar{u}_j & \min\omega < C_\omega \\ \tilde{u}_j^S & \text{otherwise} \end{cases} \tag{4.15}$$

Experimentation, shown in Figure 4.6 suggests that $C_\omega = 10^{-3}, r = 0.2$ removes the most oscillations without introducing excessive dissipation in a one-

dimensional test case (a vertical cross-section of the moving shock on the right of Configuration 16). The large undershoots that remain are generated by a different mechanism, since they occur even when an upwind scheme is employed with the characteristic decomposition. This subcell relaxation strategy extends straightforwardly to arbitrarily many dimensions, and Figures 4.7 and 4.8 show the results when it is applied to Configurations 3 and 17 respectively with the same parameters. The original CCRWENO solution of Configuration 3 resolved the intersecting shocks without difficulty, but because those shocks occur in such close proximity the subcell relaxation introduces dissipation into too many cells leading to a worthless solution. The solution to Configuration 17 is less degraded – the rolled-up contact discontinuity is at least visible – but the true discontinuities are more smeared and the oscillations not entirely eliminated. Indeed, along the lower edge of the domain oscillations were created or accentuated that were not visible before (cf. Figure 3.9). Varying the weight threshold C_ω and relaxation r produced similar results, and applying the limiting to the point value as well has virtually no effect. Clearly this simple strategy costs too much accuracy for its meager improvement on the oscillations.

An alternative is the hierarchical reconstruction originated by Liu et al. in [34] for central discontinuous Galerkin schemes and in [43] for traditional finite-volume methods. This process takes a polynomial representation of the solution within a cell and limits the derivatives in a recursive fashion, starting with the highest-order, in a way that removes oscillations but preserves the original order of accuracy. Of course, the obstacle in applying the hierarchical reconstruction to the CCRWENO

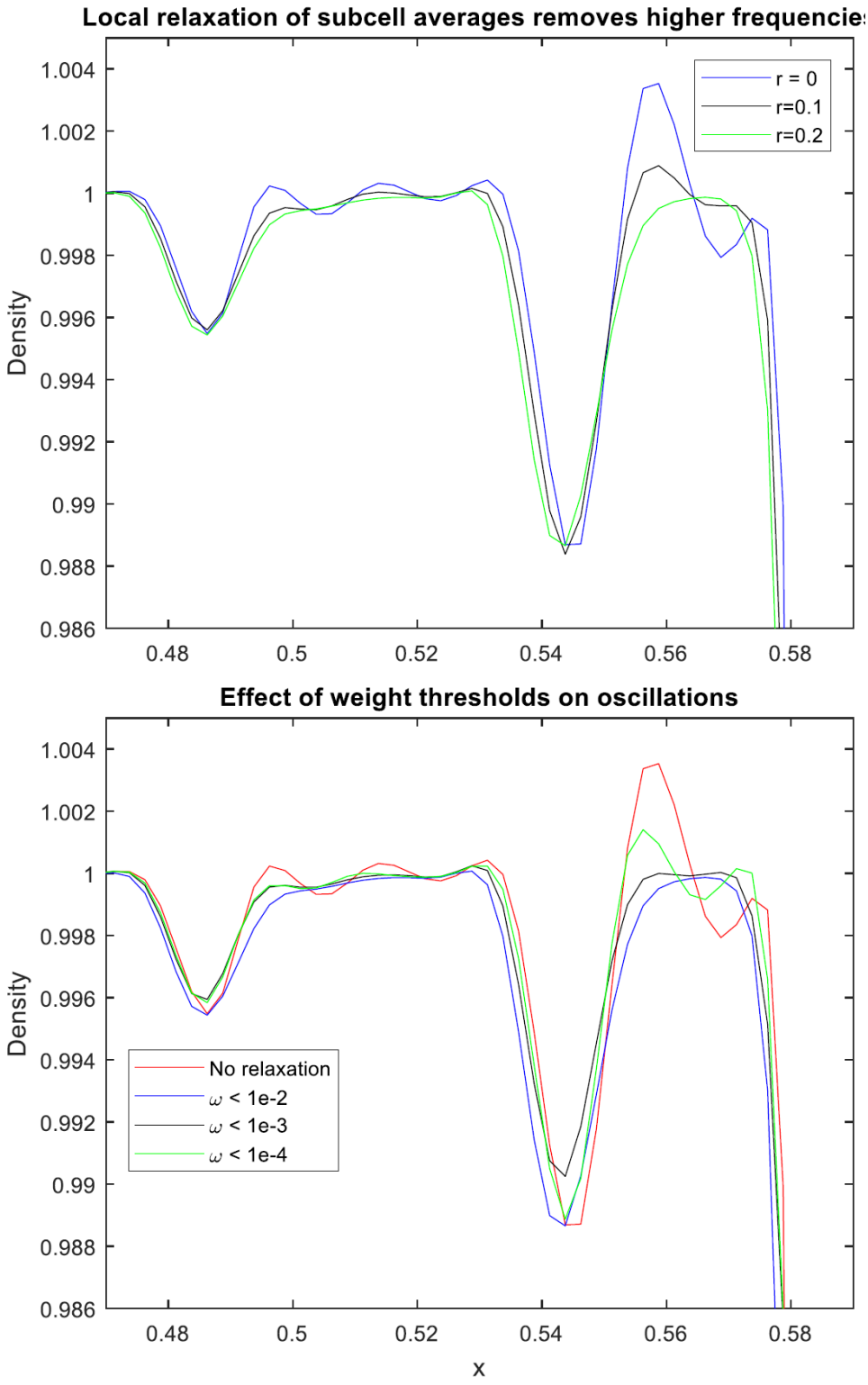

Figure 4.6: $C_\omega = 10^{-3}, r = 0.2$ give acceptable results for a simple test case.

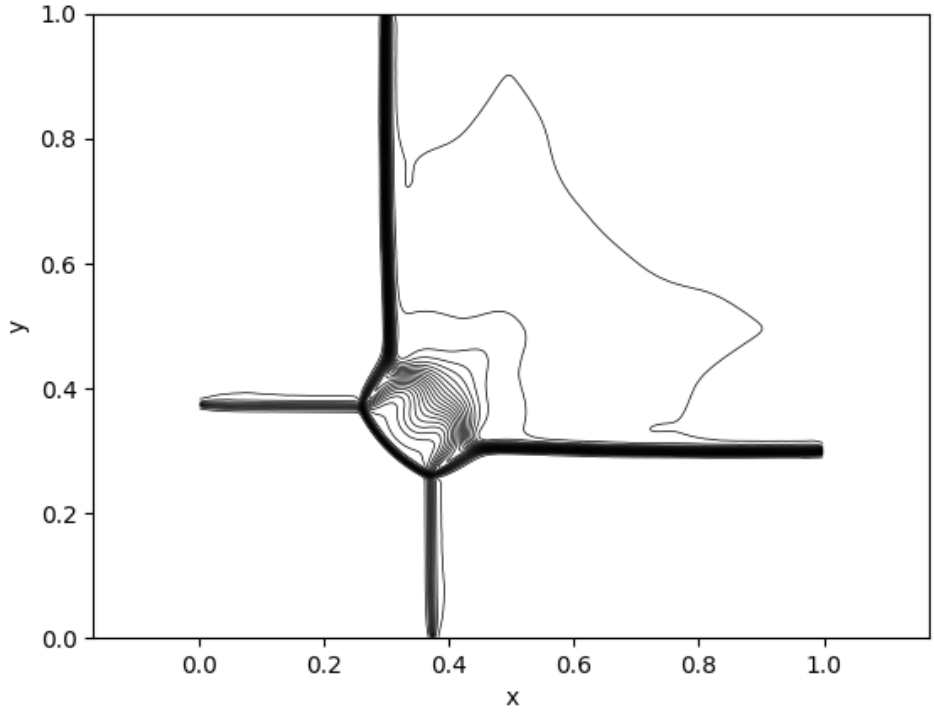

Figure 4.7: Subcell relaxation introduces far too much dissipation in Configuration 3.

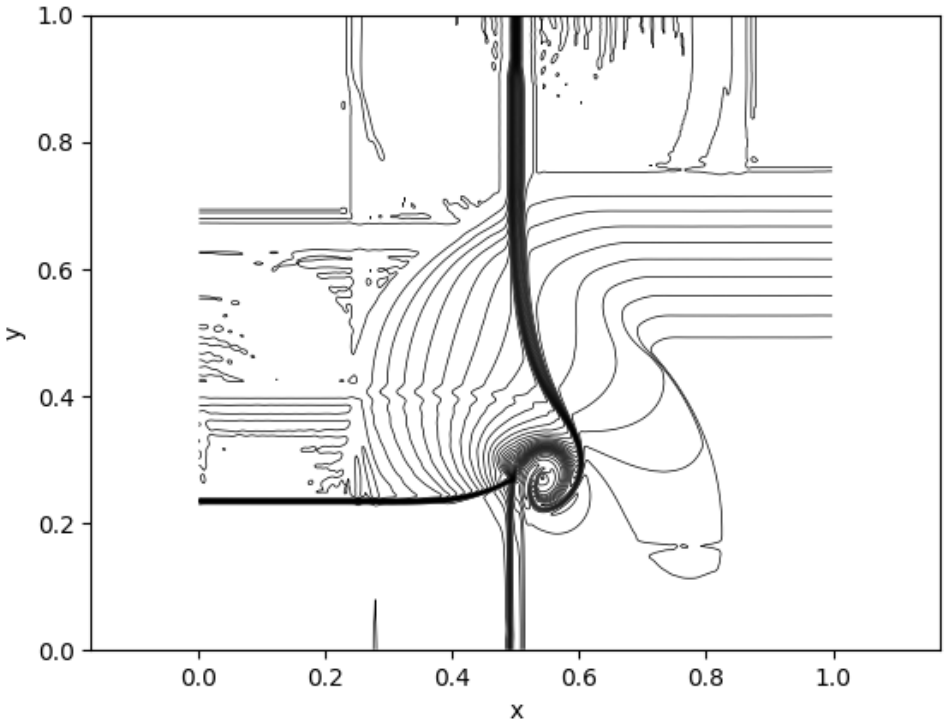

Figure 4.8: The contact roll-up in Configuration 17 is still resolved, but the spurious oscillations not entirely eliminated.

method in its present form is that no polynomial representation is directly available. We can, however, build a polynomial from the reconstructed subcell averages and the midpoint value, apply the hierarchical reconstruction, then set the new subcell averages and point values from the limited polynomial. Figure 4.9 shows the results of applying the hierarchical reconstruction with two choices of polynomial to the case of the moving shock obtained by a cross-section of Configuration 16 at $x = 0.9$. One option (HR2) is to use the quadratic that matches the two subcell averages and point value within each cell. One expects this choice to give third-order accuracy, so to obtain the desired fifth-order we include the next two adjacent subcell averages to form a quartic polynomial in each cell (HR4). In both cases the hierarchical reconstruction is applied at every cell in which at least one non-oscillatory weight is less than 0.001. This threshold is sufficient to detect cells near discontinuities without falsely marking cells in smooth regions. We see that HR2 reduces the amplitude of most of the oscillations but does not remove them entirely, whereas the result from HR4 coincides almost exactly with the unmodified solution.

The hierarchical reconstruction is ineffective because the oscillations generated are smooth features and are not generated in the cells immediately next to the shock. Figure 4.10 shows the unmodified CCRWENO solution at $t = 0.0094$, in which we can see that the discontinuity in the density is smeared asymmetrically producing a region to the left that is almost linear. This linear region produces oscillations at around $t = 0.0205$, which can be seen in Figure 4.11. The asymmetry and production of oscillations appear to result from the flux Jacobian being non-constant, by comparison with Figure 4.12 which shows the same initial condition

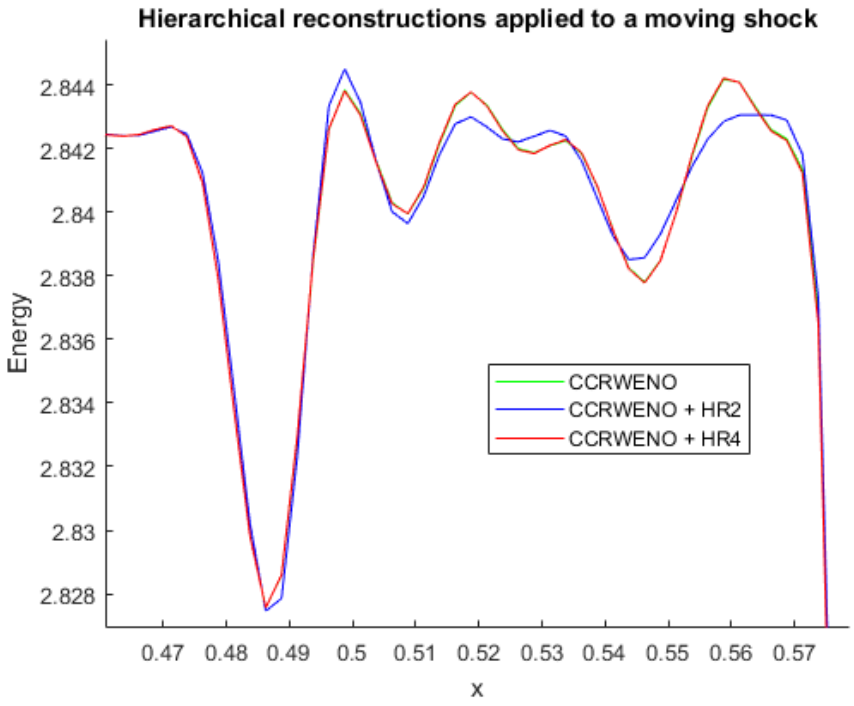

Figure 4.9: The low-degree hierarchical reconstruction provides some improvement.

evolved according to the flux:

$$f(q) = \begin{bmatrix} 0.2338 & 0.6408 & 0.0916 \\ 1.1165 & -0.1697 & 0.4825 \\ -0.8062 & 1.1153 & 1.3882 \end{bmatrix} q \tag{4.16}$$

The coefficient matrix is chosen so that the jump across the shock is an eigenvector with corresponding eigenvalue equal to the shock propagation speed $s = 1.6523$, with the other eigenvectors chosen randomly and the other eigenvalues equal to 0.8 and -1 (note that the Jacobian is not diagonally dominant). Thus the exact solution coincides with the exact solution to the Euler equations. Yet we see in Figure 4.12 that the shock is smeared almost symmetrically and no linear region appears to give rise to oscillations.

Clearly the oscillations arise due to some property of the Euler flux. Therefore a generally-applicable method that prevents them must incorporate information about the flux function, and apparently in a more comprehensive manner than merely evaluating it at point values during time advancement. Even characteristic variables are not entirely sufficient, as the largest oscillations still arise at the same locations and with similar amplitudes even when using an upwind scheme with characteristic variables, as shown in Figure 4.13. This fact indicates that multiple mechanisms produce oscillations, some of which are not addressed by the characteristic decomposition.

Liu and Osher [46] suggest that a componentwise reconstruction will perform well when it degenerates to the first-order reconstruction at discontinuities. This strategy is undesirable in the context of compact reconstructions because the local

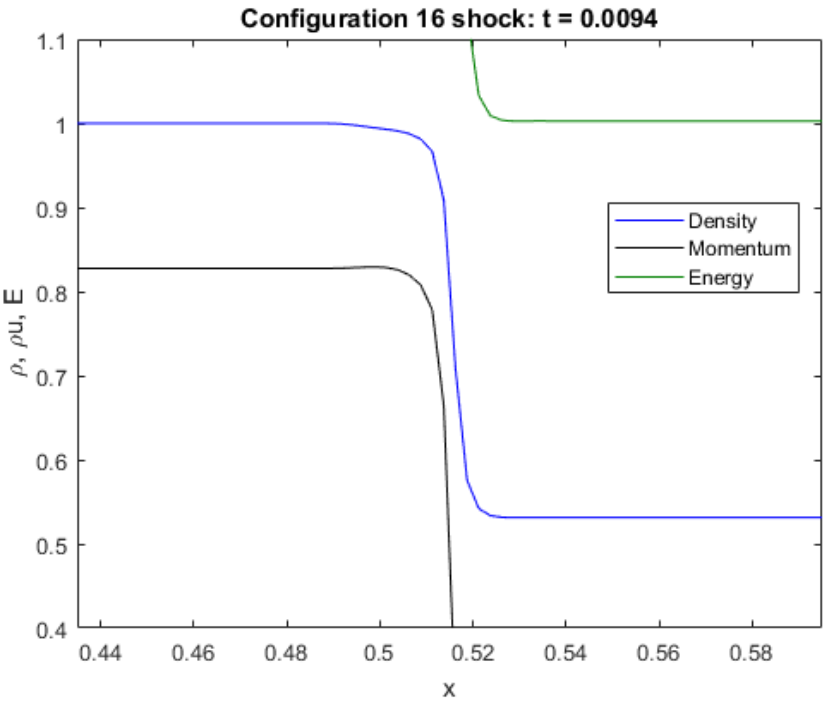

Figure 4.10: The shock in Configuration 16 is smeared asymmetrically and produces a nearly linear region behind itself.

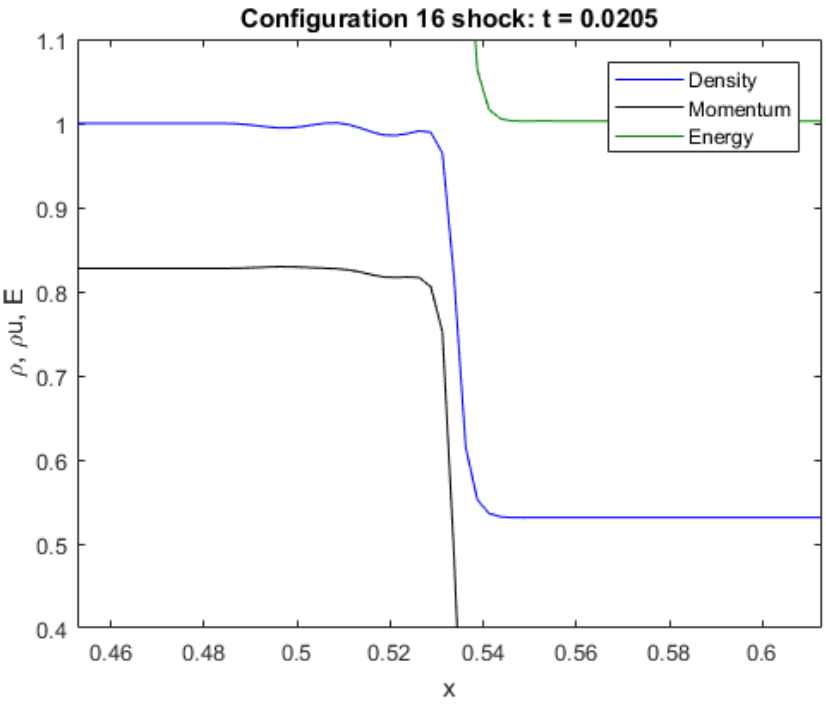

Figure 4.11: The linear region destabilizes into oscillations.

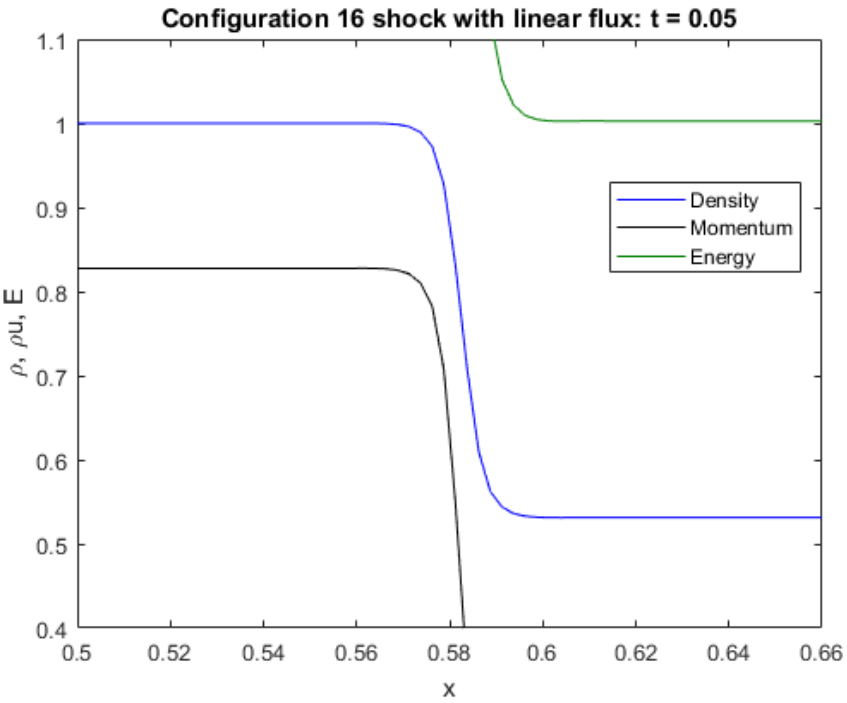

Figure 4.12: A linear flux that gives the same exact solution displays neither a linear region nor oscillations.

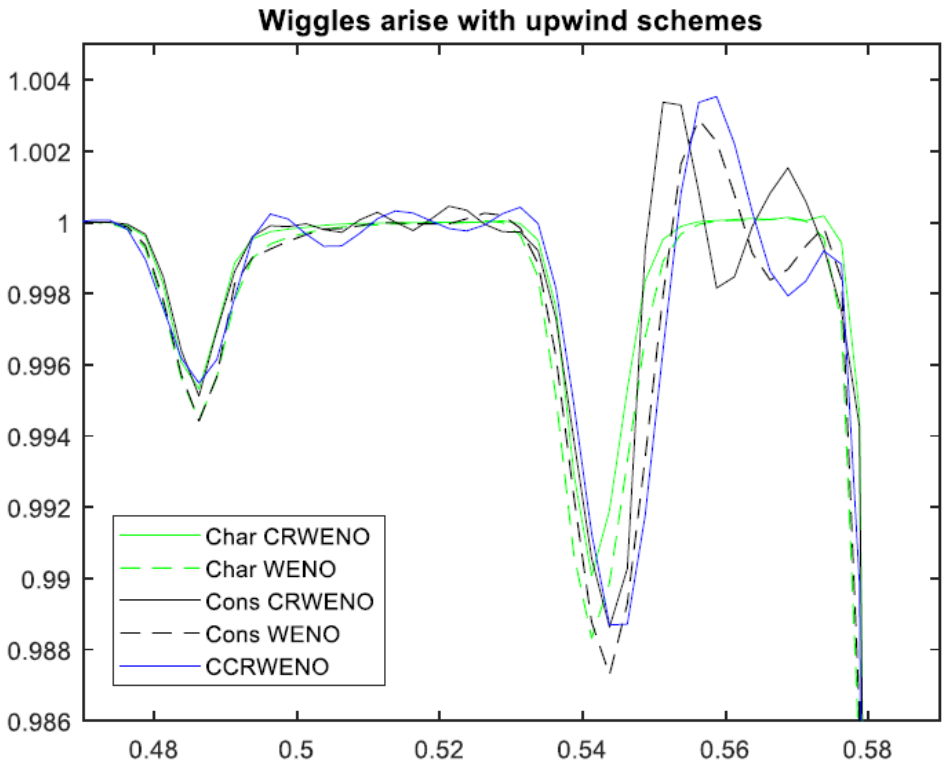

Figure 4.13: Characteristic variables do not prevent all oscillations.

first-order accuracy would pollute the accuracy away from the shock, and is impossible in the context of WENO schemes because even if the reconstruction involves only one subscheme, it will be of higher than first order. A reconstruction that can adapt from fifth-order to first order would need to be of an entirely different type than WENO.

Chapter 5: Conclusion and Future Work

5.1 Conclusion

Compact upwind schemes such as the CRWENO scheme of Ghosh and Baeder [18] lead to solving a block-tridiagonal system when reconstructing characteristic variables, which is necessary for high-order upwind schemes to be stable. Consequently, the CRWENO scheme is less computationally efficient measured by the time required to obtain a specified error compared to the non-compact WENO scheme with the same stencil and equal order of accuracy from [15]. On the other hand, compact schemes reconstruct high-frequency features more efficiently than non-compact schemes of the same order of accuracy and the upwind framework introduces many practical drawbacks, so it is desirable to have a compact scheme that is more computationally efficient at all wavenumbers. This goal, however, cannot be achieved by upwind schemes because the characteristic decomposition is required in order to reconstruct in the upwind direction.

An alternative framework is that of the central schemes, originated by Nessyahu and Tadmor [10]. Their chief advantage is that by averaging over rather than resolving the Riemann fans at cell interfaces they do not require any Riemann solver, nor do they require determining upwind directions by characteristic decomposition.

These features address many of the practical troubles one encounters with the upwind schemes, and allow central schemes to be used as a "black box" solver that only ever evaluates the physical flux function and not any of its derivatives. In practice, however, central schemes of fifth or higher order apparently need the characteristic decomposition to fully avoid spurious oscillations [29]. The literature provides no explanation for this fact that is entirely convincing; however, the juxtaposition of the fourth-order CWENO4 of [28] with the nearly identical fifth-order CWENO5 of [29] (both non-compact) would appear to provide a clue. The only difference is in the reconstruction of the point value, which is fourth-order in CWENO4 but fifth-order in CWENO5. This fact is surprising because practical experience shows that the point-value reconstruction has only a weak influence on the solution, suggesting that this subtle difference is significant in some way. One plausible explanation is that in CWENO5 the subcell and point value reconstructions use different ideal weights whereas in CWENO4 the ideal weights are the same for both. This fact is significant because it implies that the reconstructions CWENO5 cannot be expressed as an evaluation of a single polynomial. In comparison, CWENO4 by its construction forms one polynomial and uses it for both the subcell and point value reconstructions. This inconsistency in CWENO5 could manifest as an instability that the characteristic decomposition mitigates. This hypothesis has the additional benefit of explaining why central schemes beyond specifically fourth order require characteristic variables – they must use different polynomials for their reconstructions.

The fifth-order non-compact central scheme is uniquely defined, so a fifth-order

central scheme that uses one polynomial must be compact. This constraint aligns neatly with the other goal of developing a central compact-reconstruction WENO scheme for efficiency and practical flexibility. Unfortunately, compact schemes do not have a clear correspondence with polynomials so we settled for designing a compact scheme with the same ideal weights. This consideration leads to the main contribution of the work: the CCRWENO method and a variant that has a semi-discrete form. These methods retain the advantages of requiring no Riemann solver and efficiently resolving fine features, and are more efficient than the non-compact CWENO5, but still produce oscillations in many cases demonstrating that the need for characteristic variables was not fully addressed.

To modify the CCRWENO method to truly avoid the characteristic decomposition requires fully understanding why that process is necessary. Close scrutiny of the explanations given in the literature reveals that none fully explains all the experimental knowledge about when one should perform a characteristic decomposition. The second contribution of this work was to consider some promising alternative explanations and decide that the effect of the characteristic decomposition is to make the flux Jacobians in each cell more diagonally dominant. Unfortunately this conclusion does not allow us to modify the core CCRWENO method in such a way as to avoid the characteristic decomposition while retaining its desirable properties, so we consider strategies to alter the CCRWENO results in a way that suppresses the oscillations. Such a limiting strategy should be as computationally inexpensive as possible yet allow a natural extension to multiple dimensions. A simple first-order limiter was tested and found to be far too diffusive, and the hierarchical

reconstruction of [34] and [43] did not quell the oscillations in even one dimension.

5.2 Future Work

We have amassed convincing evidence that the characteristic decomposition functions by improving the diagonal dominance of the Jacobians in a given stencil, and proven that this cannot be done within the reconstruction except with a diagonal transformation matrix. Thus it is necessary to couple the components which would lead to the compact reconstruction being less efficient than its non-compact version. The viability of compact reconstructions therefore depends entirely on the quality of any limiting scheme applied to the reconstructed quantities. The first direction for future work is to develop a more sophisticated limiter for the subcell averages. To preserve the dimension-scalability of CCRWENO this limiting procedure must be inherently multidimensional, and to minimize computational expense it should make maximal use of the information already computed by the reconstruction process.

On the other hand, use of a limiter calls into question the wisdom of a performing a WENO-type reconstruction in the first place. In [43], for example, the initial polynomial to which the hierarchical reconstruction is applied is simply the central reconstruction polynomial. This situation highlights an inherent awkwardness to WENO methods that has been present throughout the current work: the WENO process avoids spurious oscillations in the reconstruction, yet additional effort is needed to prevent spurious oscillations in the evolving solution. This seeming paradox stems from the fact that reconstruction and evolution are fundamentally

different operations; the involvement of time in the latter introduces information propagation behavior which the static reconstruction process does not account for. Consequently, progress might be made toward eliminating the oscillations by employing a unified space-time formulation with a reconstruction that involves data at multiple time steps. This could eliminate the need for upwinding but does not affect Jacobians. Another possibility to achieve the same end is to involve the flux function more heavily in the reconstruction in a way that detects relevant physics. The characteristic decomposition does this, so the objective would be to obtain a similar result using only flux evaluations.

The CCRWENO method presented here requires a uniform grid with no variation in cell widths along each dimension. If a mapping from a non-uniform grid to a computationally uniform grid is not available then the method would need to be reformulated in a way that accounts for variable cell widths. It may not be possible to do this while keeping the ideal weights the same. Alternatively, local variation in cell size could be accomplished by adaptively refining the grid. The CCRWENO method already incorporates two levels of grid resolution – the main grid and the grid of subcells – so it can be applied recursively to successively finer grids in a subregions of coarser grids. The obstacle to this improvement is interpolation of the point values which are still required for time advancement. Midpoint values in the finer cells will correspond to off-center points in the coarser cells and the values at those off-center points will need to be interpolated in a non-oscillatory way. The payoff, however, is that by using coarser grids in locations where resolution is less important the size of the linear system to be solved can decrease. Because

solving the system is the most restrictive barrier to high performance the payoff for implementing the adaptive mesh refinement would be enormous.

Appendix A: Analysis of Linear Schemes by Generating Functions

A.1 Introduction

Consider a linear finite-difference approximation of the derivative operator:

$$\sum_k L_k (q')_{j+k} = \sum_k R_k q_{j+k} \tag{A.1}$$

The coefficients L_k and R_k might be chosen to cancel leading-order terms of the Taylor expansion of the truncation error or to optimize spectral properties. In either case one must solve a system of equations for the coefficients, a system which changes if the stencil (i.e. the set of indices k) changes or if the operator to be approximated changes. In finite-volume schemes the quantities on the right-hand side are cell-averages instead of point values of the solution; thus Taylor expansion of the cell averages becomes necessary, complicating the process of obtaining the coefficients.

A.2 The Generating Function of a Linear Scheme

Suppose $\tilde{u} : \mathbb{R} \to \mathbb{R}$ is an analytic function on some region of interest and let $u_j = \tilde{u}(j\Delta x)$, where Δx is a small grid spacing. Consider the Taylor expansion of

$\tilde{u}((j+k)\Delta x) = u_{j+k}$ about the point $x_j = j\Delta x$:

$$u_{j+k} = \sum_{r=0}^{\infty} k^r \tilde{u}_j^{(r)} \frac{\Delta x^r}{r!} \tag{A.2}$$

There is a clear bijection between these expansions and those of exponentials. Indeed, for each k associate $u_{j+k} \leftrightarrow e^{kz}$:

$$e^{kz} = \sum_{r=0}^{\infty} k^r \frac{z^r}{r!} \tag{A.3}$$

In determining coefficients of a finite-difference scheme we would consider the expansion of a linear combination of u_{j+k} for various values of k and cancel leading powers of Δx. In view of the aforementioned bijection, this procedure is equivalent to cancelling powers of z in the same combination of e^{kz}:

$$\begin{aligned} \sum_k c_k u_{j+k} &= \sum_{r=0}^{\infty} \left(\sum_k c_k k^r \right) \tilde{u}_j^{(r)} \frac{\Delta x^r}{r!} \\ \sum_k c_k e^{kz} &= \sum_{r=0}^{\infty} \left(\sum_k c_k k^r \right) \frac{z^r}{r!} \end{aligned} \tag{A.4}$$

Clearly the $\mathcal{O}(\Delta x^p)$ term in the first expansion is cancelled if and only if the $\mathcal{O}(z^p)$ is cancelled in the second, since their coefficients are the same. Conveniently, the coefficient of $z^r/r!$ can be extracted by taking the rth derivative of the exponential combination and evaluating it at $z = 0$:

$$\sum_k c_k k^r = \left. \frac{d^r}{dz^r} \right|_{z=0} \left(\sum_k c_k e^{kz} \right) \tag{A.5}$$

The utility of the generating function, therefore, is that it combines all the terms in the Taylor expansion into a single object that can be manipulated algebraically and differentiated to obtain the same result that would have been obtained by manipulation of individual terms in the expansion. Moreover, the conversion to the

generating function abstracts away the dependence of the output of an operator on the specific function $\tilde{u}$; observe that no derivatives appear in the z expansions, nor even dependence on the point j. As a result, the coefficients in the expansion cannot depend on $\tilde{u}$ - they must be constants. This restriction enables useful properties of the generating function but limits the type of operator that can be considered.

Definition A.1 (Admissible Operator)**.** *Let $\mathcal{A}$ be the space of analytic functions. An operator $P : \mathcal{A} \to \mathcal{A}$ is admissible if for any $\tilde{u} \in A$ the output $P\tilde{u}$ can be written as:*

$$(P\tilde{u})(y) = \sum_{r=0}^{\infty} P_r \tilde{u}^{(r)}(y) \frac{\Delta x^r}{r!} \tag{A.6}$$

where P_r are constants that depend only on r.

The grid spacing Δx appears on the right-hand side because the operator P may depend on Δx, as in the case of the shift operator which shifts the function $\tilde{u}$ by Δx. From the definition it is clear admissible operators must be linear due to the linearity of differentiation. We can now define the generating function of an admissible operator.

Definition A.2. *Let $P : \mathcal{A} \to \mathcal{A}$ be an admissible operator. Then the generating function $\hat{P}(z)$ of P is:*

$$\hat{P}(z) = \sum_{r=0}^{\infty} P_r \frac{z^r}{r!} \tag{A.7}$$

From this definition it is clear that the transformation $P \to \hat{P}$ is injective.

Theorem A.1. *If the generating functions for operators P and Q are $\hat{P}(z)$ and $\hat{Q}(z)$ respectively, then the generating function of the composition $P \circ Q$ is $\hat{P}(z)\hat{Q}(z)$.*

Proof. Consider the composition of two operators P and Q:

$$\begin{aligned}(P\circ Q)\tilde{u}(j\Delta x) = P(Q\tilde{u})(j\Delta x) &= \sum_{r=0}^{\infty} P_r \frac{d^r}{dx^r}\bigg|_{x=j\Delta x} (Q\tilde{u})\frac{\Delta x^r}{r!} \\ &= \sum_{r=0}^{\infty} P_r \frac{d^r}{dx^r}\bigg|_{x=j\Delta x} \left(\sum_{s=0}^{\infty} Q_s \tilde{u}_j^{(s)} \frac{\Delta x^s}{s!}\right)\frac{\Delta x^r}{r!} \\ &= \sum_{r=0}^{\infty}\sum_{s=0}^{\infty} P_r Q_s \tilde{u}_j^{(r+s)} \frac{\Delta x^{r+s}}{r!s!} \\ &= \sum_{r=0}^{\infty}\sum_{s=0}^{\infty} P_r Q_s \binom{r+s}{r} \tilde{u}_j^{(r+s)} \frac{\Delta x^{r+s}}{(r+s)!}\end{aligned} \tag{A.8}$$

Let $m = r + s$:

$$\begin{aligned}(P\circ Q)\tilde{u}(j\Delta x) &= \sum_{m=0}^{\infty}\sum_{r=0}^{m} P_r Q_{m-r} \binom{m}{r} \tilde{u}_j^{(m)} \frac{\Delta x^m}{m!} \\ &= \sum_{m=0}^{\infty} (PQ)_m \tilde{u}_j^{(m)} \frac{\Delta x^m}{m!}\end{aligned} \tag{A.9}$$

Therefore the composition $P \circ Q$ has the generating function:

$$P\circ Q \leftrightarrow \widehat{P\circ Q}(z) = \sum_{m=0}^{\infty} (PQ)_m \frac{z^m}{m!} \tag{A.10}$$

where the Taylor coefficients depend on those of P and Q themselves according to:

$$(PQ)_m = \sum_{r=0}^{m} P_r Q_{m-r} \binom{m}{r} \tag{A.11}$$

On the other hand, consider the product of $\hat{P}(z)$ and $\hat{Q}(z)$:

$$\begin{aligned}\hat{P}(z)\hat{Q}(z) &= \sum_{m=0}^{\infty} \frac{d^m}{dz^m}\bigg|_{z=0} (\hat{P}(z)\hat{Q}(z))\frac{z^m}{m!} \\ &= \sum_{m=0}^{\infty} \left(\sum_{r=0}^{m} \binom{m}{r} \hat{P}^{(r)}(0)\hat{Q}^{(m-r)}(0)\right)\frac{z^m}{m!} \\ &= \sum_{m=0}^{\infty} \left(\sum_{r=0}^{m} \binom{m}{r} P_r Q_{m-r}\right)\frac{z^m}{m!}\end{aligned} \tag{A.12}$$

since the rth derivative of the generating function at $z = 0$ is the rth Taylor coefficient of the corresponding operator. But this is the same expression that appeared

in the expansion of the composition $P \circ Q$. It follows that $\widehat{P \circ Q}(z) = P(z)Q(z) = Q(z)P(z) = \widehat{Q \circ P}(z)$. □

The restriction that the Taylor coefficients be independent of $\tilde{u}$ implies that we can only consider operators that commute with each other. As long as we confine our attention to linear schemes, however, this is not a relevant limitation.

Corollary A.1. *Let P be an admissible operator. If P has an inverse P^{-1}, then the generating function of the inverse satisfies $1 = \hat{P}(z)\widehat{P^{-1}}(z)$*

Proof. If an operator P is invertible, then its inverse P^{-1} must satisfy $(P \circ P^{-1})(\tilde{u}) = \tilde{u}$. Clearly, then, the Taylor expansion of $(P \circ P^{-1})(\tilde{u})$ must be

$$(P \circ P^{-1})(\tilde{u})(j\Delta x) = \tilde{u}(j\Delta x) = \sum_{r=0}^{\infty} P_r \tilde{u}^{(r)}(j\Delta x)\frac{z^r}{r!} \tag{A.13}$$

where $P_r = \delta_{r,0}$. Then the generating function is simply:

$$\widehat{P \circ P^{-1}}(z) = 1 \tag{A.14}$$

But we know from Theorem A.1 that the generating function of a composition is the product of the individual generating functions. Therefore:

$$1 = \hat{P}(z)\widehat{P^{-1}}(z) \tag{A.15}$$

□

It follows that if $\hat{P}(z) = 0$ for any z then the operator P is not invertible. The prime example of this fact is $P = D$, the differentiation operator. In that case $\hat{D}(z) = z$ which vanishes when $z = 0$, and as we know differentiation is invertible only up to an additive constant.

Table A.1 the generating functions corresponding to operators that frequently arise in the context of finite-difference and finite-volume schemes.

Table A.1: Generating functions for common FD/FV operators

Operation	Definition	P_r	$\hat{P}(z)$
Shift	$(E\tilde{u})(x) = \tilde{u}(x + \Delta x)$	1	e^z
Derivative	$(D\tilde{u})(x) = \Delta x \tilde{u}'(x)$	$\delta_{1,r}$	z
Averaging	$(A\tilde{u})(x) = \frac{1}{\Delta x} \int_{x-\Delta x/2}^{x+\Delta x/2} \tilde{u}(s)\, ds$	$\frac{1+(-1)^r}{2^{(r+1)}(r+1)}$	$\frac{2\sinh(z/2)}{z}$

Remark. *One might find it helpful to think of the variable z in the generating functions as an operator $z = \Delta x D$ where D is the differentiation operator. Then the definition of the shift operator becomes $E = e^{zD}$ which is simply a restatement of Taylor's theorem. This conception can be used to obtain the same practical results as the generating function system, but leads to expressions where one differentiates with respect to the operator z. Rather than trust the formal equivalence between power-series expressions involving the operator z and the corresponding functions of the variable z, we eschew the operator notation from the beginning and deal solely in generating functions. This approach has the advantage that, because we are dealing with functions, all the usual rules for manipulating functions are available due to linearity of the conversion to generating functions and Theorem A.1.*

A.3 Applications

In this section we present applications of the generating function approach to several problems that arise in the construction of finite-volume and finite-difference

schemes.

A.3.1 Truncation Error of Reconstructions

In the development of the CCRWENO method we need to construct subschemes of prescribed order of accuracy for the reconstruction of subcell averages from cell averages. The ordinary approach to this problem would be to construct a table of the Taylor coefficients of the cell- and subcell-averages involved and solve a linear system for the coefficients that cancel leading-order error terms. This approach rapidly becomes tedious because the expansion for the average over one cell does not directly lead to the expansion for the average over a different cell. The generating function system, by abstracting away the location-dependence of the expansion, removes this obstacle. For example, suppose we want a pth-order reconstruction. Let P and Q be operators (e.g. cell average and subcell average). Then the scheme takes the form:

$$\left(\sum_k L_k E^k\right) Q = \left(\sum_k R_k E^k\right) P + T \tag{A.16}$$

where T is an operator that returns the truncation error. We can solve for the generating function of T by considering the generating function of the other two terms.

$$\left(\sum_k L_k e^{kz}\right) \hat{Q}(z) = \left(\sum_k R_k e^{kz}\right) \hat{P}(z) + \hat{T}(z) \tag{A.17}$$

Therefore:

$$\hat{T}(z) = \left(\sum_k L_k e^{kz}\right) \hat{Q}(z) - \left(\sum_k R_k e^{kz}\right) \hat{P}(z) \tag{A.18}$$

The accuracy condition amounts to requiring the first p derivatives of $\hat{T}(z)$ to vanish at $z = 0$. Note that the truncation error expansion is centered on the point corresponding to $k = 0$.

We are interested in ensuring that the subcell and point value reconstructions in the CCRWENO method are stable, in the sense that if a reconstruction is applied to a Fourier mode then the output will be a mode with lesser or equal amplitude. Let A be the cell-average operator and let Q be either the point-value or subcell-average operator and consider a scheme of the form Eq. (A.16). The application of that scheme to a Fourier mode e^{imx} produces a mode of the same wavenumber m with amplitude G:

$$\left(\sum_k L_k E^k\right) QG(m)e^{imx} = \left(\sum_k R_k E^k\right) Ae^{imx} \tag{A.19}$$

For stability we need $|G(m)| \leq 1$ for all m. The amplification factor $G(m)$ can be found using the generating function. Consider an admissible operator P acting on e^{imx}:

$$\begin{aligned} P(e^{imx})(y) &= \sum_{r=0}^{\infty} P_r \left((im)^r e^{imy}\right) \frac{\Delta x^r}{r!} \\ &= \left(\sum_{r=0}^{\infty} P_r \frac{(im\Delta x)^r}{r!}\right) e^{imy} \\ &= \hat{P}(im\Delta x)e^{imy} \end{aligned} \tag{A.20}$$

Applying this fact to both sides of Eq. (A.19) and canceling e^{imy} on both sides gives an expression for the amplification factor:

$$\left(\sum_k L_k e^{ikm\Delta x}\right) \hat{Q}(im\Delta x)G(m) = \left(\sum_k R_k e^{ikm\Delta x}\right) \hat{A}(ikm\Delta x) \tag{A.21}$$

$$G(m) = \frac{\sum_k R_k e^{ikm\Delta x}}{\sum_k L_k e^{ikm\Delta x}} \frac{\hat{A}(im\Delta x)}{\hat{Q}(im\Delta x)} \tag{A.22}$$

We have recovered the result of traditional Fourier analysis but with the added capability of accounting for the effects of averaging.

A.3.2 Error Cancellation in CRWENO Reconstructions

A WENO-type scheme built from a given set of subschemes need not have positive ideal weights. Negative ideal weights can cause instability unless the reconstruction is modified [36], but the special treatment required to prevent instability incurs extra computational expense. One would therefore prefer to design subschemes that together have only positive ideal weights. The CRWENO schemes of Ghosh and Baeder [18] use the following subschemes to reconstruct the right interface value $q_{j+1/2}$ from cell averages (cf. Eq. (1.32)):

$$\begin{aligned}
\frac{2}{3}\hat{q}_{j-1/2}+\frac{1}{3}\hat{q}_{j+1/2} &= \frac{1}{6}\bar{q}_{j-1}+\frac{5}{6}\bar{q}_j & \bar{\omega}_1 &= \frac{1}{5} \\
\frac{1}{3}\hat{q}_{j-1/2}+\frac{2}{3}\hat{q}_{j+1/2} &= \frac{5}{6}\bar{q}_j + \frac{1}{6}\bar{q}_{j+1} & \bar{\omega}_2 &= \frac{1}{2} \\
\frac{2}{3}\hat{q}_{j+1/2} + \frac{1}{3}\hat{q}_{j+3/2} &= \frac{1}{6}\bar{q}_j + \frac{5}{6}\bar{q}_{j+1} & \bar{\omega}_3 &= \frac{3}{10}
\end{aligned} \tag{A.23}$$

The coefficient arrays for the left- and right-hand sides are:

$$L = \begin{bmatrix} \frac{2}{3} & \frac{1}{3} & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 \\ 0 & \frac{2}{3} & \frac{1}{3} \end{bmatrix}, \quad R = \begin{bmatrix} \frac{1}{6} & \frac{5}{6} & 0 \\ 0 & \frac{5}{6} & \frac{1}{6} \\ 0 & \frac{1}{6} & \frac{5}{6} \end{bmatrix} \tag{A.24}$$

Observe that the second subscheme is the reflection of the first about point j and that the third is simply the first shifted one cell to the right. One might conjecture that this structure is sufficient to ensure positive weights and this question can be answered using the generating function system.

Let Q be the left-hand side operator, which returns the function value $\Delta x/2$ to the right: $Q = E^{1/2}$. Also let the right-hand side operator P be the cell-average operator. Then the generating function for truncation error of the first subscheme is:

$$\hat{T}_1(z) = \left(\sum_{k=-1}^{1} L_{1,k} e^{kz} \right) e^{z/2} - \left(\sum_{k=-1}^{1} R_{1,k} e^{kz} \right) \frac{2\sinh(z/2)}{z} \tag{A.25}$$

Since the third subscheme is the first subscheme shifted by one cell (i.e. by Δx) we must have that its truncation error satisfies:

$$\hat{T}_3(z) = e^z \hat{T}_1(z) \tag{A.26}$$

The reflection of the first subscheme through the point j is given by:

$$\left(\sum_{k=-1}^{1} L_{1,k} e^{-kz} \right) e^{-z/2} = \left(\sum_{k=-1}^{1} R_{i,k} e^{-kz} \right) \frac{2\sinh(z/2)}{z} \tag{A.27}$$

The factor $e^{z/2}$ on the left-hand side of the original subscheme becomes $e^{-z/2}$ in order to refer to the correct faces. We can see that the result of the reflection is the same as replacing z in the original subscheme with $-z$, thus the truncation error for the second subscheme is related to that of the first by:

$$\hat{T}_2(z) = \hat{T}_1(-z) \tag{A.28}$$

Eqs. (A.25)-(A.28) hold for any CRWENO scheme constructed by reflecting and shifting an initial subscheme. Suppose the first subscheme is pth-order accurate. Then:

$$\hat{T}_1^{(r)}(0) = 0, \quad r = 0, 1, \cdots, p-1 \tag{A.29}$$

We are interested in finding positive solutions to the system (all derivatives are

evaluated at $z = 0$):

$$\begin{bmatrix} 1 & 1 & 1 \\ \hat{T}_1^{(p)} & \hat{T}_2^{(p)} & \hat{T}_3^{(p)} \\ \hat{T}_1^{(p+1)} & \hat{T}_2^{(p+1)} & \hat{T}_3^{(p+1)} \end{bmatrix} \begin{bmatrix} \bar{\omega}_1 \\ \bar{\omega}_2 \\ \bar{\omega}_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \tag{A.30}$$

where we normalize the ideal weights by requiring that they sum to 1. Using Eqs. (A.25)-(A.28) we can express the truncation errors of the second and third subschemes in terms of those of the first:

$$\begin{bmatrix} 1 & 1 & 1 \\ \hat{T}_1^{(p)} & \hat{T}_2^{(p)} & \hat{T}_3^{(p)} \\ \hat{T}_1^{(p+1)} & \hat{T}_2^{(p+1)} & \hat{T}_3^{(p+1)} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ a & (-1)^p a & a \\ b & (-1)^{p+1} b & (p+1)a + b \end{bmatrix} \tag{A.31}$$

where $a = \hat{T}_1^{(p)}$, $b = \hat{T}_1^{(p+1)}$ and we have used the fact that:

$$\left.\frac{d^r}{dz^r}\right|_{z=0} e^z \hat{T}_1(z) = \sum_{s=0}^{r} \binom{r}{s} \hat{T}_1^{(s)} = \sum_{s=p}^{r} \binom{r}{s} \hat{T}_1^{(s)} \tag{A.32}$$

since the first subscheme is pth-order accurate. The system Eq. (A.30) can be solved by Cramer's rule:

$$\begin{aligned} D &= a^2(p+1)((-1)^p - 1) \\ \bar{\omega}_1 &= \frac{(-1)^p}{D}(2ab + a^2(p+1)) \\ \bar{\omega}_2 &= \frac{1}{D}(-a^2(p+1)) \\ \bar{\omega}_3 &= \frac{(-1)^p}{D}(-2ab) \end{aligned} \tag{A.33}$$

The determinant D is nonzero only if the pth-order truncation coefficient a is nonzero (which it is by definition of p) and if p is odd, in which case $D < 0$. Taking p to be

odd, Eq. (A.33) becomes:

$$\begin{aligned} D &= -2a^2(p+1) \\ \bar{\omega}_1 &= \frac{-1}{D}(2ab + a^2(p+1)) \\ \bar{\omega}_2 &= \frac{-1}{D}(a^2(p+1)) \\ \bar{\omega}_3 &= \frac{-1}{D}(-2ab) \end{aligned} \tag{A.34}$$

Thus for positivity of all the $\bar{\omega}_i$ we need:

$$\begin{aligned} 0 &< 2ab + a^2(p+1) \\ 0 &< a^2(p+1) \\ 0 &< -2ab \end{aligned} \tag{A.35}$$

The second of these inequalities is satisfied by default. The third implies that a and b must have opposite sign and $b \neq 0$, and combined with the first gives the necessary and sufficient condition:

$$0 < -2ab < a^2(p+1) \tag{A.36}$$

Since the two nonzero terms are positive, the latter inequality is equivalent to $2|a||b| < a^2(p+1) \Leftrightarrow 2|b| < |a|(p+1)$ which can be written:

$$\frac{|b|}{|a|} < \frac{p+1}{2} \tag{A.37}$$

Thus positive weights exist if the truncation error coefficients of the first subscheme do not grow too rapidly. The generating function method enabled this result by allowing us to easily relate the truncation errors of the subschemes.

A.3.3 Dispersion, Dissipation, and the Modified Equation

The generating function method extends straightforwardly to space-time discretizations by involving multiple variables in the generating function, which allows the dispersion and dissipation of a numerical method to be analyzed holistically. Analogously to the one-variable case, suppose an operator $P : \mathcal{A} \to \mathcal{A}$ is given by:

$$(P\tilde{u})(x,t) = \sum_{r,s=0}^{\infty} P_{r,s} \left(\frac{\partial^{r+s}}{\partial x^r \partial^s t} \tilde{u}(x,t) \right) \frac{\Delta x^r \Delta t^s}{r!s!} \tag{A.38}$$

Then the corresponding generating function is:

$$\hat{P}(\xi,\tau) = \sum_{r,s=0}^{\infty} P_{r,s} \frac{\xi^r \tau^s}{r!s!} \tag{A.39}$$

As before, the space $\mathcal{A}$ consists of all functions that are analytic on a region of interest and $P_{r,s}$ are constants that depend only on r and s. The construction of the generating function parallels the single-variable process. For example, consider the first-order upwind scheme for the linear advection equation:

$$q_j^{n+1} = q_j - \sigma(q_j^n - q_{j-1}^n) \tag{A.40}$$

where σ is the CFL number. Relative to the value q_j^n, the value q_j^{n+1} at the next time step is shifted in time in the same way that q_{j-1}^n is shifted in space. Converting both sides of Eq. (A.40) to their generating functions gives:

$$e^{\tau} = 1 - \sigma(1 - e^{-z}) \tag{A.41}$$

As before, the action of this method on a Fourier mode e^{imx} can be found by simply replacing z by $im\Delta x$. By analogy with the traditional Fourier analysis, one sees

that the analog of the amplification factor is the expression e^{τ} since both describe the change at a point over one time step. The conversion to generating functions thereby gives the amplification factor immediately:

$$G(m\Delta x) = 1 - \sigma(1 - e^{-im\Delta x}) = (1 - \sigma + \sigma\cos(m\Delta x)) - i\sigma\sin(m\Delta x) \tag{A.42}$$

The generating function method can be used to analyze more complicated methods. In the general case, the conversion of a fully discrete method can be written as:

$$M(e^{\tau}) = P(z) \tag{A.43}$$

The amplification factor can be found simply as:

$$G = M^{-1}(P(im\Delta x)) \tag{A.44}$$

where M^{-1} is the inverse function of M (not its reciprocal).

With propagation speed a and an initial condition given by $q(x,0) = e^{imx}$, the exact solution of the linear advection equation Eq. (1.49) is clearly:

$$q(x,t) = e^{im(x-at)} = e^{imx}e^{-imat} = \exp\left(-im\Delta x\sigma\frac{t}{\Delta t}\right)q(x,0) \tag{A.45}$$

where $\sigma = a\Delta t/\Delta x$. Discretizing this exact solution:

$$q_j^{n+1} = e^{-im\Delta x\sigma}q_j^n \tag{A.46}$$

Therefore the exact amplification factor is $G = \exp(-im\Delta x\sigma)$. The amplification factor Eq. (A.44) of the fully discrete method produces an approximation G^* to the exact amplification factor.

$$G^* = \exp(-im\Delta x(\sigma_R^* + i\sigma_I^*)) = \exp(-im\Delta x\sigma_R^*)\exp(m\Delta x\sigma_I^*) \tag{A.47}$$

where σ_R^* and σ_I^* are respectively the real and imaginary parts of σ^*. We see that the σ_I^* controls the magnitude of G^* (the dissipation) while σ_R^* controls the propagation speed of the discrete solution and thus the dispersion. σ_R^* and σ_I^* can be found by:

$$\sigma_R^* = \frac{-1}{m\Delta x}\Im(\ln G^*) = -\frac{\ln G^* - \overline{\ln G^*}}{m\Delta x} = -\frac{1}{m\Delta x}\ln\left(\frac{G^*}{\overline{G^*}}\right) \tag{A.48}$$

$$\sigma_I^* = \frac{1}{m\Delta x}\ln|G^*| \tag{A.49}$$

σ_R^* should approximate σ while σ_I^* should approximate 0.

Thus far the generating function approach has provided an alternative, perhaps more elegant path to the same results that traditional Fourier analysis would obtain. We close this appendix by describing and demonstrating an application in which it has a unique and decisive advantage. Warming and Hyett [47] introduced the concept of a modified equation, a partial differential equation for which the solution of the discretized equation is an exact solution. Given a discretization of a PDE Eq. (A.43), the modified equation takes the form:

$$\Delta t q_t = \sum_{r=0}^{\infty} c_r \Delta x^r \frac{\partial^r q}{\partial x^r} \tag{A.50}$$

which we have adapted slightly compared to [47] so that the coefficients c_r are all dimensionless. In practice, one traditionally obtains the modified equation by taking Taylor expansions of the solution at each point involved in the fully discrete scheme, then isolating the q_t term. At this point the right-hand side contains derivatives with respect to both x and t. The time derivatives are cancelled by first differentiating *the entire equation* with respect to t and substituting the result in place of the q_{tt} term on the right-hand side, and repeating this process for all terms that involve

time derivatives. This approach is immensely tedious due to the need to differentiate and track infinitely many terms at each step. It is also not always clear in advance how many terms at each step one must keep in order to obtain the leading terms in the final result. The generating function method sidesteps all of these problems.

Observe that Eq. (A.50) involves a single time derivative on the left and only spatial derivatives on the right-hand side. Thus after transforming to generating functions, the left-hand side must be simply τ (the generating function for a lone first derivative with respect to time) while the generating function on the right must not depend on τ at all. Because the solution to the modified equation Eq. (A.50) and the fully discrete method Eq. (A.43) are the same, the generating functions for the modified equation and the fully discrete method must be equivalent i.e. each can be obtained from the other. We desire, therefore, to obtain an equation for τ in terms of z from Eq. (A.43). But this is simple:

$$M(e^{\tau}) = P(z) \Rightarrow \tau = \ln(M^{-1}(P(z))) = \ln G(z) \tag{A.51}$$

In this sense of generating functions, the modified equation is simply the natural logarithm of the amplification factor. To demonstrate, consider again the first-order upwind method Eq. (A.40). The generating function for the modified equation is:

$$Y(z) = \ln(1 - \sigma(1 - e^{-z})) \tag{A.52}$$

The coefficients of individual terms can be extracted by evaluating derivatives of

$Y(z)$ at $z = 0$. Clearly $Y(0) = 0$.

$$
\begin{aligned}
Y'(z) &= \frac{-\sigma e^{-z}}{1-\sigma+\sigma e^{-z}} \Rightarrow Y'(0) = -\sigma \\
Y''(z) &= \frac{(\sigma-\sigma^2)e^{-z}}{(1-\sigma+\sigma e^{-z})^2} \Rightarrow Y''(0) = \sigma - \sigma^2 \\
Y'''(z) &= \frac{(\sigma-\sigma^2)e^{-z}(2\sigma e^{-z} - (1-\sigma+\sigma e^{-z}))}{(1-\sigma+\sigma e^{-z})^3} \Rightarrow Y'''(0) = (2\sigma-1)(\sigma-\sigma^2)
\end{aligned}
\tag{A.53}
$$

Therefore the first three terms of the modified equation are (note the division by factorials due to the definition of the generating function):

$$
\Delta t q_t = -\sigma \Delta x q_x + (\sigma - \sigma^2)\frac{\Delta x^2}{2} q_{xx} + (2\sigma-1)(\sigma-\sigma^2)\frac{\Delta x^3}{6} q_{xxx} + \cdots \tag{A.54}
$$

Compare this result with the modified equation presented in e.g. [9] and then compare the effort required to obtain it by generating functions versus the effort required by the traditional method described there.

Bibliography

[1] Sergei Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 89(3):271–306, 1959.

[2] Constantine M Dafermos. Hyperbolic conservation laws in continuum physics, volume 325 of grundlehren der mathematischen wissenschaften [fundamental principles of mathematical sciences], 2010.

[3] Eitan Tadmor. Entropy stable scheme. In *Handbook of Numerical Analysis*, volume 17, pages 467–493. Elsevier, 2016.

[4] Randall J LeVeque. *Numerical Methods for Conservation Laws*. BirkhaØ user Basel, 1992.

[5] Randall J LeVeque. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.

[6] Alexander Kurganov. Central schemes: A powerful black-box solver for nonlinear hyperbolic pdes. In *Handbook of Numerical Analysis*, volume 17, pages 525–548. Elsevier, 2016.

[7] Philip Roe. Approximate riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics*, 27:250–258, 1978.

[8] Eleuterio F Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.

[9] Richard H Pletcher, John C Tannehill, and Dale Anderson. *Computational fluid mechanics and heat transfer*. CRC Press, 2013.

[10] Haim Nessyahu and Eitan Tadmor. Non-oscillatory central differencing for hyperbolic conservation laws. *Journal of Computational Physics*, 87(2):408–463, 1990.

[11] Peter D Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Communications on pure and applied mathematics*, 7(1):159–193, 1954.

[12] Kurt O Friedrichs. Symmetric hyperbolic linear differential equations. *Communications on pure and applied Mathematics*, 7(2):345–392, 1954.

[13] Alexander Kurganov and Eitan Tadmor. New high-resolution central schemes for nonlinear conservation laws and convection–diffusion equations. *Journal of Computational Physics*, 160(1):241–282, 2000.

[14] Xu-Dong Liu, Stanley Osher, and Tony Chan. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics*, 115(1):200–212, 1994.

[15] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[16] Andrew K Henrick, Tariq D Aslam, and Joseph M Powers. Mapped weighted essentially non-oscillatory schemes: achieving optimal order near critical points. *Journal of Computational Physics*, 207(2):542–567, 2005.

[17] Debojyoti Ghosh. *Compact-Reconstruction Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*. PhD thesis, University of Maryland, 2012.

[18] Debojyoti Ghosh and James D Baeder. Compact reconstruction schemes with weighted eno limiting for hyperbolic conservation laws. *SIAM Journal on Scientific Computing*, 34(3):A1678–A1706, 2012.

[19] Sanjiva K Lele. Compact finite difference schemes with spectral-like resolution. *Journal of Computational Physics*, 103(1):16–42, 1992.

[20] Rafael Borges, Monique Carmona, Bruno Costa, and Wai Sun Don. An improved weighted essentially non-oscillatory scheme for hyperbolic conservation laws. *Journal of Computational Physics*, 227(6):3191–3211, 2008.

[21] Nail K Yamaleev and Mark H Carpenter. A systematic methodology for constructing high-order energy stable weno schemes. *Journal of Computational Physics*, 228(11):4248–4272, 2009.

[22] Sigal Gottlieb and Chi-Wang Shu. Total variation diminishing runge-kutta schemes. *Mathematics of computation of the American Mathematical Society*, 67(221):73–85, 1998.

[23] Sigal Gottlieb, Chi-Wang Shu, and Eitan Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM review*, 43(1):89–112, 2001.

[24] Chi-Wang Shu. A survey of strong stability preserving high order time discretizations. *Collected Lectures on the Preservation of Stability Under Discretization*, 109:51–65, 2002.

[25] S Gottlieb and David I Ketcheson. Time discretization techniques. In *Handbook of Numerical Analysis*, volume 17, pages 549–583. Elsevier, 2016.

[26] Marino Zennaro. Natural continuous extensions of runge-kutta methods. *Mathematics of Computation*, 46(173):119–133, 1986.

[27] Franca Bianco, Gabriella Puppo, and Giovanni Russo. High-order central schemes for hyperbolic systems of conservation laws. *SIAM Journal on Scientific Computing*, 21(1):294–322, 1999.

[28] Doron Levy, Gabriella Puppo, and Giovanni Russo. Central weno schemes for hyperbolic systems of conservation laws. *ESAIM: Mathematical Modelling and Numerical Analysis*, 33(3):547–571, 1999.

[29] Jianxian Qiu and Chi-Wang Shu. On the construction, comparison, and local characteristic decomposition for high-order central weno schemes. *Journal of Computational Physics*, 183(1):187–209, 2002.

[30] Doron Levy, Gabriella Puppo, and Giovanni Russo. A fourth-order central weno scheme for multidimensional hyperbolic systems of conservation laws. *SIAM Journal on Scientific Computing*, 24(2):480–506, 2002.

[31] Paul Woodward and Phillip Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of computational physics*, 54(1):115–173, 1984.

[32] Dinshaw S Balsara. Multidimensional hlle riemann solver: Application to euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 229(6):1970–1993, 2010.

[33] Doron Levy, Gabriella Puppo, and Giovanni Russo. Compact central weno schemes for multidimensional conservation laws. *SIAM Journal on Scientific Computing*, 22(2):656–672, 2000.

[34] Yingjie Liu, Chi-Wang Shu, Eitan Tadmor, and Mengping Zhang. Central discontinuous galerkin methods on overlapping cells with a nonoscillatory hierarchical reconstruction. *SIAM Journal on Numerical Analysis*, 45(6):2442–2467, 2007.

[35] Gary A Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of computational physics*, 27(1):1–31, 1978.

[36] Jing Shi, Changqing Hu, and Chi-Wang Shu. A technique of treating negative weights in weno schemes. *Journal of Computational Physics*, 175(1):108–127, 2002.

[37] Yingjie Liu. Central schemes on overlapping cells. *Journal of Computational Physics*, 209(1):82–104, 2005.

[38] Peter D Lax and Xu-Dong Liu. Solution of two-dimensional riemann problems of gas dynamics by positive schemes. *SIAM Journal on Scientific Computing*, 19(2):319–340, 1998.

[39] Carsten W Schulz-Rinne, James P Collins, and Harland M Glaz. Numerical solution of the riemann problem for two-dimensional gas dynamics. *SIAM Journal on Scientific Computing*, 14(6):1394–1414, 1993.

[40] Kilian Cooley and James Baeder. A central compact-reconstruction weno method for hyperbolic conservation laws. In *2018 AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, 2018.

[41] Zhengfu Xu and Chi-Wang Shu. Anti-diffusive flux corrections for high order finite difference weno schemes. *Journal of Computational Physics*, 205(2):458–485, 2005.

[42] Lin Fu, Xiangyu Y Hu, and Nikolaus A Adams. A family of high-order targeted eno schemes for compressible-fluid simulations. *Journal of Computational Physics*, 305:333–359, 2016.

[43] Yingjie Liu, Chi-Wang Shu, Eitan Tadmor, and Mengping Zhang. Non-oscillatory hierarchical reconstruction for central and finite volume schemes. *Communications in Computational Physics*, 2(5):933–963, 2007.

[44] Chi-Wang Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws. In *Advanced numerical approximation of nonlinear hyperbolic equations*, pages 325–432. Springer, 1998.

[45] ke Björck and Gene H Golub. Numerical methods for computing angles between linear subspaces. *Mathematics of computation*, 27(123):579–594, 1973.

[46] Xu-Dong Liu and Stanley Osher. Convex eno high order multi-dimensional schemes without field by field decomposition or staggered grids. *Journal of computational physics*, 142(2):304–330, 1998.

[47] RF Warming and BJ Hyett. The modified equation approach to the stability and accuracy analysis of finite-difference methods. *Journal of computational physics*, 14(2):159–179, 1974.