

UC Santa Cruz

UC Santa Cruz Electronic Theses and Dissertations

Title

High-order Kernel-based Finite Volume Methods for Systems of Hyperbolic Conservation Laws

Permalink

<https://escholarship.org/uc/item/7bv507r6>

Author

May, Ian Charles Tromba

Publication Date

2024

Copyright Information

This work is made available under the terms of a Creative Commons Attribution-ShareAlike License, available at <https://creativecommons.org/licenses/by-sa/4.0/>

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA
SANTA CRUZ

**HIGH-ORDER KERNEL-BASED FINITE VOLUME METHODS FOR
SYSTEMS OF HYPERBOLIC CONSERVATION LAWS**

A dissertation submitted in partial satisfaction of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in

APPLIED MATHEMATICS

by

Ian C. T. May

March 2024

The Dissertation of Ian C. T. May
is approved:

Professor Dongwook Lee, Chair

Professor Hongyun Wang

Professor Nicholas Brummell

Professor Francis Giraldo

Peter Biehl
Vice Provost and Dean of Graduate Studies

Copyright © by

Ian C. T. May

2024

Table of Contents

List of Figures	vi
List of Tables	xiii
Abstract	xiv
Dedication	xvi
Acknowledgments	xvii
1 Introduction	1
1.1 An overview of the finite volume method	4
1.1.1 Weighted Essentially Non-Oscillatory (WENO) methods in one dimension	6
1.2 The content and structure of this dissertation	9
2 Linear and nonlinear kernel methods for function reconstruction	12
2.1 Introduction	12
2.2 Reproducing kernel Hilbert spaces and optimal recovery	14
2.3 Symmetric kernel approximation	17
2.3.1 Symmetric kernel-based interpolation	17
2.3.2 Optimal recovery	19
2.3.3 Inclusion of polynomials	24
2.4 Asymmetric kernel approximation	26
2.5 Nonlinear reconstruction for functions with poor regularity	29
2.5.1 Stencils and substencils	30
2.5.2 Adaptive order WENO	35
2.5.3 Smoothness indicators and nonlinear weights	38

3	A kernel-based finite volume method for the compressible Euler equations	42
3.1	Introduction	42
3.2	Selection of reconstruction variables	44
3.2.1	Characteristic variables	46
3.2.2	Linearized primitive variables	48
3.3	Avoiding WENO via a KXRCF indicator	52
3.4	Positivity preservation	56
3.4.1	Constraining density	60
3.4.2	Constraining pressure	61
3.5	An aside on time integrators	62
3.5.1	Time step size selection	65
3.6	Numerical results	68
3.6.1	Isentropic vortex	70
3.6.2	Sod shock tube	73
3.6.3	Richtmeyer-Meshkov instability	75
3.6.4	Astrophysical jets	79
3.6.5	Taylor-Green vortex	82
3.6.6	Rayleigh-Taylor instability	86
4	A kernel-based finite volume method for ideal magnetohydrodynamics	90
4.1	Introduction	90
4.2	Solenoidal magnetic fields	92
4.2.1	The projection method	94
4.2.2	Constrained transport	95
4.2.3	Eight-wave formulation	97
4.2.4	Generalized Lagrange multipliers	98
4.3	KXRCF indicators, reconstruction variables, and positivity	101
4.3.1	KXRCF troubled cell indicator	102
4.3.2	Reconstruction variables	103
4.3.3	Positivity preservation	104
4.4	Treatment of GLM source terms	107
4.4.1	Terms involving field divergence	108
4.4.2	Terms involving $\nabla\psi$	110
4.5	Implementation details	112
4.5.1	The Kokkos library	113
4.5.2	Overview of the solver	114
4.6	Numerical results	117
4.6.1	Isodensity vortex	119
4.6.2	Brio-Wu shock tube	122
4.6.3	Orszag-Tang vortex	125

4.6.4	Field loop advection	128
4.6.5	Jet generation problem	133
4.6.6	Astrophysical jet	135
5	Conclusion	139
5.0.1	Directions for future study	142
	Bibliography	144

List of Figures

- 2.1 The full radius-2 stencil \mathcal{S}_0 is shown on the left with its five substencils, $\mathcal{S}_q, q = 1, \dots, 5$, on the right. The cell with the diamond in each (sub) stencil indicates the central cell where reconstruction is performed. The full stencil \mathcal{S}_0 has 13 cells, while each of the substencils has five. . . . 31
- 2.2 The full radius-3 stencil \mathcal{S}_0 is shown on the left with its five substencils, $\mathcal{S}_q, q = 1, \dots, 5$, on the right. The cell with the diamond in each (sub)stencil indicates the central cell where reconstruction is being performed. The full stencil \mathcal{S}_0 has 29 cells, the central substencil \mathcal{S}_1 has 13 cells, and each of the remaining biased substencils, $\mathcal{S}_2, \dots, \mathcal{S}_5$, has 10. . . 32

3.1	Shown is a trace of the density in Sod shock tube problem as obtained from four different cases. The solid black line shows the exact solution. The dashed lines with circle and square markers show the results of the radius $R = 2$ and $R = 3$ schemes in the grid-aligned configuration. The dotted lines with cross and triangle markers show the results of the radius $R = 2$ and $R = 3$ schemes in the grid-tilted configuration. The inset shows a zoom-in of the region near contact discontinuity.	74
3.2	Shown is the density field for the Richtmeyer-Meshkov instability at the final time of $t = 3.33$ as solved by the radius $R = 2$ scheme on a grid with spacing $\Delta = 1/512$. To highlight the interface the view has been zoomed into the region $[5/2, 11/2] \times [0, 1]$	75
3.3	Shown is the density field for the Richtmeyer-Meshkov instability at the final time of $t = 3.33$ as solved by the radius $R = 3$ scheme on a grid with spacing $\Delta = 1/512$. To highlight the interface the view has been zoomed into the region $[5/2, 11/2] \times [0, 1]$	77
3.4	Shown in black are the cells flagged for WENO reconstruction at the final time for the radius $R = 2$ scheme. The view has been zoomed in to match Figure 3.2.	78

3.5	Shown are the logarithmic density fields for the high-density (left) and low density (right) astrophysical jets in two dimensions at the final times of $t = 0.002$ and $t = 0.04$ respectively. In both cases radius $R = 3$ stencils were used on grids with spacing $\Delta = 1/512$	79
3.6	The left panel shows the logarithmic density field for the low-density astrophysical jet in three dimensions at the final time of $t = 0.035$ as solved by the radius $R = 2$ scheme on a grid with spacing $\Delta = 1/384$. The right panel shows the corresponding numerical Schlieren image defined as $\ln(1 + \nabla\rho)$	81
3.7	The $Q = 0.1$ isosurface of the Q-criterion colored by velocity magnitude is shown for the Taylor-Green vortex at nondimensional time $t = 8$ on a grid with spacing $\Delta = \pi/192$	85
3.8	The evolution of the kinetic energy dissipation rate is compared against the fully converged reference data from [110].	85

- 3.9 Shown is the density field for the viscous Rayleigh-Taylor instability at the final time of $t = 2.5$ with Reynolds number $Re = 20,000$ and Froude number $Fr = 1$. These results were obtained with the radius $R = 3$ scheme on a grid with spacing $\Delta = 1/1024$, though they are insensitive to both grid resolution and to the stencil radius. Note that the domain has been rotated 90° clockwise, such that the y -axis points to the right, for the sake of plotting. 88
- 4.1 The quadrature points associated to a single cell in 2 dimensions are shown with $N = 3$ points per direction. The blue crosses indicate the points on the cell boundary where the Riemann states are located, and the red diamonds indicate the points in the interior. The dashed box highlights how one line of N interior Gauss-Legendre points can be augmented by two boundary points to give $N + 2$ Gauss-Legendre-Lobatto points. Similar groupings hold throughout the cell. 110

4.2	One dimensional traces of the density in the Brio-Wu shock tube problem are shown for four different configurations, and compared to a high resolution solution. Red circles and blue squares correspond to the grid-aligned cases with the radius $R = 2$ and $R = 3$ schemes respectively. Green crosses and purple triangles correspond to the grid-tilted cases with the radius $R = 2$ and $R = 3$ schemes respectively. The inset zooms in on the density spike to the right of the rarefaction.	124
4.3	The density (left column) and magnetic pressure (right column) are shown for the two dimensional Orszag-Tang vortex at times of $t = 0.5$ (top row) and $t = 1$ (bottom row). The radius $R = 2$ scheme was used with a grid spacing of $\Delta = 1/1024$	127
4.4	The density (left column) and magnetic pressure (right column) are shown for the two dimensional Orszag-Tang vortex at times of $t = 0.5$ (top row) and $t = 1$ (bottom row). The radius $R = 3$ scheme was used with a grid spacing of $\Delta = 1/1024$	128
4.5	The density (left column) and magnetic pressure (right column) are shown for the three dimensional Orszag-Tang vortex at times of $t = 0.5$ (top row) and $t = 1$ (bottom row). The radius $R = 2$ scheme was used with a grid spacing of $\Delta = 1/256$	129

- 4.6 Shown is the magnetic pressure in the two dimensional field loop problem as obtained by the radius $R = 2$ scheme (left) and radius $R = 3$ scheme (right) at the final time of $t = 1$. Both cases utilize a grid spacing of $\Delta = 1/256$ 130
- 4.7 Shown is the magnetic pressure in the three dimensional field loop problem as obtained by the radius $R = 2$ scheme at the final time of $t = 1$ on a grid with spacing $\Delta = 1/256$. To more easily visualize the solution all cells with $p_b < 10^{-8}$ have been omitted. 132
- 4.8 The density (top left) and gas pressure (bottom left) are shown along the $y = 0$ plane of the jet generation problem as solved by the $R = 2$ scheme to a final time of $t = 5$ on a grid with spacing $\Delta = 24/320$. The right panel shows one magnetic field line colored by magnetic pressure. 134
- 4.9 The left two panels show the logarithmic density and magnetic pressure fields in the magnetized astrophysical jet problem at $t = 0.002$ as obtained from the radius $R = 2$ scheme. The right two panels show the same results as obtained from the radius $R = 3$ scheme. Both schemes utilize a grid spacing of $\Delta = 1/1024$ 136

4.10 The logarithmic density (left) and magnetic pressure (right) are shown
for the three dimensional magnetized astrophysical jet as obtained with
the radius $R = 2$ scheme at the final time $t = 0.002$ on a grid with
spacing $\Delta = 1/320$ 137

List of Tables

3.1	Time step size controller parameters for the considered schemes. The parameters for the $[3_+^*]$ schemes come from [79] and those for the SSP schemes come from [30].	66
3.2	Shown are the experimental orders of convergence for the described method with radius $R = 2$ and $R = 3$ stencils as tested on the isentropic vortex problem.	71
4.1	Shown are the experimental orders of convergence for the described method with radius $R = 2$ and $R = 3$ stencils as tested on the two dimensional isodensity magnetic vortex problem.	121
4.2	Shown are the experimental orders of convergence for the described method with radius $R = 2$ and $R = 3$ stencils as tested on the three dimensional isodensity magnetic vortex problem.	122

Abstract

High-Order Kernel-Based Finite Volume Methods for Systems of Hyperbolic Conservation Laws

by

Ian C. T. May

Systems of hyperbolic conservation laws (HCLs) commonly arise as mathematical descriptors of the natural world, and are particularly ubiquitous in fluid dynamics. These laws appear as complicated and highly nonlinear partial differential equations describing the evolution of fundamental conserved quantities such as mass, momentum, and energy. Solving these equations analytically is entirely intractable for all but the simplest cases, and investigating problems with real world importance falls to numerical approaches more and more frequently. Most HCLs exhibit rich dynamics with complicated smooth flows and discontinuities coexisting, often with shocks arising from initially smooth data. Designing numerical schemes that can efficiently and accurately represent smooth phenomena, while also remaining robust and reliable in the vicinity of shocks, is very challenging. Finite volume methods are one particularly useful approach to designing such methods as conservation is enforced discretely, and discontinuities can be represented quite naturally. An unfortunate drawback of these methods is that achieving high-order accuracy in multiple space dimensions is difficult. This dissertation overcomes these challenges by developing a kernel-based non-polynomial

reconstruction scheme that is manifestly multidimensional. This scheme is first posed as a linear recovery problem in a reproducing kernel Hilbert space. This linear reconstruction method is then cast into a weighted essentially non-oscillatory (WENO) framework so that it may represent both smooth and discontinuous data. This scheme is then incorporated into solvers for the compressible Euler equations, compressible Navier-Stokes equations, and ideal magnetohydrodynamics (MHD) equations. In doing so, a novel set of variables that are more suited to multidimensional reconstruction, dubbed the linearized primitive variables, are introduced. Troubled cell indicators are developed that allow for a more accurate and efficient treatment of smooth solutions in an entirely automatic fashion. Positivity preserving limiters are also incorporated, and allow for the evolution of flows with extremely strong shocks. A highly parallel multi-GPU implementation is provided, and the proposed method is tested against a variety of stringent benchmark problems.

To my family, partner,
friends, and mentors.

Acknowledgments

I am grateful to my wonderful advisor Professor Dongwook Lee, without whom none of this work would have been possible. His support and guidance have been invaluable. I must also thank my masters advisors Professor Steven Ruuth and Professor Ronald Haynes who played pivotal roles in making me into the researcher I am today. I would also like to thank some of the incredible professors I have had over the years, including Pascale Garaud and Nicholas Brummell from our department here at UCSC, Ralf Wittenberg at Simon Fraser University, and Lakshmi Dasi from Colorado State University (now at the Georgia Institute of Technology). Of course, I must also thank my committee members for all of their time and effort.

My father Michael, and mother Kristin, are my original heroes. I have met few people with their breadth of knowledge, varied perspectives, and humility. My sister Jordan and her incredible work ethic is a source of real inspiration. She has been invaluable through the trials of these last few years. I would also like to thank my significant other, Jessica. Her tenacity as a researcher has kept me driven, while her love and support as a partner has kept me sane.

Finally, I would like to thank my friends near and far. I am so fortunate that you are all too numerous to list.

Chapter 1

Introduction

Systems of hyperbolic conservation laws are ubiquitous throughout the mathematical description of the natural world around us, and are particularly prominent in fluid dynamics. In d spatial dimensions these laws can be written as partial differential equations in divergence form as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathcal{F}(\mathbf{U}) = 0, \quad (1.1)$$

where $\mathbf{U} \in \mathbb{R}^n$ is a vector of n conserved quantities and \mathcal{F} is a flux tensor that is generally nonlinear with respect to \mathbf{U} . Considering a Cartesian coordinate system, $\mathbf{x} = (x_1, x_2, \dots)$, with fluxes $\mathcal{F} = (\mathbf{F}_1, \mathbf{F}_2, \dots)$, eq. (1.1) can be expanded into the associated quasilinear form

$$\frac{\partial \mathbf{U}}{\partial t} + \sum_j \mathbf{A}_j \frac{\partial \mathbf{U}}{\partial x_j} = 0, \quad \mathbf{A}_j = \frac{\partial \mathbf{F}_j}{\partial \mathbf{U}}, \quad (1.2)$$

by applying the chain rule to the flux derivatives. The conservation form in eq. (1.1) is a more natural starting point for the development of many numerical methods, including the finite volume method lying at the core of this manuscript. However, the quasilinear form in eq. (1.2) allows for more direct discussion of some of the relevant mathematical properties of these equations.

Let $\boldsymbol{\eta} \in \mathbb{R}^d$ be a unit vector and let $\mathbf{F}_{\boldsymbol{\eta}} = \sum_j \eta_j \mathbf{F}_j$ denote the flux in the $\boldsymbol{\eta}$ direction. The system in eq. (1.2) is called hyperbolic [104] if the flux Jacobian

$$\mathbf{A}_{\boldsymbol{\eta}}(\tilde{\mathbf{U}}) = \left. \frac{\partial \mathcal{F}_{\boldsymbol{\eta}}}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} = \sum_j \eta_j \left. \frac{\partial \mathbf{F}_j}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} = \sum_j \eta_j \mathbf{A}_j(\tilde{\mathbf{U}}), \quad (1.3)$$

is diagonalizable with real eigenvalues for an arbitrary unit vector $\boldsymbol{\eta} \in \mathbb{R}^d$, and all admissible states $\tilde{\mathbf{U}}$.¹ There are several interesting refinements to this baseline definition of hyperbolicity. *Strict* hyperbolicity further requires that the eigenvalues of $\mathbf{A}_{\boldsymbol{\eta}}(\tilde{\mathbf{U}})$ all be distinct. *Symmetrizable* hyperbolic systems are those whose flux Jacobians can be made symmetric through an appropriate change of variables. These systems have associated convex entropy functions which play a deep role in the well-posedness of these equations [38]. In the other direction, *weak* hyperbolicity applies when all eigenvalues are real but there is not a complete set of linearly independent eigenvectors (see e.g. [53]). This is particularly important in the consideration of non-conservative hyperbolic systems where the matrices \mathbf{A}_i in eq. (1.2) do not arise as the Jacobians of any associated flux. The analysis of these equations is significantly more complicated (see

¹From a purely mathematical perspective it may be more informative to require hyperbolicity then determine the set of admissible states as those that preserve this property.

e.g. [21, 75]).

Systems of hyperbolic conservation laws have a number of distinctive features. First, and most obviously, is that the quantities \mathbf{U} are conserved. This can be seen directly by integrating eq. (1.1) over an arbitrary fixed volume $\Omega \subset \mathbb{R}^d$ and applying the divergence theorem to obtain

$$\frac{\partial}{\partial t} \int_{\Omega} \mathbf{U} d\mathbf{x} + \int_{\partial\Omega} \mathcal{F} \cdot d\mathbf{S} = 0, \quad (1.4)$$

where $d\mathbf{S}$ is a differential surface element oriented with the outward pointing normal. Hence, the total amount of \mathbf{U} in the region Ω can only change by flows of \mathbf{U} through the boundary $\partial\Omega$. Second, information in these systems propagates at finite speeds given by the eigenvalues of the flux Jacobian, earning them the more descriptive name of wavespeeds. Finally, and perhaps most relevant to the remainder of this dissertation, is that these systems naturally support discontinuous solutions in the weak sense, and discontinuities can arise from initially smooth data in finite time.

Analytic solutions to systems of nonlinear hyperbolic conservation laws are exceedingly rare, and numerical solutions are increasingly important as a tool for investigating physical systems obeying such laws. These features together make the design of efficient and trustworthy numerical solvers a challenging, but important, task. The remainder of this introduction will discuss the finite volume method as one such approach with brief commentary on alternative approaches. Afterwards, an overview of the structure and content of the ensuing dissertation will be given.

1.1 An overview of the finite volume method

The finite volume method takes the integral form of the governing equations, as written in eq. (1.4), as the fundamental starting point for discretization. The global domain Ω where we seek to solve the equations is partitioned into a set of M disjoint cells, $\{\Omega_h\}_{h=1}^M$, such that $\Omega = \bigcup_h \Omega_h$. Integrating eq. (1.1) over each Ω_h and dividing through by the volume of the cell $|\Omega_h|$ yields

$$\frac{\partial \langle \mathbf{U} \rangle_h}{\partial t} + \frac{1}{|\Omega_h|} \int_{\partial \Omega_h} \widehat{\mathcal{F}}(\mathbf{U}^-, \mathbf{U}^+) \cdot d\mathbf{S} = 0, \quad (1.5)$$

where the cell-average value of \mathbf{U} has been notated with angle brackets indexed at h , and the flux has been replaced by a two argument flux. Here \mathbf{U}^- and \mathbf{U}^+ are the limiting values of the solution \mathbf{U} as the boundary is approached from inside and outside the cell respectively. Crucially, because the solution may be discontinuous these limiting values need not be equal. This two argument flux returns the physically relevant flux as determined from an associated Riemann problem with states \mathbf{U}^- and \mathbf{U}^+ , and serves to couple neighboring cells together.

To this point no approximations have been made and the cell averages that evolve via eq. (1.5) are identical to what would be obtained from averaging the solution of eq. (1.4), though this is of course only possible if the boundary integral could be evaluated exactly. This formulation is the foundation of the finite volume method, and arriving at a numerically tractable method primarily consists of finding an appropriate discretization of this boundary integral.

To make this discussion concrete, consider a Cartesian coordinate system in three dimensions with $\Omega = [0, L_x] \times [0, L_y] \times [0, L_z]$ partitioned into a uniform set of cubic cells with side length Δ . Identifying the tuple (i, j, k) with the single index h enumerating all cells allows the individual finite volumes to be written as $\Omega_h = [x_i \pm \Delta/2] \times [y_j \pm \Delta/2] \times [z_k \pm \Delta/2]$ where the cell center is given by $x_i = (i + \frac{1}{2}) \Delta$, $y_j = (j + \frac{1}{2}) \Delta$, and $z_k = (k + \frac{1}{2}) \Delta$. Then, eq. (1.5) takes the form

$$\begin{aligned} \frac{\partial \langle \mathbf{U} \rangle_h}{\partial t} + \frac{1}{\Delta} \left(\langle \widehat{\mathbf{F}}_1 \rangle_{(i+\frac{1}{2}, j, k)} - \langle \widehat{\mathbf{F}}_1 \rangle_{(i-\frac{1}{2}, j, k)} \right) \\ + \frac{1}{\Delta} \left(\langle \widehat{\mathbf{F}}_2 \rangle_{(i, j+\frac{1}{2}, k)} - \langle \widehat{\mathbf{F}}_2 \rangle_{(i, j-\frac{1}{2}, k)} \right) \\ + \frac{1}{\Delta} \left(\langle \widehat{\mathbf{F}}_3 \rangle_{(i, j, k+\frac{1}{2})} - \langle \widehat{\mathbf{F}}_3 \rangle_{(i, j, k-\frac{1}{2})} \right) = 0, \end{aligned} \quad (1.6)$$

where the face-average flux in the x -direction is given by

$$\langle \widehat{\mathbf{F}}_1 \rangle_{(i+\frac{1}{2}, j, k)} = \frac{1}{\Delta^2} \int_{y_j-\frac{\Delta}{2}}^{y_j+\frac{\Delta}{2}} \int_{z_k-\frac{\Delta}{2}}^{z_k+\frac{\Delta}{2}} \widehat{\mathbf{F}}_1 \left(\mathbf{U}^- \left(x_i + \frac{\Delta}{2}, y, z \right), \mathbf{U}^+ \left(x_i + \frac{\Delta}{2}, y, z \right) \right) dy dz, \quad (1.7)$$

with similar expressions for the face-averaged fluxes in the y - and z -directions. Note that angle brackets located at half-indices have now been used to denote face averages. Due to the consideration of cubic cells, these integrals can easily be discretized by a (tensor product in dimensions $d > 2$) Gauss-Legendre rule. Similar expressions hold for more general cell shapes, though of course appropriate quadrature rules will be needed. Regardless, the discretization of these surface integrals require the production of point-valued estimates of the solution over the boundary of each cell. Hence, eq. (1.6) becomes a genuine semi-discretization of eq. (1.1) if we can specify a proce-

cedure for recovering these point values of the solution \mathbf{U} given only the cell-averages $\langle \mathbf{U} \rangle_h = \langle \mathbf{U} \rangle_{(i,j,k)}$, and subsequently producing single-valued fluxes from the inevitably multi-valued traces.

The first procedure, called reconstruction, for converting cell-average data $\langle \mathbf{U} \rangle_h$ into pointwise data on cell interfaces, \mathbf{U}^\pm , lies at the core of the finite-volume method. Designing a robust, efficient, and accurate, reconstruction scheme in multiple dimensions is highly challenging and comprises the bulk of the present dissertation. The latter procedure for producing single valued fluxes is tied to the solution of an associated Riemann problem as discussed in regards to eq. (1.5), which for efficiency will typically be done through the use of an approximate Riemann solver. While interesting, this dissertation will not deliberate on these approximate Riemann solvers any more than necessary. The interested reader is referred to [104], which provides an exhaustive presentation of these solvers for the compressible Euler equations among several others.

1.1.1 Weighted Essentially Non-Oscillatory (WENO) methods in one dimension

One dimensional reconstruction schemes are naturally much simpler to implement and understand, and hence a discussion of this case will provide useful context for the remaining text. As mentioned above, hyperbolic conservation laws generally support

discontinuous solutions, and it is crucial that the developed reconstruction scheme accounts for this possibility. Godunov's theorem [46] states that no stable (monotone) linear scheme can exist that also achieves greater than first-order accuracy. Here, linearity does not imply that the reconstructed functions are linear, but rather that the reconstructed values on the cell interfaces depend linearly on the nearby data. Notably, this includes fitting polynomials locally to a given set of cell-average data.

The baseline notion behind WENO schemes is to simultaneously achieve high-order accuracy (potentially of arbitrary order given some minor assumptions) and stability by sacrificing linearity of the reconstruction procedure. In fact, as the major innovation of WENO methods lies in the robust approximation of functions with mixed regularity they have also seen use in applications having nothing to do with PDEs or Godunov's theorem [96, 81]. WENO methods have surged in popularity (see again the references within [96]), so this section will only concern the fundamentals of WENO methods rather than the full gamut of variations thereof.

These methods take a particularly simple approach to nonlinear reconstruction. The fundamental idea is to consider several candidate linear reconstruction schemes, and then form a combination of these in a data-dependent fashion. The simpler and older case of essentially non-oscillatory (ENO) methods select one such reconstruction from the candidates with the selection procedure introducing the necessary nonlinearity [51, 49]. As in [71, 59, 96], consider a set of equispaced cells in one dimension with

spacing Δx and cell-averages $\langle \mathbf{U} \rangle_i$. Consider the stencil of values

$\{\langle \mathbf{U} \rangle_{i-2}, \langle \mathbf{U} \rangle_{i-1}, \langle \mathbf{U} \rangle_i, \langle \mathbf{U} \rangle_{i+1}, \langle \mathbf{U} \rangle_{i+2}\}$. One could consider three different approximations of the point value $\mathbf{U}_{i+1/2}$ between two cells as

$$\mathbf{U}_{i+1/2}^{(1)} = \frac{1}{3}\langle \mathbf{U} \rangle_{i-2} - \frac{7}{6}\langle \mathbf{U} \rangle_{i-1} + \frac{11}{6}\langle \mathbf{U} \rangle_i \quad (1.8)$$

$$\mathbf{U}_{i+1/2}^{(2)} = -\frac{1}{6}\langle \mathbf{U} \rangle_{i-1} + \frac{5}{6}\langle \mathbf{U} \rangle_i + \frac{1}{3}\langle \mathbf{U} \rangle_{i+1} \quad (1.9)$$

$$\mathbf{U}_{i+1/2}^{(3)} = \frac{1}{3}\langle \mathbf{U} \rangle_i + \frac{5}{6}\langle \mathbf{U} \rangle_{i+1} - \frac{1}{6}\langle \mathbf{U} \rangle_{i+2} \quad (1.10)$$

each obtained by fitting a quadratic polynomial to the associated cell-averages. For smooth data these each provide 3rd-order accurate approximations. Alternatively, one could use all available data to construct a single quartic polynomial. Evaluating this yields the 5th-order approximation

$$\mathbf{U}_{i+1/2}^{(4)} = \frac{1}{30}\langle \mathbf{U} \rangle_{i-2} - \frac{13}{60}\langle \mathbf{U} \rangle_{i-1} + \frac{47}{60}\langle \mathbf{U} \rangle_i + \frac{9}{20}\langle \mathbf{U} \rangle_{i+1} - \frac{1}{20}\langle \mathbf{U} \rangle_{i+2}. \quad (1.11)$$

The 5th-order approximation can be recovered from the three 3rd-order approximations as

$$\mathbf{U}_{i+1/2}^{(4)} = \gamma_1 \mathbf{U}_{i+1/2}^{(1)} + \gamma_2 \mathbf{U}_{i+1/2}^{(2)} + \gamma_3 \mathbf{U}_{i+1/2}^{(3)}, \quad (1.12)$$

where the *linear weights* are

$$\gamma_1 = \frac{1}{10}, \quad \gamma_2 = \frac{3}{5}, \quad \gamma_3 = \frac{3}{10}. \quad (1.13)$$

Finally, these linear approximations are combined as

$$\mathbf{U}_{i+1/2} = \omega_1 \mathbf{U}_{i+1/2}^{(1)} + \omega_2 \mathbf{U}_{i+1/2}^{(2)} + \omega_3 \mathbf{U}_{i+1/2}^{(3)}, \quad (1.14)$$

where the *nonlinear weights* ω_k are now chosen in a data-dependent fashion. For smooth data each nonlinear weight should approach its linear counterpart so that eq. (1.12) approaches eq. (1.14), and the scheme as a whole tends towards the 5th-order approximation. Alternatively, when one of the substencils contains rough data its nonlinear weight should tend towards zero to remove its contribution, which suppresses any Gibbs' type oscillations hence providing stability.

It is the calculation of these nonlinear weights that has enjoyed the greatest amount of community effort and exploration. All schemes rely on some measure of smoothness of the data contained within a substencil, and then form the nonlinear weights by combining this information with the linear weights. The earliest scheme was introduced by Jiang and Shu (WENO-JS) [59]. The WENO-Z [7, 13] and mapped WENO [54, 32, 33, 111, 69, 70] methods modified these weights to improve the resolution near critical points. Other directions considered adding in information from the full stencil, such as WENO-AO [4]. Of most relevance to the present article are the kernel-based methods of GP-WENO [82, 83], RBF-CWENO [55], and the optimal recovery finite volume method from [98].

1.2 The content and structure of this dissertation

The remainder of this dissertation is split into three main chapters followed by a conclusion with possible directions for future studies.

Chapter 2: Linear and nonlinear kernel methods for function reconstruction

Based on: I. May and D. Lee [73]

Chapter 2 discusses kernel-based methods in the abstract. The reconstruction problem is posed as a recovery problem in a certain type of function space called a reproducing kernel Hilbert space (RKHS). These spaces are defined and the relevant properties discussed. Symmetric and asymmetric formulations for reconstruction are presented and the inclusion of additional polynomial subspaces is deliberated upon. Ultimately this chapter builds to the development of a novel approach for nonlinear recovery in these spaces yielding a fully multidimensional kernel-based weighted non-oscillatory method, referred to hereafter as KFVM-WENO.

Chapter 3: A kernel-based finite volume method for the compressible Euler equations

Based on: I. May and D. Lee [73]

Chapter 3 discusses the compressible Euler equations and compressible Navier-Stokes equations. The preceding KFVM-WENO method is formulated for scalar fields and this chapter will discuss its adaptation to systems of conservation laws. Transformations of variables for the sake of reconstruction are discussed including a newly proposed approach using the so-called linearized primitive variables. A troubled cell indicator, adapted from the KXRCF indicator [63], is presented and allows for the effective, but computationally demanding, WENO process to be sidestepped. Positivity preservation, as necessary for the treatment of flows with extremely strong shocks, is

incorporated into the method. A brief digression is made to discuss the time integration methods chosen for this work, and the chapter is concluded with variety of illustrative numerical results.

Chapter 4: A kernel-based finite volume method for ideal magnetohydrodynamics

Chapter 4 extends this machinery to the equations of ideal magnetohydrodynamics (MHD). Special attention is paid to the solenoidal constraint on the magnetic field, leading ultimately to the selection of a method utilizing generalized Lagrange multipliers (GLM). Reconstruction variables, troubled cell indicators, and positivity preservation are again discussed with the relevant adaptations from the former chapter to the MHD case. The choice to use a GLM style method requires an accurate treatment of relatively complicated source terms, the discretization of which is discussed. The software written to support the research presented in this dissertation is overviewed. Finally, a set of numerical results are presented to showcase the capabilities of the proposed scheme.

Chapter 2

Linear and nonlinear kernel methods for function reconstruction

This chapter is based on the paper [73] submitted to the *Astrophysical Journal*.

2.1 Introduction

As discussed in Chapter 1, the ability to reconstruct point values of the solution from a discrete set of cell-averages lies at the core of the finite volume method. Moreover, for systems supporting discontinuous solutions the only way to achieve greater than first order accuracy while maintaining a stable scheme is to make this reconstruction procedure nonlinear with respect to the data[46]. Reconstruction in one spatial dimension is mostly straightforward; the cell-averages can easily be fit with polynomi-

als. The nonlinear limiting necessary to obtain stable schemes is now highly advanced in the one-dimensional case, as evidenced by the plethora of schemes mentioned at the close of Section 1.1.1.

Multidimensional reconstruction is significantly more challenging than its one dimensional counterpart. Fitting data scattered in multiple dimensions with polynomials requires that the corresponding multivariate polynomial space be unisolvent with respect to the layout of the known data [9, 81]. To ameliorate this one could seek to fit lower degree polynomials in a least squares sense, weakening the dependence on data layout but also making non-optimal use of all known information. Alternatively, tensor-product representations could be used [5, 90], but these again utilize more data than necessary to reach a given accuracy. Furthermore this approach strongly limits the geometries that can be treated, and thus provides little benefit over simpler finite difference methods. As an aside, the tensor-product formulation does provide a great deal of structure to the function approximation problem which can be leveraged to create very efficient numerical methods despite the use of more information than strictly necessary; this is particularly useful in the design of discontinuous Galerkin methods [43]. All of these issues are strongly exacerbated in the context of nonlinear reconstruction.

In this chapter a rather different approach is taken. Instead of reconstructing functions over multivariate polynomial spaces. the approximation problem will be posed in *reproducing kernel Hilbert spaces* (RKHSs). This approach side steps many of the

issues inherent to polynomial approximation, and furthermore allows a unified formulation of the reconstruction process that is largely independent of the spatial dimension. These methods have a long history in function approximation [89, 112, 37, 28], statistics and machine learning [80], theory of partial differential equations [76], numerical methods for partial differential equations [92, 29, 36] (particularly elliptic equations), and undoubtedly many other fields.

This chapter begins by introducing the core concepts and properties of kernel-based methods and RKHSs. This first section is necessarily more abstract than the subsequent ones. The linear interpolation and reconstruction problems are posed in this framework in symmetric and asymmetric forms in Sections 2.3 and 2.4 respectively. The sections beyond this build to a fully nonlinear reconstruction technique that generalizes WENO methods to multidimensional non-polynomial techniques. This chapter will only consider the reconstruction of scalar valued data, and the appropriate extensions to multicomponent data will be made in subsequent chapters.

2.2 Reproducing kernel Hilbert spaces and optimal recovery

Let the domain $\Omega \subset \mathbb{R}^d$ be open and not necessarily bounded, and let \mathcal{H}_K be a Hilbert space, with inner product $(\cdot, \cdot)_{\mathcal{H}_K}$, consisting of functions whose domain is Ω .

Crucially, \mathcal{H}_K is properly a space of functions, and not of equivalence classes of functions, precluding for instance Lebesgue spaces like $L_2(\Omega)$. In such spaces, point evaluation can be cast as a functional $\delta_{\mathbf{x}}$ for all $\mathbf{x} \in \Omega$ such that $\delta_{\mathbf{x}}f \rightarrow f(\mathbf{x})$ for all $f \in \mathcal{H}_K$. If these point evaluation functionals are linear and continuous for all $\mathbf{x} \in \Omega$, then $\delta_{\mathbf{x}} \in \mathcal{H}_K^*$ where \mathcal{H}_K^* denotes the continuous dual of \mathcal{H}_K , and \mathcal{H}_K is called a *reproducing kernel Hilbert space* (RKHS). Furthermore, by the Riesz representation theorem each evaluation functional $\delta_{\mathbf{x}} \in \mathcal{H}_K^*$ has an associated representative $K_{\mathbf{x}} \in \mathcal{H}_K$ such that

$$\delta_{\mathbf{x}}f = (K_{\mathbf{x}}, f)_{\mathcal{H}_K} = f(\mathbf{x}), \quad \forall f \in \mathcal{H}_K, \forall \mathbf{x} \in \Omega. \quad (2.1)$$

A symmetric function $K : \Omega \times \Omega \rightarrow \mathbb{R}$ is called a positive definite kernel if

$$\sum_{i,j} c_i c_j K(\mathbf{x}_i, \mathbf{x}_j) \geq 0, \quad (2.2)$$

for all distinct sets of points $\mathbf{x}_1, \dots, \mathbf{x}_n \in \Omega$ and (nonzero) coefficients $c_1, \dots, c_n \in \mathbb{R}$.

The squared exponential kernel

$$K(\mathbf{x}, \mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\ell^2}}, \quad (2.3)$$

is one such example. Noting the presence of the non-strict inequality in eq. (2.2) one can see that positive definite is a slight misnomer, and perhaps that positive semi-definite kernel would be more apt nomenclature. The former is more prevalent in the broader literature, mostly for historical reasons [76].

Interestingly, for RKHSs the \mathcal{H}_K representatives of point evaluation functionals always correspond to such a kernel function in that $K_{\mathbf{x}}(\cdot) = K(\mathbf{x}, \cdot)$. In fact, for any linear

functional $\mu^{(\mathbf{x})} \in \mathcal{H}_K^*$ the function in \mathcal{H}_K representing it will be $\mu^{(\mathbf{x})}K(\mathbf{x}, \cdot)$, and for any pair of linear functionals $\mu^{(\mathbf{x})}, \lambda^{(\mathbf{x})} \in \mathcal{H}_K^*$ the induced \mathcal{H}_K^* inner product can be evaluated as

$$\left(\mu^{(\mathbf{x})}, \lambda^{(\mathbf{x})}\right)_{\mathcal{H}_K^*} = \left(\mu^{(\mathbf{x})}K(\mathbf{x}, \cdot), \lambda^{(\mathbf{x})}K(\mathbf{x}, \cdot)\right)_{\mathcal{H}_K} = \mu^{(\mathbf{x})}\lambda^{(\mathbf{y})}K(\mathbf{x}, \mathbf{y}). \quad (2.4)$$

The superscripts on each functional are present to indicate which variable the functional is being applied with respect to, and temporary dummy variables may be inserted as needed. There are two fundamental results from functional analysis regarding these spaces: (i) every RKHS has a unique¹ reproducing kernel function, though its precise form may be unknown [76, 87], and (ii) every symmetric positive definite kernel function generates an associated RKHS. This second result is the content of the celebrated Moore-Aronszajn theorem [76].

This machinery is rather abstract, but lays a solid foundation for function approximation in multiple dimensions. Before formulating the interpolation and reconstruction problems concretely, it is worth deliberating on a result from multivariate interpolation as a whole. An implication of the Mairhuber-Curtis theorem [29] is that interpolation over a scattered set of points in \mathbb{R}^d with respect to an n -dimensional linear space of functions is only generically possible if either d or n is equal to 1 (or trivially both $d = n = 1$). This is the theoretic underpinning of the earlier discussion on the difficulties inherent to interpolation by multivariate polynomials. The essential failure in the

¹This uniqueness holds up to choice of norm. Multiple RKHSs may contain the same set of functions, but have different norms and hence be different spaces with different associated kernel functions.

multivariate polynomial case is that in dimensions $d > 1$ one can always find sets of distinct points $\mathbf{x}_1, \dots, \mathbf{x}_n \in \Omega$ which are roots of some polynomial in the space. Hence interpolation is not uniquely defined as arbitrary multiples of this polynomial could be added to any interpolant. As will be seen in the next two sections, kernel-based interpolation/reconstruction avoids this issue in a rather unique fashion.

2.3 Symmetric kernel approximation

The selection of a kernel function K , such as the squared exponential kernel in eq. (2.3), implicitly selects a RKHS \mathcal{H}_K where function approximation can be done [76, 87]. The interpolation and reconstruction problems can be naturally cast as optimization problems in this RKHS. The interpolation problem is simpler to formulate and builds useful intuition for the slightly more complicated reconstruction problem, hence it is considered first despite not being useful in and of itself for the finite volume method presently being investigated. The process of building a function in \mathcal{H}_K to match some given data is then generalized and discussed in the context of optimal recovery.

2.3.1 Symmetric kernel-based interpolation

Consider a finite set of distinct points, $\mathbf{x}_1, \dots, \mathbf{x}_n \in \Omega$, and collocated real valued data $g_1, \dots, g_n \in \mathbb{R}$. A function $f \in \mathcal{H}_k$ is an interpolant for this data if

$$\delta_{\mathbf{x}_h} f = g_h, \quad \forall h = 1, \dots, n, \quad (2.5)$$

holds. There are infinitely many such interpolants, but one privileged interpolant, \tilde{f} , can be found by minimizing the \mathcal{H}_K -norm²

$$\tilde{f} = \underset{\substack{\delta_{\mathbf{x}_h} f = g_h \\ h=1,n}}{\operatorname{argmin}} \|f\|_{\mathcal{H}_K}. \quad (2.6)$$

Recall that on \mathcal{H}_K the evaluation functionals $\delta_{\mathbf{x}}$ are linear and continuous, and hence also bounded. This means that the values a function can take are bounded as

$$|\tilde{f}(\mathbf{x})| = |\delta_{\mathbf{x}} \tilde{f}| \leq C_K \|\tilde{f}\|_{\mathcal{H}_K}, \quad (2.7)$$

where C_K is a constant dependent on the kernel function and the set of points $\mathbf{x}_1, \dots, \mathbf{x}_n$, but independent of \tilde{f} . Minimizing the norm of the interpolant thus constrains the behavior of the interpolant between data locations. This is qualitatively similar to seeking the minimum degree interpolant in univariate polynomial fitting; one could find infinitely many polynomial interpolants for a given problem, but the most natural choice is the unique interpolant of minimal degree.

Finding \tilde{f} requires solving an optimization problem posed over an infinite dimensional space, which at first glance may seem like a challenging problem. Fortunately this optimization problem is easy to solve. The representer theorem [76] asserts that \tilde{f} lies in the subspace $H = \operatorname{span}\{K(\cdot, \mathbf{x}_l) | l = 1, \dots, n\} \subset \mathcal{H}_K$, hence we seek an interpolant of the form

$$\tilde{f}(\mathbf{x}) = \sum_{l=1}^n a_l K(\mathbf{x}, \mathbf{x}_l). \quad (2.8)$$

²This norm is associated to the \mathcal{H}_K inner product in the usual fashion with $\|f\|_{\mathcal{H}_K}^2 := (f, f)_{\mathcal{H}_K}$.

Enforcing the interpolation conditions, $\delta_{\mathbf{x}_h} \tilde{f} = g_h$, and gathering the coefficients and known data into the vectors \mathbf{a} and \mathbf{g} respectively, yields the linear system of equations

$$\mathbf{K}\mathbf{a} = \mathbf{g}, \quad (2.9)$$

where the entries of the matrix are $K_{hl} = K(\mathbf{x}_h, \mathbf{x}_l)$. By eq. (2.2) we can immediately see that \mathbf{K} is symmetric positive definite, and hence invertible, for any strictly positive definite kernel function $K(\mathbf{x}, \mathbf{y})$. This means that the interpolation problem is always well-posed, and we can now appreciate how kernel-based methods avoid the implications of the Mairhuber-Curtis theorem. Crucially, the basis functions used in writing the interpolant, $K(\cdot, \mathbf{x}_l)$, are intrinsically tied to the points where the data lives, and hence there is not a single fixed finite dimensional space of functions used for all interpolation problems. This can be contrasted with the situation for multivariate polynomial interpolation which is only well-posed for particular sets of points. In the kernel-based approach the space of functions where interpolants are sought is always adapted to the set of points being used.

2.3.2 Optimal recovery

The minimum norm interpolant discussed above, and its particular form given in eq. (2.8) via the representer theorem, is a specific construction from the broader topic of optimal recovery [55, 89, 98, 29]. The more general optimal recovery framework will provide a more natural setting for the reconstruction problem. Consider a set of linearly

independent functionals $\{\lambda_l^{(\mathbf{x})}\}_{l=1}^n$, where as above the superscript (\mathbf{x}) is present to indicate which variable the functional is being applied to. Let $f \in \mathcal{H}_K$ be a function for which the values $\lambda_l^{(\mathbf{x})} f$ are known. A quasi-interpolant $S_f \in \mathcal{H}_K$ for f can then be written as

$$S_f(\mathbf{x}) = \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(\mathbf{x})} f(\mathbf{x}), \quad (2.10)$$

where the basis functions $v_l \in \mathcal{H}_K$ are linearly independent, but otherwise arbitrary. The process of constructing this quasi-interpolant is called linear recovery, meaning that the recovery of S_f from f respects the linear structure³ of \mathcal{H}_K . In particular, if these basis functions are cardinal with respect to the data functionals

$$\lambda_h^{(\mathbf{x})} v_l(\mathbf{x}) = \delta_{hl} = \begin{cases} 1, & h = l \\ 0, & h \neq l \end{cases}, \quad (2.11)$$

then eq. (2.10) will be a proper interpolant. Since $S_f \in \mathcal{H}_K$ the difference $(f - S_f) \in \mathcal{H}_K$, and the pointwise error at any $\mathbf{x} \in \Omega$ can be written as

$$f(\mathbf{x}) - S_f(\mathbf{x}) = \delta_{\mathbf{x}} f - \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(\mathbf{x})} f \quad (2.12)$$

$$= \left(\delta_{\mathbf{x}} - \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(\mathbf{x})} \right) f \quad (2.13)$$

$$= \varepsilon_{\mathbf{x}} f \quad (2.14)$$

where the parenthetical term in the second line has been identified with the error functional $\varepsilon_{\mathbf{x}}$. Evaluating $\varepsilon_{\mathbf{x}} f$ is generally impossible as that would require perfect knowl-

³E.g. for $f, g \in \mathcal{H}_K$ and $a, b \in \mathbb{R}$ one finds that $S_{(af+bg)} = aS_f + bS_g$.

edge of f , hence nullifying the need for a (quasi-)interpolant.

However, just as the minimum norm interpolant was argued for above, one can instead seek the norm of this functional and use that to bound the possible error accrued by the linear recovery process that builds the (quasi-)interpolant in eq. (2.10). The norm of the error functional is dependent on the evaluation point \mathbf{x} and the set of functionals $\{\lambda_l^{(\mathbf{x})}\}_{l=1}^n$, but independent of the function f [87]. This information is embodied by the *power function* defined as $P(\mathbf{x}) = \|\epsilon_{\mathbf{x}}\|_{\mathcal{H}_K}$, though note that generally $P \notin \mathcal{H}_K$. This norm can be evaluated directly as

$$\begin{aligned}
P^2(\mathbf{x}) &= (\epsilon_{\mathbf{x}}, \epsilon_{\mathbf{x}})_{\mathcal{H}_K^*} & (2.15) \\
&= (\delta_{\mathbf{x}}, \delta_{\mathbf{x}})_{\mathcal{H}_K^*} - 2 \left(\delta_{\mathbf{x}}, \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(\mathbf{x})} \right)_{\mathcal{H}_K^*} + \left(\sum_{h=1}^n v_h(\mathbf{x}) \lambda_h^{(\mathbf{x})}, \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(\mathbf{x})} \right)_{\mathcal{H}_K^*} \\
&= K(\mathbf{x}, \mathbf{x}) - 2 \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(\mathbf{y})} K(\mathbf{x}, \mathbf{y}) + \sum_{h=1}^n \sum_{l=1}^n v_h(\mathbf{x}) v_l(\mathbf{x}) \lambda_h^{(\mathbf{y})} \lambda_l^{(\mathbf{z})} K(\mathbf{y}, \mathbf{z}), & (2.16)
\end{aligned}$$

where in passing from the second to the third line the dual space inner products have been evaluated via eq. (2.4). Finally one obtains the error bound

$$|f(\mathbf{x}) - S_f(\mathbf{x})| = |\epsilon_{\mathbf{x}} f| \leq P(\mathbf{x}) \|f\|_{\mathcal{H}_K}. \quad (2.17)$$

This discussion of error functionals and the associated power function holds for any linear recovery process, including ones that are not kernel-based [88]. A natural question is whether a privileged set of basis functions $\{v_l\}_{l=1}^n$ can be found that minimizes the power function, and hence minimizes the pointwise error of S_f , for a given set of data functionals $\{\lambda_l^{(\mathbf{x})}\}_{l=1}^n$. Fortunately, the quadratic form $P^2(\mathbf{x})$ in eq. (2.16)

can be minimized with respect to the basis functions [89, 88], and yields two key facts. First the recovery process must be interpolatory, so eq. (2.11) must hold. Second, all of the basis functions can be written as linear combinations of functions of the form $\lambda_l^{(y)} K(\mathbf{x}, \mathbf{y})$. This second fact implies that $S_f(\mathbf{x})$, via equation eq. (2.10), is also a linear combination of such functions and hence one may take

$$S_f(\mathbf{x}) = \sum_{l=1}^n a_l \lambda_l^{(y)} K(\mathbf{x}, \mathbf{y}), \quad (2.18)$$

as the starting form for the recovered function. Letting $g_h = \lambda_h^{(x)} f(\mathbf{x})$ denote the given data, and enforcing that $\lambda_h^{(x)} S_f(\mathbf{x}) = g_h$ again yields a linear system of equations for the coefficients \mathbf{a}

$$\mathbf{C}\mathbf{a} = \mathbf{y}, \quad (2.19)$$

where now the entries of the matrix are $C_{hl} = \lambda_h^{(x)} \lambda_l^{(y)} K(\mathbf{x}, \mathbf{y})$. Note that this reduces to eq. (2.9) when the data functionals $\{\lambda_l^{(x)}\}_{l=1}^n$ are all point evaluations.

Interpolatory processes, those with $\lambda_h^{(x)} v_l(\mathbf{x}) = \delta_{hl}$, allow for several useful simplifications. The double sum in eq. (2.16) can be absorbed into the former single sum yielding

$$P^2(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}) - \sum_{l=1}^n v_l(\mathbf{x}) \lambda_l^{(y)} K(\mathbf{x}, \mathbf{y}). \quad (2.20)$$

Furthermore, by letting \mathbf{E} be a vector with entries $E_l = \lambda_l^{(y)} K(\mathbf{x}, \mathbf{y})$ the power function can be written more compactly as

$$P^2(\mathbf{x}) = K(\mathbf{x}, \mathbf{x}) - \mathbf{E}^T \mathbf{C}^{-1} \mathbf{E}. \quad (2.21)$$

This is obtained from eq. (2.20) by expanding the cardinal functions $v_l(\mathbf{x})$ in the standard basis of functions $\lambda_l^{(y)}K(\mathbf{x}, \mathbf{y})$ and recognizing that the expansion coefficients lie in the associated columns of \mathbf{C}^{-1} . Finally, since all given data are matched exactly the interpolant of the residual satisfies $S_{(f-S_f)} = 0$, and the bound in eq. (2.17) can be iterated to obtain

$$|f(\mathbf{x}) - S_f(\mathbf{x})| \leq P(\mathbf{x}) \|f - S_f\|_{\mathcal{H}_K}. \quad (2.22)$$

Moreover, the residual $(f - S_f)$ is \mathcal{H}_K -orthogonal to the subspace $H = \text{span} \left\{ \lambda_l^{(y)}K(\mathbf{x}, \mathbf{y}) \right\}$, bearing striking similarity to Galerkin methods [43]. This implies that $\|f - S_f\|_{\mathcal{H}_K} < \|f\|_{\mathcal{H}_K}$ holds, and that the bound in eq. (2.22) is tighter than that in eq. (2.17). In fact, the notion of power functions is applicable to more than just the treatment of kernel-based methods as discussed here, and can be formulated for many other recovery processes including standard polynomial methods [88].

This discussion is very general and applies to any set of linearly independent functionals describing the known data for some function $f \in \mathcal{H}_K$. More practically these functionals will take one of a few forms, and the recovery process will be referred to by special names to reflect this. If all of the data functionals are point evaluations then this process is simply called interpolation as discussed in Section 2.3.1. If these functionals additionally consist of point evaluations of derivatives this process is called Hermite-Birkhoff interpolation. In the case relevant to finite volume methods the data

functionals correspond to cell-averaging

$$\lambda_h^{(\mathbf{x})} = \frac{1}{|\Omega_h|} \int_{\Omega_h} \cdot d\mathbf{x}, \quad (2.23)$$

and the process is called reconstruction. This is the only case to be considered in the remainder of this dissertation, and hereafter eq. (2.23) is taken as the definition for each $\lambda_h^{(\mathbf{x})}$ over the set of cells $\{\Omega_h\}_{h=1}^M$.

2.3.3 Inclusion of polynomials

The use of kernel-based methods for multidimensional interpolation and reconstruction was motivated primarily by the deficiencies of multivariate polynomials. However, the existence of these deficiencies does not imply that multivariate polynomials have no utility in multidimensional interpolation and reconstruction. Consider adding a polynomial tail to the sought reconstruction function in eq. (2.18) yielding the form

$$S_f(\mathbf{x}) = \sum_{l=1}^n a_l \lambda_l^{(\mathbf{y})} K(\mathbf{x}, \mathbf{y}) + \sum_{|\boldsymbol{\alpha}_{(v)}| \leq p} b_v \mathbf{x}^{\boldsymbol{\alpha}_{(v)}}. \quad (2.24)$$

Here $\boldsymbol{\alpha}_{(v)} \in \mathbb{N}_0^d$ is a multiindex, where $|\boldsymbol{\alpha}| = \sum_i \alpha_i$, and exponentiation is applied componentwise such that $\mathbf{x}^{\boldsymbol{\alpha}} = \prod_i x_i^{\alpha_i}$. In fact there are kernels in which this tail is required for the interpolation and reconstruction processes to be well posed, which are the so-called conditionally positive definite kernels [87], though these will not be considered any further here.

Enforcing that eq. (2.24) match the known data now results in an under-determined

system of equations for the vectors of coefficients \mathbf{a} and \mathbf{b} given by

$$\mathbf{C}\mathbf{a} + \mathbf{P}\mathbf{b} = \mathbf{g} \quad (2.25)$$

$$C_{hl} = \lambda_h^{(\mathbf{x})} \lambda_l^{(\mathbf{y})} K(\mathbf{x}, \mathbf{y}) \quad (2.26)$$

$$P_{hv} = \lambda_h^{(\mathbf{x})} \mathbf{x}^{\alpha_{(v)}}. \quad (2.27)$$

By including this polynomial tail the reconstruction process can be made exact for all polynomials $p(\mathbf{x}) \in \mathbb{P}_p^d$, that is, for all d -variate polynomials of total degree less than or equal to p . To force the reconstruction process to be exact for these polynomials the coefficients \mathbf{a} associated with the kernel part of eq. (2.24) must be set to zero whenever the given data \mathbf{g} can be described by the polynomial part. More specifically, \mathbf{a} must be zero whenever \mathbf{g} lies in the column space of \mathbf{P} , hence the condition $\mathbf{P}^T \mathbf{a} = \mathbf{0}$ can be added yielding the following block linear system for the vectors of coefficients \mathbf{a} and \mathbf{b}

$$\begin{bmatrix} \mathbf{C} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix} = \begin{pmatrix} \mathbf{g} \\ \mathbf{0} \end{pmatrix}. \quad (2.28)$$

For the relatively small systems of interest in the subsequent sections it will be easiest numerically to treat eq. (2.28), or its asymmetric version in eq. (2.32), as one large system ignoring the block structure. On the other hand, interpreting the behavior of this reconstruction scheme is well served by examining this block structure more closely.

Consider solving eq. (2.28) block-wise where the Schur complement first gives the

polynomial coefficients, \mathbf{b} , as the solution of

$$\left(\mathbf{P}^T \mathbf{C}^{-1} \mathbf{P}\right) \mathbf{b} = \mathbf{P}^T \mathbf{C}^{-1} \mathbf{g}, \quad (2.29)$$

and subsequently the kernel coefficients, \mathbf{a} , as the solution of

$$\mathbf{C} \mathbf{a} = \mathbf{g} - \mathbf{P} \mathbf{b}. \quad (2.30)$$

Ignoring the kernel-based nature of the reconstruction scheme for a moment, consider the case where $\mathbf{C} = \mathbf{I}$. Under this restriction eq. (2.29) simply reduces to the normal equations for standard least-squares fitting. Slightly more generally one could consider replacing \mathbf{C} by some diagonal matrix with positive entries, in which case eq. (2.29) would reduce to weighted least-squares.

This yields a useful interpretation of this reconstruction process. First, a generalized least-squares problem is solved with respect to the (generally full) weight matrix \mathbf{C} . Second, regular kernel-based reconstruction is applied to the residual $(\mathbf{g} - \mathbf{P} \mathbf{b})$ in eq. (2.30). Together these components yield a reconstructed function that matches the supplied data exactly.

2.4 Asymmetric kernel approximation

The symmetric reconstruction procedure described in the preceding sections has an elegant and well understood theory. Unfortunately this approach is somewhat unwieldy due to the application of the cell-averaging functionals $\lambda_l^{(\mathbf{x})}$ to the kernel function within

the approximant in eq. (2.24). This process can be simplified by seeking an approximant in the form

$$S_f(\mathbf{x}) = \sum_{l=1}^n a_l K(\mathbf{x}, \mathbf{x}_l) + \sum_{|\boldsymbol{\alpha}_{(v)}| \leq p} b_v \mathbf{x}^{\boldsymbol{\alpha}_{(v)}}. \quad (2.31)$$

Enforcing that this match the given, cell-averaged, data now yields the block linear system

$$\begin{bmatrix} \mathbf{Q} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{pmatrix} \mathbf{a} \\ \mathbf{b} \end{pmatrix} = \begin{pmatrix} \mathbf{g} \\ \mathbf{0} \end{pmatrix}, \quad (2.32)$$

where the matrix \mathbf{Q} has entries $Q_{hl} = \lambda_h^{(\mathbf{x})} K(\mathbf{x}, \mathbf{x}_l)$ and may no longer be symmetric since generally $\lambda_h^{(\mathbf{x})} K(\mathbf{x}, \mathbf{x}_l) \neq \lambda_l^{(\mathbf{x})} K(\mathbf{x}, \mathbf{x}_h)$. Note that polynomial exactness is enforced in precisely the same way as before. This simplification is beneficial in that integrating the kernel function can be challenging, and may require the use of numerical quadrature. In particular, the entries $C_{hl} = \lambda_h^{(\mathbf{x})} \lambda_l^{(\mathbf{y})} K(\mathbf{x}, \mathbf{y})$ involve integrals in $\Omega \times \Omega \subset \mathbb{R}^{(d+d)}$. These can be analytically computed for the squared exponential kernel in eq. (2.3) and in the $d = 1$ case are already quite complicated [82]. This kernel has the remarkable property of being separable, and for simple cell shapes (rectangles in $d = 2$ and cuboids in $d = 3$) the higher dimensional integrals can be written as products of integrals in $\mathbb{R}^{(1+1)}$. On the other hand, many other kernels of interest are not separable, and worse yet may not be integrable in closed form. In these cases one would need to employ quadrature rules in $\mathbb{R}^{(d+d)}$ which is quite cumbersome.

This simplification can be further appreciated by considering the evaluation of the

reconstructed function and its derivatives. Evaluation of eq. (2.31) at the point \mathbf{x}_* , given that the coefficient vectors have already been obtained from eq. (2.32), can be written as

$$S_f(\mathbf{x}_*) = \mathbf{T}^T \mathbf{a} + \mathbf{S}^T \mathbf{b} \quad (2.33)$$

$$T_l = K(\mathbf{x}_*, \mathbf{x}_l) \quad (2.34)$$

$$S_v = \mathbf{x}_*^{\boldsymbol{\alpha}^{(v)}}, \quad (2.35)$$

where \mathbf{T} and \mathbf{S} will be called sample vectors⁴, which notably do not include the cell-averaging functionals. Moreover, the entries of \mathbf{T} and \mathbf{S} could be replaced with derivatives of the kernel function and monomials to obtain derivatives of the reconstructed function.

As a final refinement it is beneficial to observe that reconstructing a point value from given data consists of computing

$$S_f(\mathbf{x}_*) = \begin{pmatrix} \mathbf{T} \\ \mathbf{S} \end{pmatrix}^T \begin{bmatrix} \mathbf{Q} & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix}^{-1} \begin{pmatrix} \mathbf{g} \\ \mathbf{0} \end{pmatrix}, \quad (2.36)$$

and noting that most of this process is completely independent of the data. Reorganizing this slightly to first solve

$$\begin{bmatrix} \mathbf{Q}^T & \mathbf{P} \\ \mathbf{P}^T & \mathbf{0} \end{bmatrix} \begin{pmatrix} \mathbf{r} \\ \mathbf{w} \end{pmatrix} = \begin{pmatrix} \mathbf{T} \\ \mathbf{S} \end{pmatrix}, \quad (2.37)$$

⁴This terminology is borrowed from Gaussian processes, though it is a small misnomer due to the additional polynomial terms.

then allows point evaluation to proceed as

$$S_f(\mathbf{x}_*) = \mathbf{r}^T \mathbf{y}. \quad (2.38)$$

Hence, the reconstruction vector \mathbf{r} can be computed once ahead of time, and subsequently any data can be quickly reconstructed to the point \mathbf{x}_* through a simple dot product. This recasting of the process is crucial for time-dependent problems. Note also that \mathbf{w} does not participate in evaluating the reconstructed function, and that the cost of applying eq. (2.38) is independent of the presence of the polynomial terms in eq. (2.31). The contents of \mathbf{w} are Lagrange multipliers that serve to enforce the polynomial exactness constraints, which have now been encoded into \mathbf{r} .

2.5 Nonlinear reconstruction for functions with poor regularity

All of the preceding sections concern linear recovery processes, irrespective of whether interpolation, symmetric reconstruction, or asymmetric reconstruction is being performed. In all cases the generated function matching the given data depends linearly on that data, and by extension all evaluations of that function depend linearly on the given data. However, as mentioned in Section 1.1 Godunov's theorem [44] asserts that linear processes yielding greater than first order accuracy can not be used in the treatment of generic hyperbolic conservation laws. These equations support discon-

tinuous solutions and the essential failure of these linear recovery processes is the onset of Gibbs type oscillations that feedback nonlinearly, hence ruining stability.

To achieve greater than first order accuracy one needs an inherently nonlinear means of reconstructing functions from given data. This section concerns the extension of the asymmetric reconstruction scheme to a weighted non-oscillatory (WENO) scheme that is appropriate for handling data arising from functions with poor regularity. The preceding sections have all been generic to handle mostly arbitrary layouts of cells and arbitrary evaluation points. From now the discussion will be more concrete, and concern only cells coming from uniform Cartesian grids. At the same time, most of what is presented can still be extended to unstructured grids with minimal changes to the underlying mathematical formulation⁵.

2.5.1 Stencils and substencils

The first step in making the reconstruction procedures concrete is to select sets of cells whose solution averages will be used as the given data. In the usual parlance these sets will be called stencils. WENO type methods rely also on subsets of data, and these stencils must also be broken into several substencils. This section is dedicated to stencil layout on Cartesian grids, and the designed stencils should thus exhibit the same symmetries as this grid to avoid having directional preferences. Similarly, the

⁵Though, the implementation will be much more complicated, hence the ignorance of unstructured grids within this work.

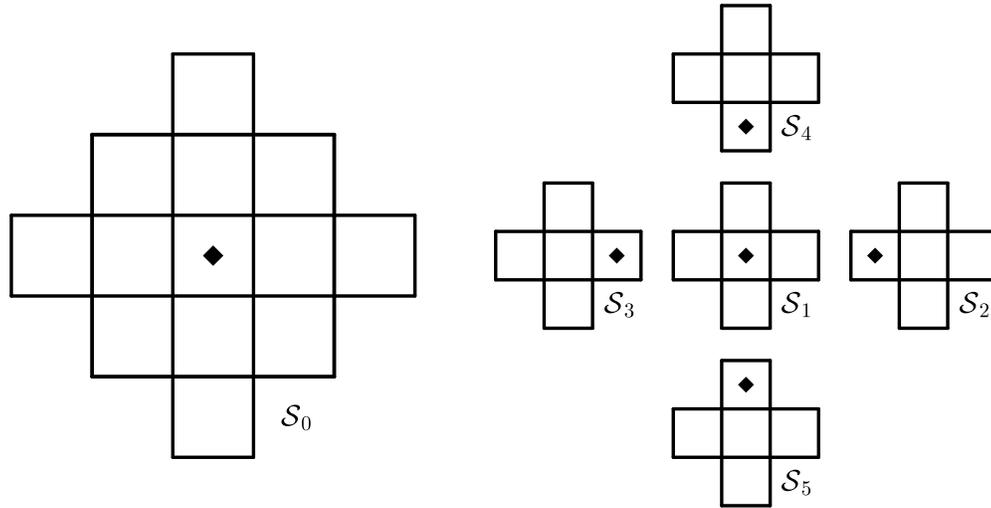


Figure 2.1: The full radius-2 stencil S_0 is shown on the left with its five substencils, $S_q, q = 1, \dots, 5$, on the right. The cell with the diamond in each (sub) stencil indicates the central cell where reconstruction is performed. The full stencil S_0 has 13 cells, while each of the substencils has five.

substencils should be arranged in a fashion that also respects this underlying symmetry.

The design of these (sub)stencils is relatively simple due to the geometric flexibility of kernel-based reconstruction methods. This work utilizes stencils that are roughly circular in two space dimensions and roughly spherical in three space dimensions. The radius of a stencil is denoted by the integer R , which in this work takes the values $R = 2$ and $R = 3$.

The stencil descriptions will make use of local grid-based indices where $(0,0)$ or $(0,0,0)$ will denote the central cell where reconstruction is being performed, and $(i, j) \in \mathbb{Z}^2$ or $(i, j, k) \in \mathbb{Z}^3$ will label cells relative to the center. The application of formulas from previous sections will naturally require the cells in each (sub)stencil to be enumerated such that they may be referred to by a single index. In two dimensions the

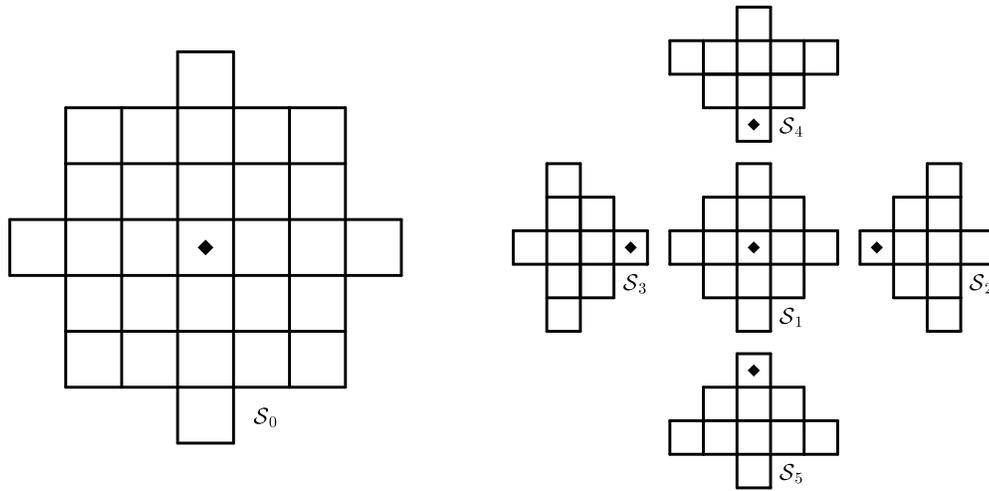


Figure 2.2: The full radius-3 stencil \mathcal{S}_0 is shown on the left with its five substencils, $\mathcal{S}_q, q = 1, \dots, 5$, on the right. The cell with the diamond in each (sub)stencil indicates the central cell where reconstruction is being performed. The full stencil \mathcal{S}_0 has 29 cells, the central substencil \mathcal{S}_1 has 13 cells, and each of the remaining biased substencils, $\mathcal{S}_2, \dots, \mathcal{S}_5$, has 10.

full stencil consists of the cells $\mathcal{S}_0 = \{(i, j) : (i^2 + j^2) \leq R^2\}$, and in three dimensions the full stencil consists of the cells $\mathcal{S}_0 = \{(i, j, k) : (i^2 + j^2 + k^2) \leq R^2\}$. The substencils, \mathcal{S}_q labeled by q , consist of one central substencil of smaller radius and $2d$ biased conical substencils aligned with the positive and negative axis directions for each dimension.

In two dimensions there are 5 substencils defined as

$$\mathcal{S}_1 = \{(i, j) \in \mathcal{S}_0 : (i^2 + j^2) \leq (R-1)^2\},$$

$$\mathcal{S}_2 = \{(i, j) \in \mathcal{S}_0 : |j| \leq i\},$$

$$\mathcal{S}_3 = \{(i, j) \in \mathcal{S}_0 : |j| \leq -i\},$$

$$\mathcal{S}_4 = \{(i, j) \in \mathcal{S}_0 : |i| \leq j\},$$

$$\mathcal{S}_5 = \{(i, j) \in \mathcal{S}_0 : |i| \leq -j\}.$$

For radius $R = 2$ the full stencil has 13 cells and all of the substencils have 5 cells. For radius $R = 3$ the full stencil has 29 cells, the central substencil has 13 cells, and the biased substencils each have 10 cells. The two dimensional radius $R = 2$ and $R = 3$ stencils and substencils can be seen in Figures 2.1 and 2.2 respectively. Similarly, in three dimensions there are 7 substencils defined as

$$\mathcal{S}_1 = \{(i, j, k) \in \mathcal{S}_0 : (i^2 + j^2 + k^2) \leq (R-1)^2\},$$

$$\mathcal{S}_2 = \{(i, j, k) \in \mathcal{S}_0 : |j| \leq i, |k| \leq i\},$$

$$\mathcal{S}_3 = \{(i, j, k) \in \mathcal{S}_0 : |j| \leq -i, |k| \leq -i\},$$

$$\mathcal{S}_4 = \{(i, j, k) \in \mathcal{S}_0 : |i| \leq j, |k| \leq j\},$$

$$\mathcal{S}_5 = \{(i, j, k) \in \mathcal{S}_0 : |i| \leq -j, |k| \leq -j\},$$

$$\mathcal{S}_6 = \{(i, j, k) \in \mathcal{S}_0 : |i| \leq k, |j| \leq k\},$$

$$\mathcal{S}_7 = \{(i, j, k) \in \mathcal{S}_0 : |i| \leq -k, |j| \leq -k\}.$$

For radius $R = 2$ the full stencil has 33 cells, the central substencil has 7 cells, and the biased substencils each have 11 cells. For radius $R = 3$ the full stencil has 123 cells, the central substencil has 33 cells, and the biased substencils each have 32 cells.

Due to the use of a Cartesian grid these stencils can be applied to any cell in the domain, as evidenced by the use of local grid indices. On an unstructured grid the stencil selection process would need to be done on a cell-by-cell basis, and the layout of biased stencils may be particularly difficult. The geometric flexibility of these kernel-based methods will again show their utility, and this stencil selection procedure will inevitably be simpler than the corresponding polynomial cases.

The final item to specify regarding the (sub)stencils is the total degree of polynomials to be used in the tail of eq. (2.31). Importantly, these maximal degrees can be specified after setting the shapes of the stencils to avoid having any predisposed requirements on layout from their presence. This work follows the approach advocated for in [35] and the degree for each substencil is set as high as possible while maintaining unisolvence. For radius $R = 2$ stencils this amounts to setting the maximal degrees as $p = 3$ for the full stencil and $p = 1$ for the substencils. For radius $R = 3$ stencils the maximal degrees are set as $p = 5$ for the full stencil, $p = 3$ for the central substencil, and $p = 2$ for the biased substencils.

2.5.2 Adaptive order WENO

With the stencil and substencil layouts fixed from the previous section, the kernel-based weighted essentially non-oscillatory (WENO) method can finally be described. The overall goal of WENO schemes is to perform several reconstructions for each cell over a set of substencils. A weighted combination of these reconstructions is then evaluated to generate the Riemann states on the interfaces between cells, and possibly at any interior points for the evaluation of source terms. For smooth data these weights should take some optimal values that serve to maximize the accuracy. Alternatively, for rough data these weights should be chosen in a way that effectively removes the contributions of the substencils containing this data, ultimately yielding a reconstruction that is (essentially) non-oscillatory.

Traditional WENO schemes in one space dimension, e.g. those in [59, 7, 13, 54], perform reconstruction only over the substencils, and through cleverly chosen optimal weights recover the accuracy of the full stencils reconstruction implicitly. The existence of such optimal weights is not guaranteed. Alternatively, they may exist but with some weights being negative, which is a source of numerical instability requiring elaborate splitting schemes for use in practical applications [93].

Consider again the goal of reconstructing a point value of the solution at \mathbf{x}_* . There are now multiple reconstruction vectors that arise from solving the system eq. (2.37) over each substencil. Let $\mathbf{r}_{(q)}$ denote the reconstruction vector over the q^{th} substencil,

and similarly let $\mathbf{g}_{(q)}$ denote the restriction of the given data to that substencil. The optimal weights $\gamma_k \in \mathbb{R}$ must then satisfy

$$\mathbf{r}_{(0)}\mathbf{g}_{(0)} = \sum_{q=1}^{N_s} \gamma_q \mathbf{r}_{(q)}\mathbf{g}_{(q)}, \quad (2.39)$$

for arbitrary given data \mathbf{g} . If $\tilde{\mathbf{r}}_{(q)}$ denotes padded versions of the substencil reconstruction vectors to match the full stencil, with zeros added for all cells not in a particular substencil, then the existence of such optimal weights is equivalent to the statement $\mathbf{r}_{(0)} \in \text{span}\{\tilde{\mathbf{r}}_{(q)}\}$ for $q = 1, \dots, N_s$. These requirements generally do not hold for kernel-based reconstruction methods. Traditional one-dimensional WENO schemes have been extended to kernel-based reconstruction schemes by instead finding the optimal weights approximately in the least-squares sense [82]. This works in part because the approximating functions constructed in the kernel-based methods tend towards polynomials in the limit of large length scales presuming that an appropriate kernel is used [25].

In multiple dimensions the situation worsens. Not only will exact optimal weights satisfying eq. (2.39) generally fail to exist, but also approximate weights found through a least-squares procedure similar to that in [82] will not yield satisfactory results and the accuracy of the full stencil will not be realized by the combined substencils. Additionally, the optimal weights depend on the evaluation point \mathbf{x}_* , hence even if such optimal weights could exist their use would complicate the implementation substantially.

Clearly traditional WENO approaches are inappropriate for use in the sought mul-

tidimensional reconstruction scheme. Fortunately, a fix is readily available. Instead of attempting to reproduce the action of the full stencil reconstruction implicitly through the substencil reconstructions, one can simply include the full stencil reconstruction explicitly. One immediate benefit is that a single function can be built that provides a valid reconstruction throughout a cell, which can then be evaluated at all points where Riemann states or interior points are needed. This sidesteps the issue of having different sets of optimal weights for different evaluation points, or indeed the non-existence of such weights. This work utilizes the adaptive order WENO (WENO-AO) formulation from [4], though it also bears similarity to the related central WENO (CWENO) methods [20, 19].

Ultimately a point value at \mathbf{x}_* is reconstructed as

$$\tilde{f}(\mathbf{x}_*) = \frac{\omega_0}{\gamma_0} \mathbf{r}_{(0)}^T \mathbf{g}_{(0)} + \sum_{q=1}^{N_S} \left(\omega_q - \omega_0 \frac{\gamma_q}{\gamma_0} \right) \mathbf{r}_{(q)}^T \mathbf{g}_{(q)}, \quad (2.40)$$

where ω_q are the nonlinear weights and γ_q are the linear weights⁶. The linear weights are fixed constants that can be chosen solely for the sake of stability, with the only constraints being that they are positive and sum to one. Adapting the guidance from [4]

⁶In the traditional WENO literature the terms *optimal weights* and *linear weights* are often used interchangeably. Here the term *linear weights* is preferred since optimal weights do not exist, though the notation is intentionally reused.

to the multidimensional setting, the linear weights are set as

$$\gamma_0 = \gamma_{hi} \tag{2.41}$$

$$\gamma_1 = (1 - \gamma_{hi})\gamma_{lo} \tag{2.42}$$

$$\gamma_q = \frac{1}{2d} (1 - \gamma_{hi})(1 - \gamma_{lo}), \quad q > 1, \tag{2.43}$$

where $\gamma_{hi} = \gamma_{lo} = 0.75$ are tunable constants. This approach places more weight on the full and central (sub)stencils, then divides the remaining weights equally among all of the biased substencils. The nonlinear weights serve to remove (sub)stencils from contributing when they contain rough data, and their construction is the topic of the next section.

2.5.3 Smoothness indicators and nonlinear weights

The nonlinear weights, ω_q , introduced in the previous section lie at the core of any WENO method, and are responsible for many of their desirable properties. These nonlinear weights need to be constructed to have two limiting behaviors. First, for smooth data each nonlinear weight should approach the corresponding linear weight, $\omega_q \rightarrow \gamma_q$. If all substencils have smooth data and all of the nonlinear weights approach their linear counterparts, then the factor ω_0/γ_0 in eq. (2.40) will tend to one, and the parenthetical term in each summand will drop to zero. In this limit only the full stencil reconstruction remains and full accuracy is achieved. Second, for rough data the substencils containing it should have weights that drop to zero. Naturally, if any substencil contains rough data

then the full stencil will also contain that rough data and ω_0 should also fall towards zero. Returning again to eq. (2.40), in this case the factors $\frac{\omega_0}{\gamma_0}$ and $\omega_0 \frac{\gamma_q}{\gamma_0}$ will both approach zero, and only those substencils with non-negligible weights will survive.

In order to build up this behavior for the nonlinear weights there first needs to be some way to quantify the smoothness or roughness of the data on a given substencil. These quantities, called smoothness indicators, attempt to capture how oscillatory the reconstructed function on a particular substencil is. One particular method for determining this roughness is to use a (scaled) Sobolev semi-norm, such as

$$|\tilde{f}_q|_{\tilde{H}_p}^2 = \sum_{0 < |\boldsymbol{\alpha}| \leq p} \Delta^{2|\boldsymbol{\alpha}| - d} \int_{\Omega_h} \left(\frac{\partial^{|\boldsymbol{\alpha}|} \tilde{f}_q}{\partial \mathbf{x}^{\boldsymbol{\alpha}}} \right)^2 d\mathbf{x}, \quad (2.44)$$

where Ω_h is the cell where reconstruction is being performed, $\tilde{f}_q(\mathbf{x})$ is the function reconstructed with respect to \mathcal{S}_q , and $\boldsymbol{\alpha} \in \mathbb{N}^d$ is a multiindex. The integral captures the cell-averages of the squares of all partial derivatives up to some maximal order p . Experimentally, p has been set equal to the stencil radius as it seems to give the best results, though a more nuanced approach would be considering in the future.

There are two scalings present here, one is the term $\Delta^{-d} = \|\Omega_h\|^{-1}$ which is present for cell-averaging, and second is the term $\Delta^{2|\boldsymbol{\alpha}|}$ which makes the smoothness indicators scale-invariant. This second scaling means that the relative roughness of a set of data is independent of the cell sizes present, and operates in essentially the same fashion as the undivided differences present in standard polynomial WENO schemes (see e.g. [96]).

These semi-norms can not be evaluated in closed form, and instead need to be ap-

proximated. Fortunately their exact values are unnecessary for their intended purpose here, which is simply to quantify the relative roughness of some given data. To simplify this approximation it is useful to first make the change of variables $\widehat{\mathbf{x}} = (\mathbf{x} - \mathbf{x}_h)\Delta$ and transform Ω_h to $\widehat{\Omega} = [-\frac{1}{2}, \frac{1}{2}]^d$ to absorb the cell-averaging factors

$$|\widetilde{f}_q|_{H_p}^2 = \sum_{0 < |\boldsymbol{\alpha}| \leq p} \Delta^{2|\boldsymbol{\alpha}|} \int_{\widehat{\Omega}} \left(\frac{\partial^{|\boldsymbol{\alpha}|} \widetilde{f}_q}{\partial \widehat{\mathbf{x}}^{\boldsymbol{\alpha}}} \right)^2 d\widehat{\mathbf{x}}. \quad (2.45)$$

The ultimate smoothness indicators β_q associated with each substencil can then be formed by approximating these integrals through a simple midpoint quadrature rule

$$\beta_q = \sum_{0 < |\boldsymbol{\alpha}| \leq p} \Delta^{2|\boldsymbol{\alpha}|} \left(\frac{\partial^{|\boldsymbol{\alpha}|} \widetilde{f}_q}{\partial \widehat{\mathbf{x}}^{\boldsymbol{\alpha}}} \Big|_{\mathbf{0}} \right)^2, \quad (2.46)$$

noting that the cell center been shifted to the origin by the given change of variables.

This in turn requires a way to evaluate arbitrary derivatives of the reconstructed functions. This can easily be achieved by revisiting eq. (2.34) and eq. (2.35) then replacing the point evaluation sample vector \mathbf{T} by evaluations of the desired derivative

$$\mathbf{T}_l = \frac{\partial^{|\boldsymbol{\alpha}|} K(\widehat{\mathbf{x}}, \widehat{\mathbf{x}}_l)}{\partial \widehat{\mathbf{x}}^{\boldsymbol{\alpha}}} \Big|_{\mathbf{0}}, \quad (2.47)$$

with a similar replacement for the \mathbf{S} vector, and subsequently finding a derivative reconstruction vector via eq. (2.37). Notably, this process needs to be repeated for each partial derivative indexed by $\boldsymbol{\alpha}$.

Finally, the smoothness indicators can be used to produce the nonlinear weights in a variety of ways. Indeed both the definitions of the smoothness indicators [55, 84]

and nonlinear weights [54, 8, 14] have given rise to vigorous research efforts over the last several years. This work considers the simplest case of the WENO-JS weights [59] with a minor modification

$$\widetilde{\omega}_q = \frac{\gamma_q}{\beta_q^2 + \varepsilon}, \quad \omega_q = \frac{\widetilde{\omega}_q}{\sum_q \widetilde{\omega}_q}, \quad (2.48)$$

where ε in the denominator of the unnormalized weights $\widetilde{\omega}_q$ is present to prevent division by zero in the case of vanishing smoothness indicators. The standard WENO-JS weights take this denominator to be $(\beta_q + \varepsilon)^2$. It has been found that it is more robust to move the power onto β_q directly, and take $\varepsilon = 10^{-40}$ at a much smaller value than the typical WENO-JS case.

Chapter 3

A kernel-based finite volume method for the compressible Euler equations

This chapter is based on the paper [73] submitted to the *Astrophysical Journal*.

3.1 Introduction

This chapter is concerned with the solution of the compressible Euler equations

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ E \end{pmatrix}, \quad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p \delta_{ij} \\ u_j (E + P) \end{pmatrix}, \quad (3.1)$$

where the $n = 5$ conserved quantities are the density ρ , linear momenta ρu_i , and total energy E , respectively. This equation is closed by the adiabatic equation of state for the

pressure, $p = (\gamma - 1) \left(E - \frac{1}{2} \rho u_k u_k \right)$, where γ is the ratio of specific heats, and Einstein summation notation has been adopted. It should be noted that the use of the adiabatic equation of state is not an essential requirement for the methods presented herein, but is rather a useful restriction to allow focus to remain on the core numerical methods.

The preceding chapter deliberated on kernel-based reconstruction methods for scalar valued data. These reconstruction techniques, both the linear and nonlinear versions, form an important foundation for the development of the finite volume method described in this chapter. However, as the equations of interest in this work are *systems* of conservation laws, such as eq. (3.1), these scalar valued reconstruction schemes will need to be adapted to treat multicomponent data.

This chapter will proceed by first performing this adaptation to the multicomponent case by discussing choices of reconstruction variables in Section 3.2. WENO style nonlinear reconstruction is notably more expensive than linear reconstruction, and Section 3.3 will present a cheap local indicator that can flag whether a given cell can safely fall back to a cheaper linear reconstruction or not. One motivation behind the development of these new methods is the ability to simulate extreme phenomena. In these cases special attention must be paid to the numerical preservation of positive density and pressure, which is discussed in Section 3.4. A brief aside will be made in Section 3.5 to discuss the chosen time integrators. Finally, a suite of informative and challenging benchmark problems will be solved and presented in Section 3.6.

3.2 Selection of reconstruction variables

The most natural approach to the reconstruction of multicomponent data is to simply treat each component separately with a standard scalar valued method, such as those presented in Chapter 2. Unfortunately, component-by-component reconstruction over the conservative variables generally leads to poor results, and the use of WENO is insufficient to suppress the onset of unacceptable oscillations [108]. The resolution of this lies in the observation that arbitrary linear combinations of conserved quantities are also conserved, and that the cell-averaging procedure respects these linear combinations.

While the standard conservative variables (e.g. density, linear momenta, and total energy for the Euler equations) may give poor results, there may be some other equivalent set of variables that can give useful results. These alternative variables will only be locally defined, hence one could not simply transform the governing equations ahead of time, though this is plenty sufficient as the reconstruction process is local anyway.

Let $\langle \mathbf{U} \rangle_h$ denote the vector of cell-averaged conservative variables on cell $\Omega_h \in \mathcal{S}_0$. The cell averages of a corresponding set of so-called reconstruction variables, $\langle \mathbf{W} \rangle_h$, can be obtained as

$$\langle \mathbf{W} \rangle_h = \mathbf{\Phi} \langle \mathbf{U} \rangle_h, \quad \forall \Omega_h \in \mathcal{S}_0. \quad (3.2)$$

where $\mathbf{\Phi} \in \mathbb{R}^{n \times n}$ is a constant invertible matrix. The WENO method described in Section 2.5 can then be applied componentwise to these variables. If \mathbf{W}_s denotes the vector of point values at some \mathbf{x}_s obtained from this process, then the corresponding

conservative variables can be found as

$$\mathbf{U}_s = \Phi^{-1} \mathbf{W}_s. \quad (3.3)$$

It is crucial that the transformation encoded by Φ be linear and constant over the whole stencil. This assures that the reconstruction variables produced by eq. (3.2) truly are cell-averaged quantities, preserving the accuracy of the underlying reconstruction scheme. Nonlinear transformations, such as the conversion from conservative variables to primitive variables (e.g. $(\rho, \rho u_i, E) \rightarrow (\rho, u_i, p)$ for the Euler equations), will irreversibly reduce the reconstruction to second-order accuracy at most because the cell-averaging procedure is not respected. For example, consider the cell average velocity and note that generally

$$\int_{\Omega_h} u_i d\mathbf{x} \neq \frac{\int_{\Omega_h} \rho u_i d\mathbf{x}}{\int_{\Omega_h} \rho d\mathbf{x}}. \quad (3.4)$$

Replacing these integrals with midpoint quadrature rules recovers equality, but induces the mentioned decay to second-order accuracy. Similar arguments hold for the recovery of the pressure, or for other more general nonlinear transformations.

The following sections outline two useful sets of reconstruction variables. First the classic choice of characteristic variables is presented and used to motivate why local linear transformations can yield superior results. While these variables are locally optimal in some sense [108], but also exhibit a few notable deficiencies that will be cataloged. After this a new set of reconstruction variables, dubbed the linearized primitive variables, are presented that attempt to circumvent these deficiencies without sacrificing

reconstruction quality.

3.2.1 Characteristic variables

As discussed in the introduction, the spectral properties of the flux Jacobians eq. (1.3) associated to a system of conservation laws have deep implications on the nature and behavior of the system. Moreover, the eigenvectors of these flux Jacobians yield sets of variables that are ideal candidates for the reconstruction process. Recall again eq. (1.3) where \mathbf{A}_η denotes the flux Jacobian in the direction η evaluated with respect to the reference state $\tilde{\mathbf{U}}$

$$\mathbf{A}_\eta = \left. \frac{\partial \mathbf{F}_\eta}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}}, \quad (3.5)$$

and consider using the eigendecomposition of the flux Jacobian to set transformations in eq. (3.2) as

$$\mathbf{A}_\eta = \mathbf{R}\mathbf{\Lambda}\mathbf{L}, \quad \Phi = \mathbf{L}, \quad \Phi^{-1} = \mathbf{R}. \quad (3.6)$$

The utility of this change of variables can be appreciated by first considering the special case of a linear and constant-coefficient hyperbolic system in one space dimension. In this case the governing equation is simply

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial}{\partial x} (\mathbf{A}\mathbf{U}) = 0 \Rightarrow \frac{\partial \mathbf{U}}{\partial t} + \mathbf{A} \frac{\partial \mathbf{U}}{\partial x} = 0. \quad (3.7)$$

Inserting $\mathbf{A} = \mathbf{R}\mathbf{\Lambda}\mathbf{L}$ one finds

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{R}\mathbf{\Lambda}\mathbf{L}\frac{\partial \mathbf{U}}{\partial x} = 0, \quad (3.8)$$

$$\mathbf{L}\frac{\partial \mathbf{U}}{\partial t} + \mathbf{\Lambda}\mathbf{L}\frac{\partial \mathbf{U}}{\partial x} = 0, \quad (3.9)$$

$$\frac{\partial \mathbf{W}}{\partial t} + \mathbf{\Lambda}\frac{\partial \mathbf{W}}{\partial x} = 0, \quad (3.10)$$

which indicates that this change of variables factors the system of conservation laws into a set of decoupled transport equations. The restriction to the linear constant-coefficient case is critical to this particular manipulation.

For nonlinear systems these variables act in a qualitatively similar fashion. While the governing equations can not be exactly decoupled, this transformation still yields a set of variables that are locally decoupled. This is beneficial for the reconstruction process in that discontinuities are isolated to only the necessary fields, and it is in this sense that these variables are the optimal choice for reconstruction.

Unfortunately, these reconstruction variables also come with a few troublesome drawbacks. First, the eigendecomposition either needs to be known symbolically or a numerical decomposition routine must be used. For simple systems, like the compressible Euler equations closed with an adiabatic equation of state, this decomposition is known analytically. Other equations, or even just the Euler equations with other equations of state¹, require this decomposition to be re-derived which can be difficult or even impossible. Using numerical routines for this eigendecomposition avoids the dif-

¹This is because the pressure appears in the flux but the Jacobian is taken with respect to the conservative variables, and hence the pressure as a function of the conservative variables must be known.

faculty of the symbolic derivation, but introduces a potentially expensive routine in the deepest level of the method. Worse yet, routines for eigendecomposition are iterative by nature, and running these in parallel on architectures that operate in lock-step (like GPUs) could be challenging.

Another deficiency of characteristic variables is unique to the case of multidimensional reconstruction. Crucially, as indicated in eq. (3.6), the flux Jacobian and hence its decomposition is tied to a particular direction $\boldsymbol{\eta}$. This means that different sets of characteristic variables are needed depending on what point is being reconstructed. That is, Riemann states on x -faces, y -faces, and z -faces, will each be reconstructed using their own sets of characteristic variables aligned with their respective face normals². As a result the smoothness indicators and nonlinear weights within the WENO procedure have to be recomputed multiple times per cell. This is the single most expensive step in the hydrodynamic solver, so doubling or tripling its cost is unacceptable. Furthermore, one of the driving motivations for pursuing multidimensional reconstruction is the ability to build all Riemann states simultaneously.

3.2.2 Linearized primitive variables

The deficiencies of characteristic variables, particularly as relevant for multidimensional reconstruction, motivates the investigation of other choices of reconstruction

²As a small aside, it is also unclear how to use characteristic variables when reconstructing internal states as there is no preferred direction to consider.

variables. One of the defining features of characteristic variables is that discontinuities are isolated to only the necessary fields. Primitive variables share this property to some extent, which is best illustrated by considering an isolated contact discontinuity. Here the density will be discontinuous while the normal velocity and pressure remain continuous. Better yet, primitive variables are naturally direction-independent and are thus significantly more economical for use in multidimensional reconstruction schemes.

Of course, primitive variables can not be used directly for reconstruction without decaying to second-order accuracy as mentioned in Section 3.2 and illustrated in eq. (3.4). To retain the benefits of primitive variables while also maintaining high-order accuracy one could instead consider a linearization of the map from conservative variables. Let \mathbf{U} denote the vector of conservative variables as before, and let \mathbf{V} similarly denote the vector of primitive variables. For the Euler equations one finds that $\mathbf{V}(\mathbf{U})$ is

$$\mathbf{V}(\mathbf{U}) = \begin{pmatrix} \rho \\ \frac{\rho u_1}{\rho} \\ \frac{\rho u_2}{\rho} \\ \frac{\rho u_3}{\rho} \\ (\gamma - 1) \left(E - \frac{(\rho u_1)^2 + (\rho u_2)^2 + (\rho u_3)^2}{2\rho} \right) \end{pmatrix} = \begin{pmatrix} \rho \\ u_1 \\ u_2 \\ u_3 \\ p \end{pmatrix}. \quad (3.11)$$

Fix $\tilde{\mathbf{U}}$ as a reference state in conservative variables, which is set to the cell-average state in the cell where reconstruction is being performed just as before. The transformation

in eq. (3.11) is linearized around $\tilde{\mathbf{U}}$ as

$$\mathbf{V}(\mathbf{U}) \approx \mathbf{V}(\tilde{\mathbf{U}}) + \left. \frac{\partial \mathbf{V}}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} (\mathbf{U} - \tilde{\mathbf{U}}), \quad (3.12)$$

which can be suggestively rearranged to

$$\mathbf{V}(\mathbf{U}) \approx \left. \frac{\partial \mathbf{V}}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} \mathbf{U} + \left(\mathbf{V}(\tilde{\mathbf{U}}) - \left. \frac{\partial \mathbf{V}}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} \tilde{\mathbf{U}} \right). \quad (3.13)$$

Notably, the term in parentheses is constant and hence plays no role in the calculation of the smoothness indicators in eq. (2.46).

This motivates the particular choice of transformation matrix

$$\Phi = \left. \frac{\partial \mathbf{V}}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ -\tilde{u}_1/\tilde{\rho} & 1/\tilde{\rho} & 0 & 0 & 0 \\ -\tilde{u}_2/\tilde{\rho} & 0 & 1/\tilde{\rho} & 0 & 0 \\ -\tilde{u}_3/\tilde{\rho} & 0 & 0 & 1/\tilde{\rho} & 0 \\ (\gamma-1)\|\tilde{\mathbf{u}}\|^2/2 & (1-\gamma)\tilde{u}_1 & (1-\gamma)\tilde{u}_2 & (1-\gamma)\tilde{u}_3 & (\gamma-1) \end{pmatrix}, \quad (3.14)$$

where tildes again indicate values obtained from the reference state $\tilde{\mathbf{U}}$. Note that the velocities $\tilde{u}_i = \tilde{\rho} \tilde{u}_i / \tilde{\rho}$ and $\|\tilde{\mathbf{u}}\|^2 = (\tilde{u}_1^2 + \tilde{u}_2^2 + \tilde{u}_3^2)$ are only second-order accurate representations since they are naively calculated from the conservative variables in the reference state. Just as in the previous section, the accuracy of these quantities has no impact on the accuracy of the reconstruction. The inverse transformation is easily obtained as the Jacobian of the inverse map $\mathbf{U}(\mathbf{V})$ evaluated at the reference primitive

state $\mathbf{V}(\tilde{\mathbf{U}})$, and takes the particular form

$$\Phi^{-1} = \frac{\partial \mathbf{U}}{\partial \mathbf{V}} \Big|_{\mathbf{V}(\tilde{\mathbf{U}})} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ \tilde{u}_1 & \tilde{\rho} & 0 & 0 & 0 \\ \tilde{u}_2 & 0 & \tilde{\rho} & 0 & 0 \\ \tilde{u}_3 & 0 & 0 & \tilde{\rho} & 0 \\ \|\tilde{\mathbf{u}}\|^2/2 & \tilde{u}_1 & \tilde{u}_2 & \tilde{u}_3 & 1/(\gamma-1) \end{pmatrix}. \quad (3.15)$$

While the primary motivation for developing these alternative reconstruction variables was the need for directional independence in multidimensional reconstruction, they have some additional useful properties that could make them an interesting choice even for one dimensional reconstruction schemes.

They are particularly simple to formulate and extend naturally to other hyperbolic systems. For instance, the ideal magnetohydrodynamics equations considered in the following chapter have degenerate characteristic variables [85] due to the fact that the governing equations are only weakly hyperbolic. These linearized primitive variables have no dependence on the hyperbolicity of the governing equations, and hence require no special attention or special cases for different reference states.

Returning to the compressible Euler equations, it should be noted that the equation of state only influences the final row of the forward and inverse transformation matrices. This means that swapping out other equations of state becomes a fairly simple matter so long as the partial derivatives $\frac{\partial P}{\partial U}$ and $\frac{\partial \mathbf{U}}{\partial P}$ can be computed. Furthermore, only one of these partials are really needed (say $\frac{\partial P}{\partial U}$) and the inverse transformation easily

backed out by substitution against the final row of Φ . This extends the utility of these variables even to cases where the equation of state is tabulated, and thus $\frac{\partial P}{\partial U}$ is only known approximately.

3.3 Avoiding WENO via a KXRCF indicator

The high computational cost of evaluating the smoothness indicators (eq. (2.46)) and nonlinear weights (eq. (2.48)) for WENO was mentioned in the previous section regarding reconstruction variables. The presented linearized primitive variables mitigate this cost by ensuring that smoothness indicators are only computed once for each cell and component. These methods also have a problematically large memory footprint, particularly when using multidimensional reconstruction. Reconstruction variables are set on a per-cell basis since they utilize the local state as a reference when creating the transformation matrices Φ and Φ^{-1} . This means that the reconstruction procedure on each cell needs a full copy of the values on its local stencil to apply these transformations to. This memory overhead becomes problematic on highly parallel co-processors (such as graphics processing units) which can process huge numbers of cells simultaneously, but are lacking in on-device storage³.

However, it is also true that many cells do not need to utilize WENO at all. Higher accuracy and lower computational expense could simultaneously be obtained if a some

³These architectures often also have limited register space, so even clever use of stack allocated workspace is unlikely to help.

criteria to determine the necessity of WENO could be evaluated. Crucially, this criteria should be cheap enough to evaluate that the overhead it adds is mitigated by the savings that it affords. Schemes that adaptively mix high-order accurate linear reconstruction with more stable nonlinear reconstruction are termed *hybrid* schemes, and have been investigated both in the context of finite volume and finite difference methods [50, 116, 15] and discontinuous Galerkin (DG) methods [63, 39, 3] among others.

WENO type schemes fall into the broader category of *a-priori* methods, meaning that limitation is performed before any forward integration in time is applied. In contrast, *a-posteriori* methods, such as multidimensional optimal order detection (MOOD), only apply limitation after attempting, and potentially failing, to integrate forward in time. Broadly speaking, MOOD methods seek to use a high-order accurate linear scheme everywhere in space irrespective of the local data. Elaborate acceptance/rejection criteria are then applied to each cell to determine if the resulting solution is valid, and inevitably some cells will have invalid solutions. The offending cells are rolled back to the previous time step, and the spatial discretization is locally reduced to one of lower order. This may be repeated several times, ultimately cascading to a stable first-order accurate scheme as needed. See [18] and [24] for an introduction to the MOOD method. More recently MOOD has been incorporated into a kernel-based approach in [9]. The essential deficiency of MOOD methods, or any *a-posteriori* method for that matter, is that they can be challenging to implement for highly parallel archi-

tures. Hybrid WENO schemes essentially try to mimic this behavior in an *a-priori* manner.

The KXRCF indicator, introduced in [63] and subsequently refined and recast in [39], is designed for use in DG methods but also happens to be well suited for the multidimensional reconstruction schemes presented herein. The total reconstruction process proceeds in three steps. First, high-order linear reconstruction is performed on all cells, which can be applied directly over the conservative variables. Second, the KXRCF indicator is computed for each cell by considering its Riemann states as compared to neighboring states. Finally, all cells where the indicator is too large are flagged, and reconstruction is redone using the WENO method.

The KXRCF indicator itself is defined as follows. Let s be an index that labels all quadrature points on all faces of a given cell. Let $q(\mathbf{U})$ be some yet to be determined indicator variable given as a function of the conservative variables, and let $q_s^{(-)}$ and $q_s^{(+)}$ denote its values inside and outside the cell respectively at the quadrature point indexed by s . The essential action of the KXRCF indicator is to flag cells where jumps in this indicator variable are too large. To this end, any cell satisfying

$$\frac{\max_s \left\{ \left| q_s^{(+)} - q_s^{(-)} \right| \right\}}{q(\tilde{\mathbf{U}})} > \Delta^{3/2}, \quad (3.16)$$

is set to use WENO. The denominator is evaluated using the cell-average state of the cell in consideration ($\tilde{\mathbf{U}} = \langle \mathbf{U} \rangle_h$), and serves to normalize the jumps in q present in the numerator. The grid scale factor of $\Delta^{3/2}$ can be appreciated by considering two cases.

First consider a r^{th} -order accurate reconstruction scheme applied on two neighboring cells having smooth data locally. Then each $q_s^{(\pm)}$ on the shared face should be accurate to $O(\Delta^r)$, and the normalized jumps on the left side of eq. (3.16) should also be of size $O(\Delta^r)$. Alternatively, if one or both cells have stencils containing discontinuous data then the values $q_s^{(\pm)}$ will generally decay to only first order accuracy, and the normalized jumps will scale like $O(\Delta) - O(1)$.

There is little guiding theory on how to choose the indicator variable $q(\mathbf{U})$. For the compressible Euler equations it should be discontinuous across shocks at a minimum. The form proposed in eq. (3.16) is simplest to apply if q is non-zero, and better yet if $q(\mathbf{U}) > 0$ for all admissible states \mathbf{U} . This work considers the entropy

$$q(\mathbf{U}) = S = \frac{p}{\rho^\gamma}, \quad (3.17)$$

which is also jumps across contact discontinuities. One reasonable alternative would be to simply use the pressure, $q = p$, which would not flag contacts. It has been observed both in MOOD methods [9] and in previous hybrid method studies [17] that order reduction may be unnecessary near contacts. Another reasonable alternative would be to use the density, $q = \rho$. This is trivial to evaluate and minimizes the overhead of flagging cells, but has a tendency to mark cells for WENO too often.

The power of $3/2$ on Δ present in the cutoff of eq. (3.16) was chosen empirically to provide meaningful computational savings while also avoiding mischaracterizing cells. Ultimately it is better to incorrectly flags cells as needing WENO than to miss

cells and obtain unreliable results. Compared to [63] and [39], eq. (3.16) provides a more conservative indicator that tags cells as needing WENO likely more often than necessary. Refining this indicator would be one interesting avenue for future work.

3.4 Positivity preservation

The previous section discussed the avoidance of WENO reconstruction when nearby data is sufficiently smooth. In the opposite extreme, there may be data with strong enough discontinuities that even WENO is incapable of providing sufficient limitation. These cases are particularly salient in astrophysical applications where densities and pressures may easily jump several orders of magnitude across a shock. It is crucial for the well-posedness of eq. (3.1) (e.g. preservation of hyperbolicity), and similarly important for their numerical solution, that density and pressure remain positive. WENO methods are only *essentially* non-oscillatory, and small oscillations near discontinuities are to be expected. Unfortunately, oscillations that are small in comparison to a strong discontinuity can still easily overshoot towards negative densities and pressures.

So-called positivity preserving schemes are those that generate solutions with positive density and pressure given any admissible initial state. Naively one could simply enforce minimum values for these quantities. This simple approach of placing floors on these quantities generally yields poor and scientifically dubious results. The primary issue is that naively placing a floor on these quantities destroys conservation, as either

mass or internal energy must be locally injected to bring the state back to valid limits [2].

Alternatively, one can seek corrections to Riemann states such that the future cell average remains admissible. Being that this correction happens prior to the calculation of the fluxes the method will naturally remain conservative. However, the criteria for selecting corrections to the Riemann states is phrased in terms of the future cell-average state after a time integration step, hence coupling neighboring cells together. Furthermore, this could only be a feasible approach with some restrictions on how the resulting flux is found from these corrected Riemann states.

In [56] and [115] a general framework for developing positivity preserving schemes for the compressible Euler equations was presented. The action of this framework is most obvious for DG schemes, though remains extensible to the finite volume case through carefully chosen quadrature rules on the cells that are compatible with the reconstruction scheme. These methods were further elaborated upon in [2] where limits on the allowed ranges of density and pressure are selected adaptively from the local data.

The limiter employed in this work mostly follows that of [2] with minor modifications. It is presumed that the cell average values all contain valid states and that only the reconstructed Riemann states are in need of correction. The limiter seeks to constrain the fluid density from above and below, and constrains the pressure only from below.

Allowable bounds on these quantities are determined by first examining their ranges in terms of the local cell-averages.

Let the cell where limitation is being applied be indexed by $(0, 0, 0)$, and take (i, j, k) to be locally defined grid indices⁴. The tightest bounds on these quantities are

$$\tilde{\rho}_{max} = \max_{-1 \leq i, j, k \leq 1} \{ \langle \rho \rangle_{(i, j, k)} \}, \quad (3.18)$$

$$\tilde{\rho}_{min} = \min_{-1 \leq i, j, k \leq 1} \{ \langle \rho \rangle_{(i, j, k)} \}, \quad (3.19)$$

$$\tilde{p}_{min} = \min_{-1 \leq i, j, k \leq 1} \{ \langle p \rangle_{(i, j, k)} \}, \quad (3.20)$$

where the overlines indicate that these are obtained from cell-averages. Naturally $\langle p \rangle_{(i, j, k)}$ is only a second-order accurate representation of the average pressure, but accuracy is largely irrelevant here as only a workable bound on pressure is needed.

Forcing all of the Riemann states to obey these bounds would be far too restrictive. Indeed, this would aggressively clip smooth extrema and limit the solver to only first-order accuracy. These tight bounds are subsequently loosened to

$$\rho_{max} = \tilde{\rho}_{max} (1 + \kappa_2 - \kappa_2 \eta), \quad (3.21)$$

$$\rho_{min} = \tilde{\rho}_{min} (1 - \kappa_2 + \kappa_2 \eta), \quad (3.22)$$

$$p_{min} = \tilde{p}_{min} (1 - \kappa_2 + \kappa_2 \eta). \quad (3.23)$$

⁴The presentation here is shown for $d = 3$ space dimensions, but the action of the limiter is easily restricted to $d = 2$ space dimensions.

Here η is a flattener variable defined as

$$\eta = \min \left\{ 1, \max \left\{ 0, -\frac{\kappa_1 \tilde{c}_{min} + (\nabla \cdot \langle \mathbf{u} \rangle) \Delta}{\kappa_1 \tilde{c}_{min}} \right\} \right\}, \quad (3.24)$$

where \tilde{c}_{min} is the minimum local speed of sound determined in the same fashion as the bounds in eq. (3.18) - eq. (3.20), and $(\nabla \cdot \langle \mathbf{u} \rangle) \Delta$ is the local velocity divergence computed from the cell-average quantities. The scale factor of Δ makes this an undivided difference, and hence independent of the grid scale on the whole. Finally, κ_1 and κ_2 are tunable parameters which are set to $\kappa_1 = \kappa_2 = 0.3$ just as in [2].

The flattener variable and the κ constants work together to set useful, data-dependent, bounds on the density and pressure. When $\eta \rightarrow 0$ the bounds in eq. (3.21) - eq. (3.23) are loosened according to κ_2 , with the minima scaled by $(1 - \kappa_2)$ and the maximum scaled by $(1 + \kappa_2)$, and permit the widest range of values without limitation. Alternatively, as $\eta \rightarrow 1$ the bounds are tightened and tend towards the local minima and maxima, hence disallowing any under/overshoots. The flattener eq. (3.24) serves to sense locally compressive velocity fields to tighten the bounds in shocked regions, and the constant κ_1 residing within weights the relative strength of compression required to activate the flattener.

The reconstructed point values must now be corrected to lie within these bounds on density and pressure, and it is here that a minor modification is made to the overall action of the limiter as described in [2]. To truly be positivity-preserving one needs to not only correct the Riemann states, but also a set of interior values (on the cleverly

chosen quadrature points mentioned above) even if source terms are not present in the system being solved. This strengthens the limiter, and the necessity of these additional corrections was proved in [56]. It is this strengthening of the limiter that assures all *future* states are also valid under some assumptions on the numerical flux. Herein, these additional corrections are elided in cases where source terms are not present as a minor optimization, which is justified for two reasons. First, the proof of positivity preservation is intimately tied to polynomial reconstruction and does not directly apply to the present case, and second, the pathological cases requiring this extra limitation do not seem to be particularly common in practice. The present implementation does support this stronger form of limitation if desired.

3.4.1 Constraining density

Corrections to the Riemann states on a cell, if needed, are applied in two stages. The density is corrected first, which will naturally change the kinetic energy and hence the pressure. As in Section 3.3, let s index all of the Riemann states on all faces of the cell in consideration (as well as internal states if present). The density of each state, ρ_s , should obey $\rho_{min} \leq \rho_s \leq \rho_{max}$.

Corrections are applied by hybridizing the Riemann states \mathbf{U}_s with the cell-average state $\langle \mathbf{U} \rangle_0$. The strongest required correction is given by

$$\theta_\rho = \min_s \left\{ 1, \frac{\langle \rho \rangle_0 - \rho_{min}}{\langle \rho \rangle_0 - \rho_s}, \frac{\rho_{max} - \langle \rho \rangle_0}{\rho_s - \langle \rho \rangle_0} \right\}, \quad (3.25)$$

and is applied via

$$\mathbf{U}_s \leftarrow \langle \mathbf{U} \rangle_0 + \theta_p (\mathbf{U}_s - \langle \mathbf{U} \rangle_0), \quad \forall s. \quad (3.26)$$

Note that the correction is applied to *all* Riemann states to ensure conservation and consistency. Additionally, if all states initially satisfy the required bounds θ_p will be one, and no correction will be applied.

3.4.2 Constraining pressure

The corrections applied in eq. (3.26) will change all of the Riemann states and hence also change the pressure, $p = (\gamma - 1) \left(E - \frac{\|\rho \mathbf{u}\|^2}{2\rho} \right)$, so the pressure bound is only tested after this initial round of correction. Again, the states will be corrected by mixing in a first order contribution

$$\mathbf{U}_s \leftarrow \langle \mathbf{U} \rangle_0 + \theta_p (\mathbf{U}_s - \langle \mathbf{U} \rangle_0), \quad (3.27)$$

though now the strength of the correction, θ_p , is slightly harder to obtain.

For an individual Riemann state that violates the pressure minimum one could consider the corrected state in eq. (3.27) and solve

$$p(\langle \mathbf{U} \rangle_0 + \theta_{p,s} (\mathbf{U}_s - \langle \mathbf{U} \rangle_0)) = p_{min}, \quad (3.28)$$

for $\theta_{p,s}$ as the s^{th} correction strength. Expanding the above ultimately yields a quadratic equation for $\theta_{p,s}$ which can be seen in [2]. The applied correction is then formed using the smallest valid $\theta_{p,s}$

$$\theta_p = \min_s \{ \theta_{p,s} \mid 0 \leq \theta_{p,s} \leq 1 \}. \quad (3.29)$$

This relies on one crucial fact: the set of all admissible states \mathbf{U} is convex. The corrections in eq. (3.27) are thus lines in state space connecting the valid reference state $\bar{\mathbf{U}}_0$ to the potentially invalid states \mathbf{U}_s parameterized by θ_p . Convexity yields that there is precisely one value of θ_p where this line crosses the boundary of the admissible set. Moreover, smaller values of θ_p always yield valid states, and applying the most restrictive value as selected in eq. (3.29) throughout the cell will naturally yield all valid Riemann states. These notions regarding convexity of the admissible set of states are not specific to the compressible Euler equations, and recently there has been some effort in formulating a broader theory of positivity preservation in geometric terms [114].

3.5 An aside on time integrators

The present work is primarily focused on the spatial discretization of systems of hyperbolic conservation laws, and has to this moment left time continuous in accordance with the method of lines. There are however a few critical factors to consider when selecting a time integrator that are worth discussing in some detail.

Fortunately, hyperbolic problems typically are not stiff and explicit time integrators are well suited to their solution. There may be widely varying time scales throughout the full history of a problem, so the selected scheme should be able to easily adjust to variable time step sizes. Great care has been put into the development of highly accurate

spatial schemes and the considered time integrators should also exhibit reasonably high-order accuracy. Finally, the stability properties of the time integrator are a crucial factor in the ability of the scheme to evolve flows with strong shocks.

Multistage Runge-Kutta type integrators are a natural choice to satisfy all of these constraints. For the sake of stability one may consider the strong stability preserving Runge-Kutta (SSP-RK) methods [62, 30, 100]. Typically the time step size is determined through a Courant-Friedrichs-Lewy (CFL) type condition derived from an estimate of the largest wavespeed in the domain, and is set solely for stability irrespective of accuracy. By selecting a time integrator with reasonably high-order accuracy, say 3rd or 4th order, this gives temporal errors that diminish with the spatial errors as the grid is refined since the CFL condition will also force a temporal refinement.

On the other hand, CFL based time step selection provides little control over the temporal error accrued at an individual spatial resolution. Outside of fluid dynamics, general purpose time integrators rely on error estimates to select time step sizes, and seek to keep difference between two solutions obtained by integrators of different orders below some tolerance. The overhead cost of producing a second solution is substantially lessened by using an embedded scheme, where the two separate integrators share right hand side evaluations. The utility of error estimator based Runge-Kutta schemes in application to computational fluid dynamics, and hyperbolic equations more broadly, have recently been investigated [79].

Before selecting time integrators for the present solver there is one final design criteria to consider. Apart from the numerical considerations mentioned above there are also implementation considerations to be mentioned. Principle among these is that the solver is intended to be run on graphics processing units (GPUs) which have substantially smaller amounts of on-board memory (10s of GB) than typical compute nodes (100s of GB). Transfers from larger pool of host memory into the device memory is much too slow to incorporate within the time integration process, so it is critical that all necessary data fit into device memory. Low storage time integrators seek to minimize the number of stored stages and right hand side evaluations by having a sparse, and typically structured, Butcher tableau [61, 79].

The selected time integrators are low-storage Runge-Kutta schemes with embedded error estimators, and potentially with strong stability preservation. The present work considers four different integrators. First are the $RK3(2)5_F[3S_+^*]$ and $RK4(3)9_F[3S_+^*]$ methods introduced in [79] which are 3rd and 4th order accurate overall and have embedded schemes of one order less. These also have the first-same-as-last (FSAL) property allowing the final right hand side evaluation in each step to be reused at the start of the next step. In [79] these methods were found to strike a good balance between accuracy and the amount of work performed within each step, and often required the smallest total number of right hand side evaluations to reach a given error out of the methods tested. These schemes perform well on relatively smooth problems, but do not

have the SSP property. To handle problems with strong shocks a four stage 3rd order SSP(4,3,2) scheme [62] and a ten stage 4th order SSP(10,4,3) scheme [61] are also considered. Each of these utilizes embedded methods of one order lower as proposed in [30]. These methods are not FSAL in nature.

3.5.1 Time step size selection

Regardless of which of the four time integrators is used, the time step sizes are determined in the same way. Let $\langle \mathbf{U} \rangle_h^n$ and $\langle \widehat{\mathbf{U}} \rangle_h^n$ denote to solutions obtained by the outer (higher order) and embedded (lower order) integrators on cell h . A scaled estimate of the error is then

$$e_n = \left(\frac{1}{MN_c} \sum_h \sum_c \left(\frac{\langle \mathbf{U}_c \rangle_h^n - \langle \widehat{\mathbf{U}}_c \rangle_h^n}{\delta_a + \delta_r \max\{|\langle \mathbf{U}_c \rangle_h^n|, |\langle \widehat{\mathbf{U}}_c \rangle_h^n|\}} \right)^2 \right)^{1/2}, \quad (3.30)$$

where the index c indicates summation over all solution components, M is the total number of cells in the domain, and N_c is the number of solution components. The constants δ_a and δ_r are the absolute and relative tolerances on the error. These are typically set equal to each other, and as in [79] will generally lie between 10^{-6} and 10^{-2} .

The time step size is then updated according to this estimator using a PI type controller [48]. A factor on the current step size is computed as

$$f_n = \tan^{-1}(e_n^{\varepsilon_1} e_{n-1}^{\varepsilon_2} - 1), \quad (3.31)$$

where ε_1 and ε_2 are the controller parameters summarized in Table 3.1 as adopted from [79] and [30], and the inverse tangent is present to smooth the transition to larger time steps when the error is sufficiently small.

Table 3.1: Time step size controller parameters for the considered schemes. The parameters for the $[3^*_+]$ schemes come from [79] and those for the SSP schemes come from [30].

Scheme	RK3(2)5 _F [3S ₊ [*]]	RK4(3)9 _F [3S ₊ [*]]	SSP(4,3)	SSP(10,4)
ε_1	-0.7/3	-0.38/4	-0.8/3	-0.8/3
ε_2	0.23/3	0.18/4	0.31/3	0.31/3

From here there are three distinct cases to consider. First, if the factor f_n is larger than some threshold the $\langle \mathbf{U} \rangle^n$ solution is accepted and the next step is taken using

$$\Delta t_{n+1} = \min \left\{ f_n \Delta t_n, C \frac{\Delta}{\lambda_{max}} \right\}, \quad (3.32)$$

where the second term is a classic CFL type constraint relying on a user-specified constant C and an estimate of the largest wavespeed λ_{max} . This extra limitation seems to be necessary in strongly shocked problems where these error-estimator based schemes are less tested. It is entirely possible that better choices of the controller parameters or other embedded schemes could sidestep this requirement.

Alternatively, if the proposed factor f_n is too small then the estimated error must be too large. In this case the solution $\langle \mathbf{U} \rangle^n$ is rejected. The solver is wound back to state $\langle \mathbf{U} \rangle^{n-1}$ and solved forward again with $\Delta t_n \leftarrow f_n \Delta t_n$. This obviously comes with an overhead cost since some amount of work needs to be thrown away. The controller

parameters are chosen to minimize the frequency of rejected steps while also growing the time step sizes when permissible to avoid over-resolving time.

As a final case, this is the ideal time to also verify that the solution is physically admissible. If any $\langle \mathbf{U} \rangle_h^n$ has negative density or pressure, or has accumulated any not-a-number (NaN) values, the step is rejected. In this case, the solver again winds back to the state $\langle \mathbf{U} \rangle^{n-1}$ and proceeds using Δt_n set according to

$$\Delta t_n \leftarrow \begin{cases} \min \left\{ \frac{\Delta t_n}{4}, \frac{C}{4} \frac{\Delta}{\lambda_{max}} \right\} & \text{first rejection} \\ \frac{\Delta t_n}{2} & \text{otherwise} \end{cases} . \quad (3.33)$$

This limits the step size to one fourth of the maximum allowed from the CFL constraint on the first rejection, and then successively cuts the time step size by a factor of two after each rejection. The solver is aborted if too many rejections occur within a single time step regardless of their origin. The allowable limit is configurable, and defaults to four.

The final item to specify is how the initial time step is selected to get the solver primed. There are elaborate schemes in existence [48] for general purpose integrators. However, as the CFL based approach is still a reasonable one it suffices to initiate Δt_1 using it. The error adaptive controller will gradually grow the time step size so a restrictive initial step size of

$$\Delta t_1 = \frac{C}{100} \frac{\Delta}{\lambda_{max}}, \quad (3.34)$$

is used. This small initial time step size is also in line with the slow-start advice given

in [67].

3.6 Numerical results

The full method for numerically solving the compressible Euler equations with an adiabatic equation of state has now been described. What remains is to test the method on a variety of benchmark problems to assess its capabilities and merits. The suite of test problems and the significance of each one is described below, but first there are a some generalities in the method that need to be fixed in place.

The stencil radius, R , is the main parameter governing the achievable order of accuracy. Throughout this section, as in Chapter 2, radius $R = 2$ and $R = 3$ stencils are considered, which should give at least 4th and 6th order accuracy respectively. To this end the number of quadrature points per direction used for flux and source term integration is set equal to R to match in accuracy. This means there will be $R^{(d-1)}$ Riemann states per face, and R^d internal states per cell if source terms are included. The length scale ℓ in the kernel function eq. (2.3) is tied to the grid scale as $\ell = 5\Delta$ in all cases.

Numerical fluxes will be formed from the reconstructed Riemann states using either the Harten-Lax-van Leer (HLL) flux [52], or a low-Mach corrected HLLC flux called HLLC+ [16]. The former is a now classic approximate Riemann solver with excellent stability and positivity properties when appropriate wavespeed estimates are used [6]. The HLLC flux is a modification of the HLL flux that incorporates an explicit contact

discontinuity into the approximate Riemann fan which yields greatly reduced dissipation around these structures [104]. The HLLC+ flux in use here further modifies this Riemann solver to correct faulty asymptotic behavior in the low-Mach limit. Without this correction low-Mach flows can exhibit a numerical decoupling of the pressure and velocity fields giving rise to checkerboard instabilities [16].

Initial conditions are integrated to find cell-averages using the same quadrature rule with R points per direction as would be used for treating any source terms. For simplicity initial conditions are listed below using primitive variables, though naturally these would need to be converted to conservative variables prior to averaging.

Each problem solved in the subsequent sections is present to highlight different features of the proposed method. First, convergence studies using isentropic vortex problem [95, 99] are shown to validate that the method reaches (and actually exceeds) the designed order of accuracy. The Sod shock tube problem [97] is a classic one dimensional test for shock capturing, which is solved here in both grid aligned and tilted configurations. This validates that the method does not have a particular preference for grid-aligned phenomena. The Richtmeyer-Meshkov problem [86] is a qualitative test that serves to demonstrate the correct representation of complicated shock structures in the presence of instabilities. To test positivity-preservation several astrophysical jet problems [2] are solved which involve extreme flows, and are lauded as being highly challenging to capture without total failure of the code.

Finally, extensions are made to solve the compressible Navier-Stokes equations by including physical viscosity terms. The precise nondimensionalization used is discussed below in the context of the Taylor-Green vortex [103, 109, 57]. This problem is a well-established test for the accurate representation of incompressible or low-Mach turbulent flow. The bulk kinetic energy dissipation provides a quantitative test of correctness. Additionally, a viscous version of the Rayleigh-Taylor problem given in [94] is considered which serves to test the incorporation of source terms into the equations.

3.6.1 Isentropic vortex

The isentropic vortex [99] is an exceptionally rare case of an exact and fully nonlinear smooth solution to the compressible Euler equations with an adiabatic equation of state. This makes it an ideal test case to demonstrate the achievable order of accuracy of a proposed scheme. The solution consists of an exponentially localized vortex overlaid on a uniform background. Initially this takes the form

$$\rho = \left(1 + \frac{1-\gamma}{2}\omega^2\right)^{\frac{1}{\gamma-1}} \quad (3.35)$$

$$u_1 = 1 - y\omega \quad (3.36)$$

$$u_2 = 1 + x\omega \quad (3.37)$$

$$u_3 = 0 \quad (3.38)$$

$$p = \frac{1}{\gamma} \left(1 + \frac{1-\gamma}{2}\omega^2\right)^{\frac{\gamma}{\gamma-1}}, \quad (3.39)$$

where $r = 1$ is the vortex radius, $\gamma = 1.4$ is the ratio of specific heats, and the rotation rate ω is set as

$$\omega = 5 \frac{\sqrt{2e}}{4\pi} e^{-\frac{1}{2}(x^2+y^2)}. \quad (3.40)$$

Clearly from these conditions the entropy $S = p/\rho^\gamma$ is constant and equal to $1/\gamma$. This vortex rides on the uniform background flow, translating from its initial position at the origin at a velocity of $\mathbf{u}_b = (1, 1, 0)^T$.

While this is technically only an exact solution when posed on all of \mathbb{R}^d (typically with $d = 2$), the exponential localization means that it is nearly an exact solution on periodic domains. With a sufficiently large truncated domain the vortex decouples from its neighbors induced by tiling the domain periodically. On a domain of $\Omega = [-10, 10]^2$ the deviation from the uniform background is on the order of double precision machine- ϵ . Evolving this system to a final time of $T = 20$ should return the vortex to its initial position, and the initial and final states can be compared to find the accumulated error.

Table 3.2: Shown are the experimental orders of convergence for the described method with radius $R = 2$ and $R = 3$ stencils as tested on the isentropic vortex problem.

Radius $R = 2$			Radius $R = 3$		
N_x	L_1	EOC	N_x	L_1	EOC
32	1.45×10^{-3}	–	32	7.51×10^{-4}	–
64	2.27×10^{-4}	2.68	64	7.50×10^{-5}	3.32
128	1.47×10^{-5}	3.95	128	1.46×10^{-6}	5.68
256	5.39×10^{-7}	4.77	256	1.76×10^{-8}	6.38

Table 3.2 reports the L_1 errors in the density field as obtained from a sequence of doubling spatial resolutions, as well as the experimental orders of convergence, as ob-

tained by the radius $R = 2$ and $R = 3$ schemes. The experimental order of convergence, EOC, is defined as

$$\text{EOC} = \frac{\ln(E_c/E_f)}{\ln 2}, \quad (3.41)$$

where E_c and E_f denote the L_1 errors on successive grid resolutions from coarse to fine. In all cases the 4th order time integrator RK4(3)9_F[3S₊^{*}] is used with absolute and relative tolerances of $\delta_a = \delta_r = 10^{-6}$. The HLLC+ Riemann solver is used to obtain the numerical fluxes, but it should be noted that the observed behavior is similar regardless of this choice, as one would expect for a smooth problem.

The radius $R = 2$ scheme should exhibit 4th order convergence asymptotically. Indeed for resolutions beyond 64×64 this is observed, and even exceeded. Similarly, the radius $R = 3$ scheme should exhibit 6th order accuracy. This is approached at a resolution of 128×128 , and subsequently exceeded. Higher orders of accuracy than what is guaranteed is to be expected since these are just the orders that arise from the included polynomial terms discussed in Section 2.3.3 and Section 2.4. However, the additional kernel part of the expansion still serves to capture smooth phenomena and yields improved accuracy. It should also be noted that one could also seek to maximize the achieved accuracy by tuning the shape parameter ℓ in the kernel function, though doing so is beyond the scope of the current study.

3.6.2 Sod shock tube

The Sod shock tube problem [97] is a Riemann problem that gives rise to one of each type of wave supported by the compressible Euler equations. The solution consists of a right-moving shock, a slower right-moving contact discontinuity, and a left-moving rarefaction. The presence of each wave type has made this a classic test problem for any shock capturing method.

The initial conditions are piecewise constant states with the left density and pressure being $(\rho_L, p_L) = (1, 1)$, the right density and pressure being $(\rho_R, p_R) = (0.125, 0.1)$, and all velocities being zero. This is a fundamentally one dimensional problem that can be brought to $d = 2$ dimensions in a few ways. First, these conditions can be imposed in a grid-aligned manner reducing to an essentially one dimensional test. Second, through a cleverly chosen space they can be initialized in a grid-tilted manner with periodic boundaries. Slices through this will yield identical exact solutions so long as the final time is chosen to avoid any wave-wave interactions.

In the grid-aligned case a domain of $\Omega = [0, 0.04] \times [0, 1]$ is used with outflow conditions in the x -directions and periodic conditions in the y -direction. The left states are initialized for $x < 1/2$ and the right states are set elsewhere. Solved to a final time of $T = 0.2$, this corresponds with the classic benchmark problem.

In the tilted configuration a layout similar to that in [60, 66] is used. In this case the domain is $\Omega = [0, \sqrt{5}] \times [0, 2\sqrt{5}]$ and periodic boundary conditions are set in both

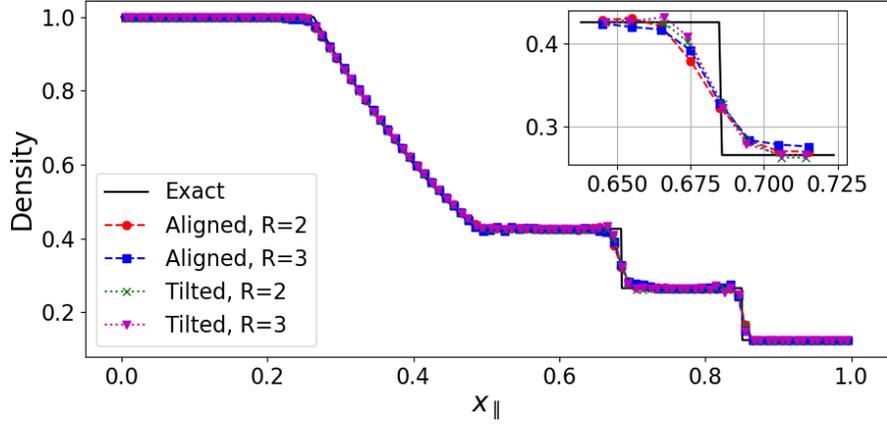


Figure 3.1: Shown is a trace of the density in Sod shock tube problem as obtained from four different cases. The solid black line shows the exact solution. The dashed lines with circle and square markers show the results of the radius $R = 2$ and $R = 3$ schemes in the grid-aligned configuration. The dotted lines with cross and triangle markers show the results of the radius $R = 2$ and $R = 3$ schemes in the grid-tilted configuration. The inset shows a zoom-in of the region near contact discontinuity.

directions. Setting $x_{||} = (2x + y)/\sqrt{5}$, the states are initialized as

$$(\rho, p) = \begin{cases} (\rho_L, p_L), & x_{||} < 0.5, 1.5 < x_{||} < 2.5, 3.5 < x_{||}, \\ (\rho_R, p_R), & \text{otherwise} \end{cases}. \quad (3.42)$$

This again is solved to the final time of $T = 0.2$, and traces of the density and pressure extracted from the line $0 < x_{||} < 1$ can be compared against the grid-aligned solution.

These configurations are solved using the SSP(4,3,2) time integrator with absolute and relative tolerances of $\delta_a = \delta_r = 10^{-4}$. The grid-aligned cases use a grid spacing of $\Delta = 1/100$, while the tilted cases use $\Delta = \sqrt{5}/250$ which yields the same number of cells along the extracted line. Traces of the density as obtained from the solver with both radius $R = 2$ and $R = 3$ stencils in each of these configurations is shown in Figure 3.1.

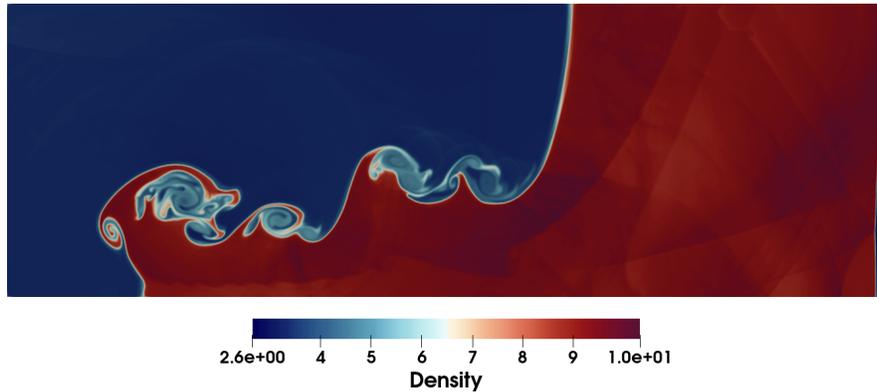


Figure 3.2: Shown is the density field for the Richtmyer-Meshkov instability at the final time of $t = 3.33$ as solved by the radius $R = 2$ scheme on a grid with spacing $\Delta = 1/512$. To highlight the interface the view has been zoomed into the region $[5/2, 11/2] \times [0, 1]$.

It can be seen, particularly in the inset that zooms in on the contact discontinuity, that these schemes exhibit minimal preference for grid-aligned phenomena over unaligned phenomena.

3.6.3 Richtmyer-Meshkov instability

The Richtmyer-Meshkov instability arises from the acceleration of an interface between fluids of differing densities. In this case a transmitted shock wave provides the acceleration. When the incoming shock is not parallel to the density interface it will refract and bend the interface, and as in the Rayleigh-Taylor case discussed below, will generally give rise to a secondary Kelvin-Helmholtz instability. The review [117] provides an excellent overview of these instabilities and the relationships between them.

The unmagnetized configuration from [86] is used herein. The domain is set $\Omega =$

$[-1/2, 11/2] \times [0, 1]$ with inflow conditions on the left, outflow conditions on the right, and reflecting conditions in the y -direction. The initial state consists of a tilted density interface and a right-moving shock. The initial density is

$$\rho = \begin{cases} \left(1 - \frac{2}{\gamma+1} \left(1 - \frac{1}{Ma^2}\right)\right)^{-1}, & x < 0.2 \\ 1, & x < y \\ \rho_D, & x \geq y \end{cases}, \quad (3.43)$$

the initial pressure is

$$p = \begin{cases} 1 + \frac{2\gamma}{\gamma+1} (Ma^2 - 1), & x < 0.2 \\ 1, & x \geq 0.2 \end{cases}, \quad (3.44)$$

and finally the initial x -velocity is

$$u_1 = \begin{cases} Ma\sqrt{\gamma} \left(1 - \frac{1}{\rho}\right), & x < 0.2 \\ 0, & x \geq 0.2 \end{cases}. \quad (3.45)$$

The shock Mach number is taken as $Ma = 2$, and the downstream density is set to $\rho_D = 3$. In Figure 3.2 and Figure 3.3, the density fields around the interface are shown at the final time of $t = 3.33$ on grids with spacing $\Delta = 1/512$ as produced by the radius $R = 2$ and $R = 3$ schemes, respectively. In both cases the SSP(4,3,2) time integrator is used with tolerance $\delta_a = \delta_r = 10^{-3}$ and a maximum CFL of 1.0.

The density profiles shown in Figure 3.2 and Figure 3.3 qualitatively agree with the results presented in [86] despite the latter making extensive use of adaptive mesh

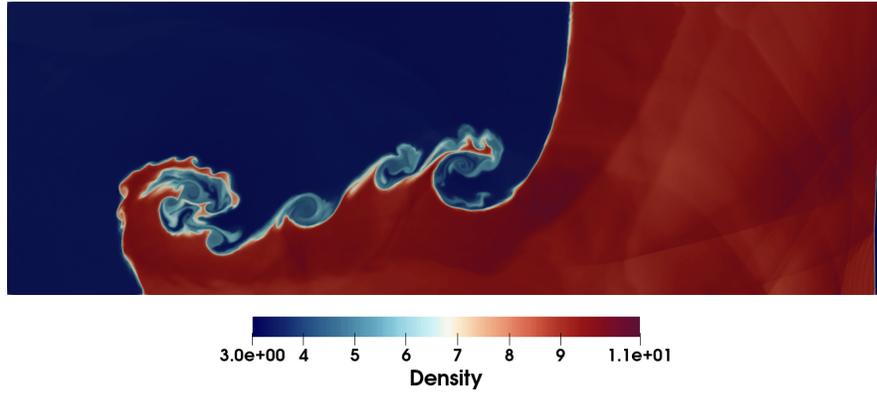


Figure 3.3: Shown is the density field for the Richtmyer-Meshkov instability at the final time of $t = 3.33$ as solved by the radius $R = 3$ scheme on a grid with spacing $\Delta = 1/512$. To highlight the interface the view has been zoomed into the region $[5/2, 11/2] \times [0, 1]$.

refinement to reach an effective grid resolution with spacing $\Delta = 1/2048$, compared to our results with $\Delta = 1/512$. The radius $R = 3$ results presented here exhibit smaller scale structures along the primary density interface separating the heavy and light fluids than are present in the radius $R = 2$ results. The density interfaces within the heavy fluid that arise from the reflections of passing shock are similar in all cases. Both results also pick up an additional Kelvin-Helmholtz instability along an internal interface below the primary one, more closely matching the referenced results despite using a grid with four times larger spacing.

This test also showcases the benefits of the KXRFCF style indicator described in Section 3.3. The cells flagged as needing WENO reconstruction in the final time step are shown in Figure 3.4, and comprise only 5.56% of the overall grid. This problem was run on an NVidia 4080 with the Kokkos profiling tools [106]. With the indicator



Figure 3.4: Shown in black are the cells flagged for WENO reconstruction at the final time for the radius $R = 2$ scheme. The view has been zoomed in to match Figure 3.2.

enabled and WENO applied only sparsely, the solver spent a total of 117 seconds doing linear reconstruction and 56 seconds doing WENO reconstruction over the entire evolution of the problem, which consisted of 21,193 right hand side evaluations. With the indicator disabled the solver always performs WENO reconstruction, which took 690 total seconds across 21,349 right hand side evaluations.

We note that the cost of activating KXRCF is small but not negligible. The average time to execute one right hand side evaluation (which includes all reconstruction, limitation, Riemann solves, etc.), with the indicator active was 0.044 seconds and without the indicator the average time was 0.050 seconds. The overhead in applying the indicator and doing a two-pass reconstruction is easily worthwhile, and would only become more useful in three dimensions or with larger stencils. Recall also that the indicator provides important memory savings, and for large problems it is useful regardless of the overhead in applying it.

3.6.4 Astrophysical jets

Astrophysical jets are excellent examples of extreme flows, often involving such great velocities that relativistic effects become relevant. Newtonian descriptions of such flows remain useful in their own right, and have slowly become a standard benchmark to demonstrate positivity preservation in schemes. The formulation of these benchmarks follows [2], just as the positivity preserving limiter presented in Section 3.4 did.

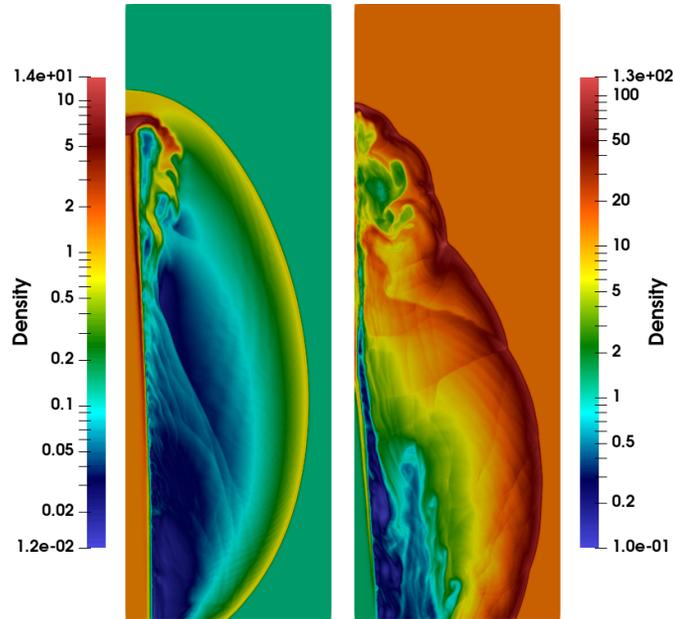


Figure 3.5: Shown are the logarithmic density fields for the high-density (left) and low density (right) astrophysical jets in two dimensions at the final times of $t = 0.002$ and $t = 0.04$ respectively. In both cases radius $R = 3$ stencils were used on grids with spacing $\Delta = 1/512$.

Two classes of astrophysical jets are considered, each having essentially the same setup. The domain is $\Omega = [0, 1/2] \times [0, 3/2]$, with reflecting conditions on the left, outflow conditions on the top and right sides, and mixed inflow/outflow conditions on

the bottom. The whole domain is initially filled with quiescent gas having a density of $\rho = \gamma$ and a pressure of $p = 1$. Inflow is prescribed along the bottom face where $x < 0.05$, and the two cases differ in terms of the in-flowing gas. A high density jet is formed by flowing in gas with a density of $\rho = 10\gamma$ and a pressure of $p = 1$ at a velocity of $u_2 = 800$, yielding a Mach 800 jet. A low density jet is similarly formed by flowing in gas with a density of $\rho = \gamma/10$ and a pressure of $p = 1$ at a velocity of $u_2 = 100$, yielding a Mach 100 jet.

Logarithmic plots of the density field at the final times of $t = 0.002$ and $t = 0.04$ for the high and low density jets respectively are shown in Figure 3.5. Each of these was obtained using a radius $R = 3$ stencil on a grid with spacing $\Delta = 1/512$ and the SSP(4,3) time integrator with tolerances of $\delta_a = \delta_r = 10^{-2}$.

Both cases exhibit densities that vary by three orders of magnitude. The high density jet yields a much simpler structure as the greater inertia of the in-flowing gas easily pushes the lower density ambient gas aside. This launches a bow shock that surrounds the jet, and the head of the jet slowly sloughs material off into the enclosed space. The low density jet has the opposite configuration, and yields a much richer structure. The low density in-flowing gas is immediately compressed by the ambient medium, narrowing to just a few cells at roughly the halfway point of its length. This compression and subsequent re-expansion launch multiple outgoing shocks that interact to generate the complicated cocoon surrounding the jet. In each case the jets yield similar mor-

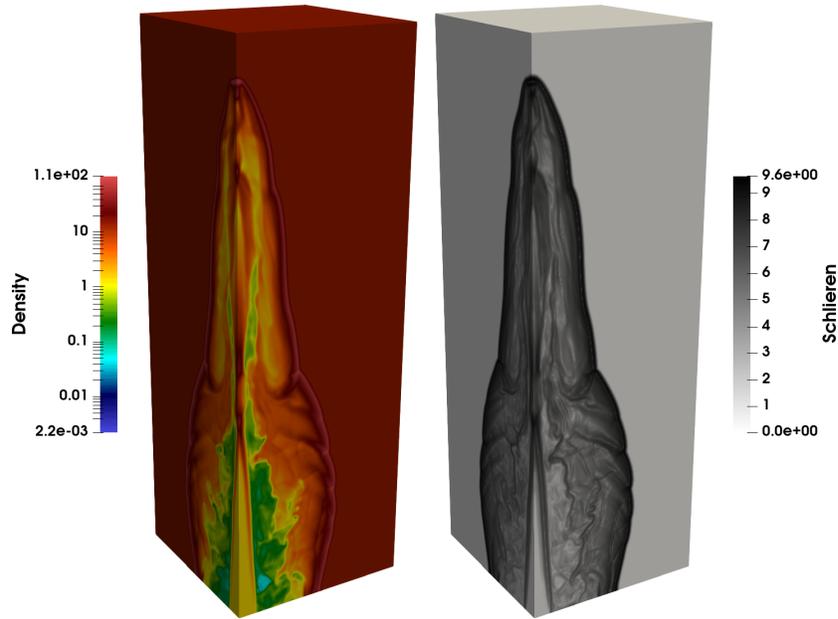


Figure 3.6: The left panel shows the logarithmic density field for the low-density astrophysical jet in three dimensions at the final time of $t = 0.035$ as solved by the radius $R = 2$ scheme on a grid with spacing $\Delta = 1/384$. The right panel shows the corresponding numerical Schlieren image defined as $\ln(1 + |\nabla\rho|)$.

phologies to those in [47, 2, 9], and succeed without generating any negative densities or pressures.

A three dimensional version of the low density jet is also considered. In this case the domain is set as $\Omega = [0, 1/2] \times [0, 3/2] \times [0, 1/2]$, outflow conditions are set on all of the upper faces, reflecting conditions are set on the lower x - and z - faces, and mixed inflow/outflow conditions are set on the lower y -face, with inflow now occurring where $\sqrt{x^2 + z^2} < 0.05$. The resulting logarithmic density field is shown alongside a numerical Schlieren image in Figure 3.6. These results were obtained with a radius $R = 2$ stencil on a grid with spacing $\Delta = 1/384$, and the same SSP(4,3,2) time integrator

as before. The overall structure is similar to the two dimensional results with the same compression and re-expansion process being visible, though the surrounding cocoon is more elongated. This is likely due to the compression process being more efficient in three dimensions. It would be interesting to add support for cylindrical coordinates in two dimensions and draw a comparison to those results. This would also be a prime candidate for adaptive mesh refinement considering that most of the domain remains occupied by the trivially resolvable ambient gas.

3.6.5 Taylor-Green vortex

A nearly incompressible Taylor-Green vortex is considered to introduce the Navier-Stokes equations and the treatment of physical viscosity. This problem has become a very valuable benchmark for high-order computational fluid dynamics codes seeking to resolve turbulent flows. The original Taylor-Green vortex was posed for incompressible solvers [103], but compressible versions have also been considered [42, 57]. Before stating the problem fully, the compressible Navier-Stokes equations need to be presented.

The compressible Euler equations in eq. (3.1) are now augmented to include viscosity as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot (\mathbf{F} - \mathbf{F}^{(v)}) = 0, \quad (3.46)$$

where the viscous fluxes $\mathbf{F}^{(v)}$ are given by

$$\mathbf{F}_j^{(v)} = \begin{pmatrix} 0 \\ \sigma_{ij} \\ q_j + \sigma_{jk}u_k \end{pmatrix}. \quad (3.47)$$

The stress tensor is given by

$$\sigma_{ij} = \frac{1}{Re} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3} (\nabla \cdot \mathbf{u}) \delta_{ij} \right), \quad (3.48)$$

where Re is the Reynolds number, and the dilatational terms related to the velocity divergence $\nabla \cdot \mathbf{u}$ are included. Finally, the heat flux is given by

$$q_j = \frac{1}{PrRe} \frac{\partial T}{\partial x_j}, \quad (3.49)$$

where $T = p/\rho$ is the temperature and Pr is the Prandtl number which is fixed to $Pr = 0.71$ throughout. Everything thus far has been presented in nondimensional form, as obtained from chosen reference scales for velocity U , length L , density $\hat{\rho}$, and kinematic viscosity ν . The Reynolds number is thus $Re = UL/\nu$, and the dimensional counterparts of the solution quantities (denoted by tildes) can be found as

$$\tilde{\rho} = \hat{\rho}\rho, \quad \tilde{\mathbf{u}} = U\mathbf{u}, \quad \tilde{p} = \hat{\rho}U^2p. \quad (3.50)$$

The compressible Taylor-Green vortex is posed on the triply periodic cube $\Omega =$

$[-\pi, \pi]^3$ with the initial conditions

$$\rho = 1 + \frac{\gamma Ma^2}{16} (\cos(2x) + \cos(2y)) (\cos(2z) + 2)$$

$$u_1 = \sin(x) \cos(y) \cos(z)$$

$$u_2 = -\cos(x) \sin(y) \cos(z)$$

$$u_3 = 0$$

$$p = \frac{1}{\gamma Ma^2} + \frac{1}{16} (\cos(2x) + \cos(2y)) (\cos(2z) + 2).$$

The velocity fields are the same as in the standard incompressible Taylor-Green vortex, as is the varying part of the pressure. The constant term in the pressure and the form of the density are chosen as in [42] to produce a constant temperature initial state at a Mach number of Ma . To make the flow nearly incompressible the Mach number is set to $Ma = 0.1$, producing a nearly constant density throughout.

This problem is solved using a radius $R = 2$ stencil on a grid with spacing $\Delta = \pi/192$, yielding a grid with 384^3 total cells, and the RK3(2)5F[3S₊^{*}] time integrator with tolerances of $\delta_a = \delta_r = 10^{-4}$. The present study is only concerned with a high-order representation of the inviscid fluxes. The physical viscosity is necessary for the solution to represent a single fixed Reynolds number independent of the grid resolution. At present, the viscous fluxes are incorporated simply through second-order accurate finite differences. Higher accuracy viscous fluxes, and more elaborate time integration schemes would be valuable directions for future study.

A snapshot of the flow can be seen in Figure 3.7 where an isosurface of the Q-

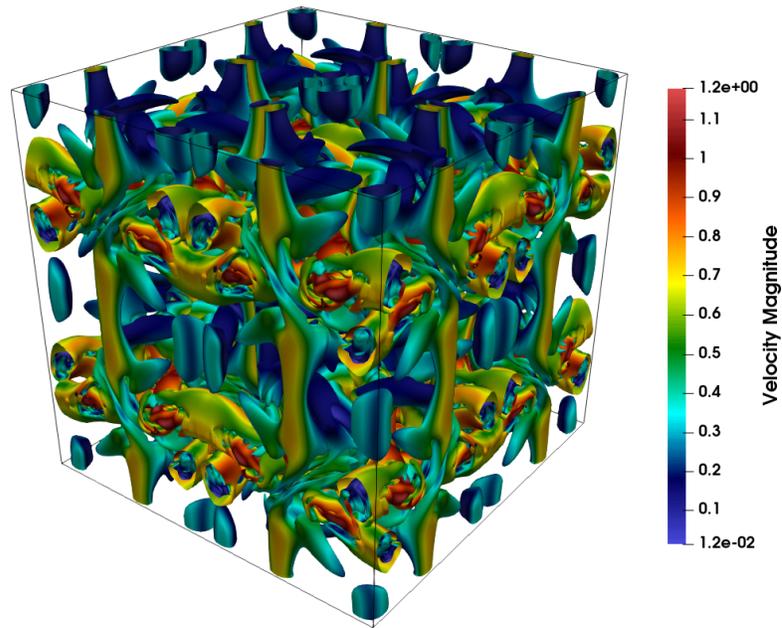


Figure 3.7: The $Q = 0.1$ isosurface of the Q-criterion colored by velocity magnitude is shown for the Taylor-Green vortex at nondimensional time $t = 8$ on a grid with spacing $\Delta = \pi/192$.

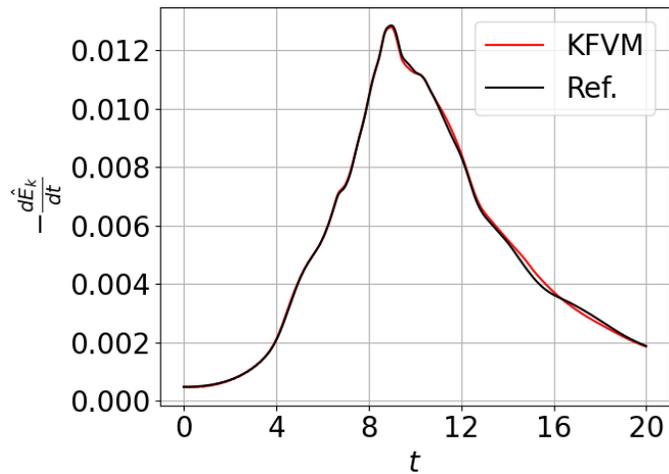


Figure 3.8: The evolution of the kinetic energy dissipation rate is compared against the fully converged reference data from [110].

criterion has been colored by the velocity magnitude, which can be directly compared to the results in [42]. A quantitative validation can be performed by tracking the total kinetic energy in the domain as a function of time. The rate of kinetic energy decay is shown in Figure 3.8 as compared to a fully resolved spectral solution from [57] using the method described in [109]. Excellent agreement can be seen with only minor deviations in the tail at long times. It is indeed impressive that a single code can resolve both nearly incompressible turbulence and extremely shock dominated flows as demonstrated in the previous section.

3.6.6 Rayleigh-Taylor instability

The final test is a two dimensional viscous Rayleigh-Taylor instability. The domain and initial conditions follow the setup from [94], though now an additional physical viscosity is added. The governing equations follow from the previous section (see eq. (3.46)) with an added source term in eq. (3.47).

Gravity is taken to point in the positive y -direction, and nondimensionalizing as before yields the source term

$$\mathbf{S} = \frac{1}{Fr^2} \left(0, 0, \rho, 0, \rho u_2 \right)^T, \quad (3.51)$$

where $Fr = \frac{U}{\sqrt{gL}}$ is the Froude number. We set the domain as $\Omega = [0, 1/4] \times [0, 1]$. Initially, the density is set to $\rho_{high} = 2$ for $y < 1/2$ and $\rho_{low} = 1$ otherwise, the pressure

is set as

$$p = \begin{cases} \frac{\rho_{high}}{Fr^2} y + 1, & y < 1/2 \\ \frac{1}{Fr^2} \left(y \rho_{low} - \frac{\rho_{high} - \rho_{low}}{2} \right) + 1, & y \geq 1/2 \end{cases},$$

and the ratio of specific heats is fixed at $\gamma = 5/3$. The Froude number is set as $Fr = 1$, the Prandtl number remains $Pr = 0.71$, and the Reynolds number is set to $Re = 20,000$.

The instability is seeded by a small vertical velocity perturbation given as

$$u_2 = -0.025 \sqrt{\frac{\gamma P}{\rho}} \cos(8\pi x). \quad (3.52)$$

Finally, the x -direction boundaries are supplied with reflecting conditions, and the y -direction boundaries are held fixed at the initial density and pressure with zero velocity.

The source terms are constant and contain only quantities for which cell averages are already available, namely the density and y -momentum, and the averaged source term $\langle \mathbf{S} \rangle$ is trivial to find for each cell. However, the implementation allows arbitrary user-provided source terms, so these gravitational sources are treated identically. As described in Section 3.4, internal states can also be reconstructed within each cell on a tensor-product Gauss-Legendre set of nodes. The source term is evaluated over these states and subsequently integrated.

The problem is evolved to the final time of $t = 2.5$ using the RK3(2)5 $_F$ [3 S^*_+] time integrator with tolerances $\delta_a = \delta_r = 10^{-3}$ and a maximum CFL number of 1.25. In Figure 3.9, we display the final density fields obtained with a radius $R = 2$ scheme on

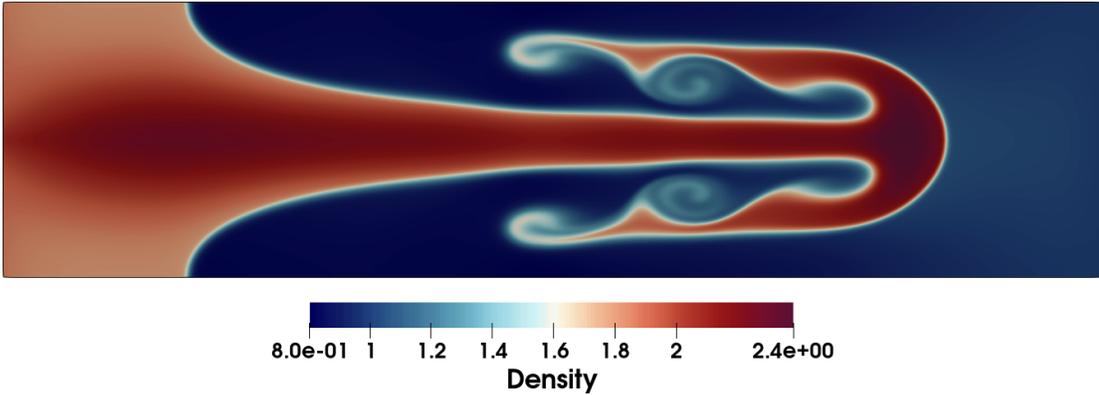


Figure 3.9: Shown is the density field for the viscous Rayleigh-Taylor instability at the final time of $t = 2.5$ with Reynolds number $Re = 20,000$ and Froude number $Fr = 1$. These results were obtained with the radius $R = 3$ scheme on a grid with spacing $\Delta = 1/1024$, though they are insensitive to both grid resolution and to the stencil radius. Note that the domain has been rotated 90° clockwise, such that the y -axis points to the right, for the sake of plotting.

a grid spacing of $\Delta = 1/1024$. Morphologically, with the explicit physical viscosity, a smooth leading cap is observed with no secondary Kelvin-Helmholtz type instabilities along its interface, and similarly there are no secondary instabilities along the central column. Secondary Kelvin-Helmholtz instabilities are visible on the inner region of the rising cap, and their structure is consistent over a range of grid resolutions. The two hooks (or arms) at the lowest part of the cap, as well as the position and the shape of the roll-up just above it, are converged and appear the same for all resolutions above $\Delta = 1/256$. The results are also consistent between the $R = 2$ and $R = 3$ schemes, with at most a 2% difference in density in corresponding cells.

However, the above convergent solution behavior is in opposition to results obtained from inviscid solvers that exclude explicit viscosity but rely only on numerical dissipa-

tion because, in that case, there is no agreeable solution for the method to converge to. Indeed finer and finer scale structures will appear each time the grid is refined or the numerical dissipation is lowered by using a larger stencil (see [94] for instance). Furthermore, methods with sufficiently low numerical dissipation are prone to breaking symmetry due to the non-associativity of floating point addition [34]. While fixes for this issue are available for dimension-by-dimension schemes, it remains unclear how one would avoid non-associativity errors in multidimensional reconstruction without drastically increasing the computational cost. Using a physical viscosity as done here avoids all of these problems by setting a single agreeable solution to converge to, and yields more scientifically meaningful results.

Chapter 4

A kernel-based finite volume method for ideal magnetohydrodynamics

4.1 Introduction

The compressible Euler equations from the previous chapter can be extended to handle conducting fluids with embedded magnetic fields. In the non-relativistic limit with vanishing magnetic resistivity the governing equations for *ideal* magnetohydrody-

namics (MHD) are

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ E \\ B_i \end{pmatrix} \quad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p_* \delta_{ij} - B_i B_j \\ u_j (E + p_*) + B_j B_k u_k \\ B_i u_j - B_j u_i \end{pmatrix}, \quad (4.1)$$

where ρ is the fluid density, u_i are the velocity components, E is the total energy, $p_* = p_g + \frac{1}{2} B_k B_k$ is the total pressure containing both gas and magnetic field contributions, and B_i are the components of the magnetic field. The gas (or thermal) pressure must be supplied by an equation of state. As in the previous chapter, the adiabatic equation of state for the gas pressure

$$p_g = (\gamma - 1) \left(E - \frac{1}{2} \rho u_k u_k - \frac{1}{2} B_k B_k \right), \quad (4.2)$$

is used with γ being the ratio of specific heats. The divergence of the magnetic field must vanish, $\nabla \cdot \mathbf{B} = \partial_k B_k = 0$, since there should be no magnetic monopoles¹. It is this requirement that makes the design of numerical methods for the ideal MHD equations significantly more challenging than for the Euler equations.

This chapter opens with a more complete discussion of the $\nabla \cdot \mathbf{B} = 0$ requirement, and a review of some prominent numerical methods for enforcing it in Section 4.2. Ultimately, the generalized Lagrange multipliers approach detailed in Section 4.2.4 will be selected for use throughout the remainder of the chapter. Section 4.3.1 concerns

¹The shorthand notation $\partial_k = \frac{\partial}{\partial x_k}$ is used here and throughout this chapter.

the adaptation of the KXRCF troubled cell indicator to the present setting. The linearized primitive reconstruction variables from Section 3.2.2 are recalled and adapted to the ideal MHD equations in Section 4.3.2. Positivity preservation remains an important topic for the ideal MHD equations, and the present approach is discussed in Section 4.3.3. The previous three sections regard direct adaptations of methods from the previous chapter to the ideal MHD equations. The approaches to produce (nearly) solenoidal magnetic fields selected from Section 4.2 involve complicated source terms. Discretizing these source terms while maintaining high-order accuracy is challenging and Section 4.4 is devoted to this task. An overview of the software supporting this work is presented in Section 4.5. Finally, a suite of informative and stringent benchmark problems are presented in Section 4.6.

4.2 Solenoidal magnetic fields

Magnetic monopoles are largely believed not to exist, and Gauss's law for magnetism asserts that the divergence of the magnetic field must vanish. For smooth solutions of the ideal MHD equations eq. (4.1) this constraint is naturally preserved as long as the initial magnetic field is divergence-free. The induction equation governs the evolution of the magnetic field, and was written above in divergence form. Isolating

this equation and writing it instead in curl form yields

$$\frac{\partial \mathbf{B}}{\partial t} + \nabla \times (\mathbf{B} \times \mathbf{u}) = 0. \quad (4.3)$$

Subsequently taking the divergence of this equation, and recalling that the divergence of the curl is zero, yields

$$\frac{\partial(\nabla \cdot \mathbf{B})}{\partial t} = 0. \quad (4.4)$$

Thus, the divergence of the magnetic field never changes and will remain zero as long as it is initially zero. This means that the divergence-free constraint is naturally preserved by the governing equations, making this constraint an *involution* of eq. (4.1).

At first glance, this constraint seems similar to the imposition of divergence-free velocity fields in the incompressible Euler and Navier-Stokes equations. A fundamental difference is that the incompressibility constraint is coupled dynamically to the pressure in these systems, and unfortunately many numerical approaches from the incompressible fluid dynamics community do not directly carry over to the present setting. The following subsections briefly review several popular approaches to handling the $\nabla \cdot \mathbf{B} = 0$ constraint, and follow loosely from [105]. The projection method described in Section 4.2.1 and the constrained transport methods described in Section 4.2.2 are often said to satisfy the divergence-free constraint exactly, or up to floating point round-off. It should be noted from the outset that this is a minor abuse of nomenclature and that these schemes enforce the constraint for a particular *discretization* of the divergence.

4.2.1 The projection method

One method from the incompressible fluid dynamics community that does carry over to the ideal MHD equations is the projection method, which was first used in this setting by Brackbill and Barnes [10]. In projection type methods the conservation laws in eq. (4.1) are evolved forward by any appropriate scheme irrespective of the divergence-free constraint. At each new time step (or stage in Runge-Kutta schemes) the updated magnetic field is unlikely to be divergence-free. Denoting this tentative field by \mathbf{B}' , a correcting field can be found by solving a corresponding Poisson equation

$$\nabla^2\phi = -\nabla \cdot \mathbf{B}', \quad (4.5)$$

and replacing the updated magnetic field by

$$\mathbf{B} = \mathbf{B}' + \nabla\phi. \quad (4.6)$$

This serves to project the tentative field \mathbf{B}' onto the nearest divergence-free field. It should be noted that the discretizations for the divergence in eq. (4.5) and the gradient in eq. (4.6) must be compatible with the discrete Laplacian in eq. (4.5). It is with respect to these discrete operators that the magnetic field will be divergence-free (assuming that eq. (4.5) is solved exactly).

This process does not conserve magnetic energy (i.e. $\|\mathbf{B}'(\mathbf{x})\| \neq \|\mathbf{B}(\mathbf{x})\|$ in general). The total energy E is evolved as a separate independent quantity, and there are two ways to incorporate this changed magnetic energy. The value for E obtained from the

hyperbolic scheme can be kept as is, in which case the now modified magnetic energy induces a change in the internal energy. Alternatively, conservation of the total energy can be sacrificed by updating E alongside \mathbf{B} to maintain the original internal energy.

Typically the linear system arising from eq. (4.5) would be solved inexactly by an iterative method (e.g. preconditioned conjugate gradients). Such methods have become highly advanced, and the projection method may well be a competitive option for problems of moderate to large size, particularly when solved on CPU based systems. However, this Poisson solve does form an expensive bottleneck that couples all unknowns together in an otherwise fully explicit solver. Ultimately the projection method has seen little use in recent years, at least within solvers aimed at shock dominated problems.

4.2.2 Constrained transport

The constrained transport (CT) method, introduced by Evans and Hawley in [27], seeks to maintain divergence-free magnetic fields in a way that is arguably closer to the continuum partial differential equations. Recall from eq. (4.4) that an initially divergence-free field will remain divergence-free for all time. Constrained transport seeks a spatial discretization of the induction equation such that a subsequent discrete divergence vanishes identically. In this way CT maintains the (discrete) divergence-free nature of the magnetic field as long as the initial field has zero (discrete) di-

vergence. The CT method has seen wide use and intense development over time [40, 41, 101, 64, 102].

The CT method is most naturally posed on a staggered grid. In three dimensions, the hydrodynamic variables (density, momentum, and total energy) are stored at cell centers, and the magnetic field components are stored on cell faces with matching normal directions (B_x on x -faces, B_y on y -faces, and B_z on z -faces). Integrating the induction equation over one face and applying Stokes' theorem then yields the update to the magnetic flux on that face as a contour integral of the electric field defined on cell edges. A fundamental operation in CT schemes is the production of properly up-winded electric fields on these edges. Subsequently discretizing this contour integral then preserves a (discrete) divergence-free condition on the magnetic fields.

The use of a staggered grid is one major hurdle towards the use of these methods. Data storage becomes more complicated than for schemes where all unknowns are co-located. Staggered grids are also challenging to combine with adaptive mesh refinement, and numerical artifacts in the magnetic field may be seen along coarse-fine interfaces therein. It was also shown that second order schemes could be recast into forms that do not rely on having a staggered grid [105]. More modern CT schemes with higher order reconstructions remain posed on staggered grids [91].

4.2.3 Eight-wave formulation

A fundamentally different approach to satisfying the divergence-free constraint is to enforce it only to the level of truncation error, rather than exactly for some given discretization. The eight wave formulation is the earliest such approach, and was presented independently by Godunov in [45] and by Powell in [78]. This approach adds a source term proportional to $\nabla \cdot \mathbf{B}$ to the governing equations that serves to advect divergence errors, which prevents them from compounding in any one location.

Powell arrived at this source term by deriving the governing equations for ideal MHD without the assumption that $\nabla \cdot \mathbf{B} = 0$. Godunov produced this source by noting that it is necessary to make the governing equations symmetrizable in the sense discussed in the introductory chapter. With the so-called Godunov-Powell source term included, the governing equations become

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ E \\ B_i \end{pmatrix} \quad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p_* \delta_{ij} - B_i B_j \\ u_j (E + p_*) + B_j B_k u_k \\ B_i u_j - B_j u_i \end{pmatrix} \quad \mathbf{S} = \begin{pmatrix} 0 \\ -B_i \partial_k B_k \\ -(u_k B_k) \partial_k B_k \\ -u_i \partial_k B_k \end{pmatrix}. \quad (4.7)$$

Notably, whenever $\nabla \cdot \mathbf{B} = 0$ does hold these equations reduce to the standard ideal MHD equations in eq. (4.1).

As in the outset of this section, the induction equation can again be isolated and the evolution of $\nabla \cdot \mathbf{B}$ determined. The fluxes \mathbf{F}_j , arising from the curl of the electric field

still do not contribute. Hence the divergence evolves as

$$\frac{\partial(\nabla \cdot \mathbf{B})}{\partial t} + \nabla \cdot (\mathbf{u}(\nabla \cdot \mathbf{B})) = 0, \quad (4.8)$$

supporting the claim that the eight wave formulation advects divergence errors along the flow.

The presence of this source term means that the equations are no longer conservative. Solutions with shocks may fail to satisfy the Rankine-Hugoniot conditions. Notably, the previous conclusion around the advection of $\nabla \cdot \mathbf{B}$ only relies on the source term in the induction equation, and is completely independent of the source terms present in the momentum and energy equations. An alternative formulation given by Janhunen in [58] omits the source terms in the momentum and energy equations which keeps them fully conservative.

4.2.4 Generalized Lagrange multipliers

The method of generalized Lagrange multipliers (GLM) arises as an extension of the eight wave formulation. A notable issue in the eight wave formulation, irrespective of the particular source term selected, is that the divergence errors can only be damped by the inherent numerical dissipation of the numerical scheme. More problematic is that the divergence errors can accumulate at stagnation points of the flow, and hence grow beyond the scale of truncation error.

The goal of GLM methods is to find a formulation that induces a more favorable

evolution equation for $\nabla \cdot \mathbf{B}$ than eq. (4.8). This approach was first proposed for use in ideal MHD by Dedner in [22], and consists of appending a new scalar field ψ to the governing equations. A variety of different evolution equations for ψ can be used, each of which induce a different evolution equation on $\nabla \cdot \mathbf{B}$.

More recently the GLM approach has been re-derived by Derigs et al. in [23] to be both Galilean invariant and consistent with the second law of thermodynamics. An extensive review and analysis of many different GLM schemes can be found therein.

The resulting equations are

$$\mathbf{U} = \begin{pmatrix} \rho \\ \rho u_i \\ E \\ B_i \\ \psi \end{pmatrix} \quad \mathbf{F}_j = \begin{pmatrix} \rho u_j \\ \rho u_i u_j + p_* \delta_{ij} - B_i B_j \\ F_j^E \\ B_i u_j - B_j u_i + C_h \psi \delta_{ij} \\ C_h B_j \end{pmatrix} \quad \mathbf{S} = \begin{pmatrix} 0 \\ -B_i \partial_k B_k \\ -S^E \\ -u_i \partial_k B_k \\ -u_k \partial_k \psi \end{pmatrix}, \quad (4.9)$$

where the flux of total energy in the j^{th} direction is $F_j^E = u_j(E + p_* - \frac{1}{2}\psi^2) + B_j(C_h \psi - u_k B_k)$, and the source term for the total energy is $S^E = (u_l B_l) \partial_k B_k + \psi u_k \partial_k \psi$. The constant C_h is the hyperbolic cleaning speed which will be discussed below. As an ansatz the total energy is now defined to include a contribution from the ψ field. The adiabatic equation of state for the gas pressure thus becomes

$$p_g = (\gamma - 1) \left(E - \frac{1}{2} \rho u_k u_k - \frac{1}{2} B_k B_k - \frac{1}{2} \psi^2 \right). \quad (4.10)$$

Following [23], the wavespeeds arising from eq. (4.9) in the x -direction are

$$\left\{ \lambda_f^{(\pm)} = u_1 \pm C_f, \lambda_s^{(\pm)} = u_1 \pm C_s, \lambda_a^{(\pm)} = u_1 \pm C_a, \lambda_h^{(\pm)} = u_1 \pm C_h, \lambda_E = u_1 \right\}, \quad (4.11)$$

where the (\pm) superscripts indicate paired wavespeeds. Of course, similar wavespeeds appear in other directions. Denoting the acoustic speed as $a = \sqrt{\gamma p / \rho}$ and the density reduced magnetic field by $\mathbf{b} = \mathbf{B} / \sqrt{\rho}$, the fast and slow magnetosonic speeds are given by

$$C_{f,s}^2 = \frac{1}{2} \left(a^2 + \|\mathbf{b}\|^2 \pm \sqrt{(a^2 + \|\mathbf{b}\|^2)^2 - 4a^2 b_1^2} \right), \quad (4.12)$$

with plus and minus yielding the fast and slow speeds respectively. The Alfvén speed is $C_a = \sqrt{b_1}$. Notably, the first six wavespeeds in eq. (4.11) are the same as in the standard ideal MHD equations, as is the entropic speed λ_E . The crucial difference here is the symmetric appearance of λ_h arising from the cleaning speed.

Note that eq. (4.9) reduces to the eight wave formulation in eq. (4.7) by setting $C_h = 0$ and initializing the ψ field to zero. In this case the cleaning waves associated with $\lambda_h^{(\pm)}$ merge to yield the so-called divergence wave. As discussed in [23], increasing C_h from zero yields a splitting of this wave into the cleaning waves. The improved divergence cleaning abilities of GLM methods over the eight wave method can now be appreciated. With non-zero C_h at least one of the paired speeds $\lambda_h^{(\pm)}$ must also be non-zero, and cleaning still occurs even around stagnation points of the flow.

The hyperbolic cleaning speed C_h determines how quickly divergence errors are re-

moved, so choosing a large speed is helpful to constrain the magnetic field divergence more strongly. On the other hand, faster cleaning speeds can artificially limit the allowable time step sizes if the corresponding wavespeeds dominate over the remaining speeds. As in [23], the cleaning speed is determined by first calculating the maximal wavespeed, λ_{max} , and componentwise maximal (absolute) velocity, v_{max} , over all Riemann states in the domain, and setting

$$C_h = \lambda_{max} - v_{max}. \quad (4.13)$$

This ensures that the cleaning speed is as large as possible without unduly limiting the allowable time step size.

4.3 KXRCF indicators, reconstruction variables, and positivity

The KXRCF indicators discussed in Section 3.3, the linearized primitive variables presented in Section 3.2.2, and the positivity preserving limiter from Section 3.4, can all be adapted to the ideal MHD equations directly. In fact, MHD was one of the motivations for developing the linearized primitive variables in the first place. As the purposes and motivations behind these components are the same as before, each will be presented briefly in the following sections.

4.3.1 KXRCF troubled cell indicator

The troubled cell indicators introduced in Section 3.3 and adapted from the discontinuous Galerkin community [63, 39], can similarly be adapted to the ideal MHD equations. These indicators operate by measuring the jump of some quantity across cell interfaces relative to its value calculated from the cell-average. Specifically, let s be an index ranging over all quadrature points on all faces of a given cell, and let $q_s^{(\pm)}$ be the values of some quantity at the s^{th} quadrature on the inside and outside of the cell. Cells are flagged as troubled if

$$\frac{\max_s \left\{ \left| q_s^{(+)} - q_s^{(-)} \right| \right\}}{q(\tilde{\mathbf{U}})} > \Delta^{3/2}, \quad (4.14)$$

where $\tilde{\mathbf{U}} = \langle \mathbf{U} \rangle_h$ is the cell average state in the cell where the indicator is being calculated.

The utility of this indicator is reliant on the choice of the indicator variable q . For the Euler equations this was chosen to be the entropy, $q = p/\rho^\gamma$, which was justified in that it jumps across both shocks and contact discontinuities. The ideal MHD-GLM equations provide a substantially more complicated setting than the compressible Euler equations. The thermodynamic entropy remains a useful indicator variable, but as will be seen, it may not always flag cells adequately. The entropy is still used as the indicator variable throughout this chapter, but this is a topic that would be an interesting avenue for future study.

4.3.2 Reconstruction variables

The linearized primitive variables from Section 3.2.2 are directly applicable to the ideal MHD-GLM equations. In fact, the density, magnetic field, and ψ field, all act as both conservative and primitive variables. The linear momentum components yield the velocity components simply by dividing by the density. The final primitive variable is the gas pressure, given by eq. (4.10), which replaces the total energy. The transformation matrix arising from the Jacobian is

$$\Phi = \left. \frac{\partial \mathbf{V}}{\partial \mathbf{U}} \right|_{\tilde{\mathbf{U}}} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\tilde{u}_1/\tilde{\rho} & 1/\tilde{\rho} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -\tilde{u}_2/\tilde{\rho} & 0 & 1/\tilde{\rho} & 0 & 0 & 0 & 0 & 0 & 0 \\ -\tilde{u}_3/\tilde{\rho} & 0 & 0 & 1/\tilde{\rho} & 0 & 0 & 0 & 0 & 0 \\ - & - & - & - & \Phi_E & - & - & - & - \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (4.15)$$

where the fifth row regarding the map to the gas pressure is given by

$$\Phi_E = (\gamma - 1) \begin{pmatrix} ||\tilde{\mathbf{u}}||^2 & -\tilde{u}_1 & -\tilde{u}_2 & -\tilde{u}_3 & 1 & -\tilde{B}_1 & -\tilde{B}_2 & -\tilde{B}_3 & -\tilde{\psi} \end{pmatrix}. \quad (4.16)$$

As before, the quantities denoted by tildes are obtained from the reference state where this transformation matrix is to be calculated (e.g. $\tilde{u}_i = \tilde{\rho} u_i / \tilde{\rho}$). It is also worth noting

that the vast majority of this matrix contains zeros, and applying the transformation can be done much more cheaply by not actually constructing the matrix. Similarly, the inverse transformation matrix is

$$\Phi^{-1} = \frac{\partial \mathbf{U}}{\partial \mathbf{V}} \Big|_{\mathbf{V}(\tilde{\mathbf{v}})} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \tilde{u}_1 & \tilde{\rho} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \tilde{u}_2 & 0 & \tilde{\rho} & 0 & 0 & 0 & 0 & 0 & 0 \\ \tilde{u}_3 & 0 & 0 & \tilde{\rho} & 0 & 0 & 0 & 0 & 0 \\ \|\tilde{\mathbf{u}}\|^2 & \widetilde{\rho u_1} & \widetilde{\rho u_2} & \widetilde{\rho u_3} & 1/(\gamma-1) & \widetilde{B_1} & \widetilde{B_2} & \widetilde{B_3} & \tilde{\psi} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad (4.17)$$

and once again the transformation is much easier to apply by not actually constructing the matrix. These two transformation matrices fully define the linearized primitive variables, and the remainder of Section 3.2.2 applies as is.

4.3.3 Positivity preservation

Positivity preservation proceeds for the ideal MHD-GLM equations in much the same fashion as for the Euler equations in Section 3.4. Fix attention to a single cell where the positivity preserving limiter is being applied, and let the grid indices (i, j, k)

be defined locally. Recall that bounds on the fluid density and gas pressure are set by gathering the cell-averaged values from nearby cells as

$$\tilde{\rho}_{max} = \max_{-1 \leq i,j,k \leq 1} \{ \langle \rho \rangle_{i,j,k} \}, \quad (4.18)$$

$$\tilde{\rho}_{min} = \min_{-1 \leq i,j,k \leq 1} \{ \langle \rho \rangle_{i,j,k} \}, \quad (4.19)$$

$$\tilde{p}_{min} = \min_{-1 \leq i,j,k \leq 1} \{ \langle p \rangle_{i,j,k} \}, \quad (4.20)$$

and subsequently widening the bounds to

$$\rho_{max} = \tilde{\rho}_{max} (1 + \kappa_2 - \kappa_2 \eta), \quad (4.21)$$

$$\rho_{min} = \tilde{\rho}_{min} (1 - \kappa_2 + \kappa_2 \eta), \quad (4.22)$$

$$p_{min} = \tilde{p}_{min} (1 - \kappa_2 + \kappa_2 \eta), \quad (4.23)$$

to avoid clipping smooth extrema. The pressure can be naively calculated to second order by directly using the cell-averaged conservative states since it only needs to provide a reasonable floor on the locally allowed pressures. Here η is a flattener variable defined as

$$\eta = \min \left\{ 1, \max \left\{ 0, -\frac{\kappa_1 \tilde{a}_{min} + (\nabla \cdot \bar{\mathbf{u}}) \Delta}{\kappa_1 \tilde{a}_{min}} \right\} \right\}, \quad (4.24)$$

where as before $\tilde{a} = \sqrt{\gamma \tilde{p} / \tilde{\rho}}$ is the acoustic speed appropriately minimized over the immediate neighbors of the cell being limited. Finally, the constants κ_1 and κ_2 are both fixed at 0.3 just as in [2].

Let s be an index running over all quadrature points on all faces of the cell in consideration, as well as all of the internal quadrature points for treating source terms,

such that $\{\mathbf{U}_s\}$ is the set of all reconstructed states associated with this cell. Density is corrected to lie in the desired bounds by finding

$$\theta_\rho = \min_s \left\{ 1, \frac{\langle \rho \rangle_0 - \rho_{min}}{\langle \rho \rangle_0 - \rho_s}, \frac{\rho_{max} - \langle \rho \rangle_0}{\rho_s - \langle \rho \rangle_0} \right\}, \quad (4.25)$$

and updating the states as

$$\mathbf{U}_s \leftarrow \langle \mathbf{U} \rangle_0 + \theta_\rho (\mathbf{U}_s - \langle \mathbf{U} \rangle_0), \quad \forall s. \quad (4.26)$$

While density is constrained in precisely the same way as before, the constraint on gas pressure is now more complicated. The correction to the states remains

$$\mathbf{U}_s \leftarrow \langle \mathbf{U} \rangle_0 + \theta_p (\mathbf{U}_s - \langle \mathbf{U} \rangle_0), \quad \forall s, \quad (4.27)$$

and the complication arises in the calculation of an appropriate value for θ_p . The gas pressure is again a function of the conservative state, now through eq. (4.10). Fixing a single quadrature point s one finds that

$$p_g(\langle \mathbf{U} \rangle_0 + \theta_{p;s} (\mathbf{U}_s - \langle \mathbf{U} \rangle_0)) = p_{min}, \quad (4.28)$$

must hold with $\theta_{p;s}$ being the local correction for just the s^{th} state. Contrary to eq. (3.28), this equation is now a cubic in $\theta_{p;s}$ rather than a quadratic. Each correction factor is found as a root of eq. (4.28) using a bracketed bisection search in the same fashion as presented in [2]. One minor difference is that 10 iterations are always taken, and the lower bound on $\theta_{p;s}$ at the final iteration is returned. This may leave $\theta_{p;s}$ at a slightly

smaller value than necessary, but guarantees that a valid state will be formed. Finally, the smallest valid root is taken over all such correction factors

$$\theta_p = \min_s \{ \theta_{p;s} \mid 0 \leq \theta_{p;s} \leq 1 \}, \quad (4.29)$$

so that eq. (4.27) can be applied uniformly to all states associated with a particular cell. Note further that, as in Section 3.4, if all states associated to a given cell are already valid then no corrections or limitations will be applied.

4.4 Treatment of GLM source terms

As has been seen, extending the hydrodynamics solver from Chapter 3 is mostly straightforward. The troubled cell indicators, linearized primitive variables, and positivity preserving limiter all have direct analogs in the MHD system. The major change in the MHD case is the presence of the GLM source term in eq. (4.9) that depends both on the state \mathbf{U} and on its spatial derivatives. The source term can further be decomposed into terms involving the field divergence $\nabla \cdot \mathbf{B}$ and terms involving $\nabla \psi$. With this splitting the source term becomes $\mathbf{S} = (\nabla \cdot \mathbf{B}) \mathbf{S}_B + (\mathbf{u} \cdot \nabla \psi) \mathbf{S}_\psi$ with

$$\mathbf{S}_B = - \left(\begin{array}{cccccccc} 0 & B_1 & B_2 & B_3 & (\mathbf{u} \cdot \mathbf{B}) & u_1 & u_2 & u_3 & 0 \end{array} \right)^T, \quad (4.30)$$

$$\mathbf{S}_\psi = - \left(\begin{array}{cccccccc} 0 & 0 & 0 & 0 & \psi & 0 & 0 & 0 & 1 \end{array} \right)^T. \quad (4.31)$$

Derivatives of the \mathbf{B} and ψ fields are necessary for the calculation of these terms, and producing these derivatives to sufficient accuracy while also maintaining a stable

scheme presents a distinct challenge. Ultimately the source term will be integrated over the cell with a tensor-product Gauss-Legendre rule, hence the states and their derivatives must be known at these points interior to the cell. The internal states are easy to generate in tandem with all of the Riemann states.

One could reconstruct the derivatives directly as was already done in forming the smoothness indicators (consult eq. (2.47)). On the other hand, the need for these derivatives is specific to the ideal MHD-GLM equations, and is limited to only a few fields therein. Alternative approaches for each type of term are examined in the following two subsections.

4.4.1 Terms involving field divergence

Ultimately the source terms must be integrated over the cell volume, and initially one could observe that integrating the magnetic field divergence over a cell is particularly simple. The divergence theorem simply converts this to a surface integral of the normally aligned field components. Of course, the source term includes more than just $\nabla \cdot \mathbf{B}$, so the volume integral of \mathcal{S}_B does not transform in this way.

Given a smooth divergence-free field, a single pass of reconstruction will give a field that is nearly divergence-free. A large simplification can be obtained by replacing

the source term volume integral by the approximation

$$\langle (\nabla \cdot \mathbf{B}) \mathbf{S}_B \rangle_h = \int_{\Omega_h} (\nabla \cdot \mathbf{B}) \mathbf{S}_B d\mathbf{x} \quad (4.32)$$

$$\approx \left(\int_{\Omega_h} \nabla \cdot \mathbf{B} d\mathbf{x} \right) \left(\int_{\Omega_h} \mathbf{S}_B d\mathbf{x} \right) = \langle \mathbf{S}_B \rangle_h \int_{\partial\Omega_h} \mathbf{B} \cdot d\mathbf{S}. \quad (4.33)$$

Splitting the initial volume integral into a product of integrals is precisely the type of operation that leads to degradation of accuracy when naively calculating averaged primitive variables from averaged conservative variables. However, for smooth fields the divergence should always be of the same order as the reconstruction. The claim that this splitting does not limit the achievable order of accuracy is validated experimentally in Section 4.6.1

The integral of \mathbf{S}_B on the far right side of section 4.4.1 is found through a standard tensor-product Gauss-Legendre rule through the cell interior. The surface integral of the field that has replaced the volume integral of its divergence is simpler in that it does not require any derivatives of \mathbf{B} . Furthermore it is easy to calculate using the available Riemann states, and it has been shown for the eight wave formulation that the $\nabla \cdot \mathbf{B}$ terms must be appropriately upwinded for the scheme to remain positive [113]. Rewriting this volume integral as a surface integral over the Riemann states allows this upwinding to be achieved in the same fashion as in [113]. To this end, the normal field components are obtained by averaging between the inner and outer Riemann states at each quadrature point on the surface.

4.4.2 Terms involving $\nabla\psi$

There is no such simple way to rewrite the terms involving derivatives of the ψ field, so they still must be found in the interior of the cell. This approach does fail to endow these terms with the same upwinding present in the $\nabla \cdot \mathbf{B}$ terms, though this only raises a small issue in one test below. The reconstruction step has produced values of ψ on the faces of the cell as part of the Riemann states, and at all of the interior quadrature points. These sets of points are illustrated in Figure 4.1 for a two dimensional case.

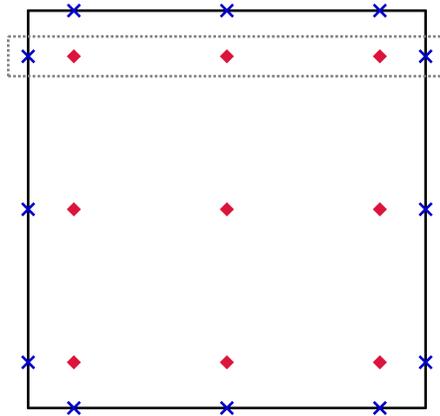


Figure 4.1: The quadrature points associated to a single cell in 2 dimensions are shown with $N = 3$ points per direction. The blue crosses indicate the points on the cell boundary where the Riemann states are located, and the red diamonds indicate the points in the interior. The dashed box highlights how one line of N interior Gauss-Legendre points can be augmented by two boundary points to give $N + 2$ Gauss-Legendre-Lobatto points. Similar groupings hold throughout the cell.

Consider isolating a line of such points through the interior, and adjoining the nearest two boundary points (see the dashed box in Figure 4.1). The N point Gauss-Legendre (GL) points in the interior with these boundary points thus give $(N + 2)$ Gauss-Legendre-Lobatto (GLL) points. A differentiation matrix mapping from the

$(N + 2)$ GLL points to the N GL points gives precisely the information needed (see for instance Section 5.5.1 of [43]). Let $\{x_i\}_{i=0}^{N+1}$ be the set of GLL nodes on the reference $[-1, 1]$ interval such that the restriction to $1 \leq i \leq N$ gives the embedded GL nodes. The $(N + 2) \times (N + 2)$ GLL Vandermonde matrix \mathbf{V} has the entries

$$V_{i+1,j+1} = x_i^j, \quad 0 \leq i, j \leq N + 1, \quad (4.34)$$

noting that the indices have been shifted by one to move from zero-based indexing to one-based indexing. Inverting this gives a map from values of some function on the GLL nodes to the coefficients of its polynomial interpolant in the monomial basis. This interpolant can trivially be differentiated, and mapped to the GL nodes through the $N \times (N + 2)$ matrix \mathbf{H} with entries

$$H_{i,j+1} = jx_i^{j-1}, \quad 1 \leq i \leq N, \quad 0 \leq j \leq N + 1. \quad (4.35)$$

Together these form the differentiation matrix \mathbf{D} that maps values of a function on the GLL points to values of its derivative on the GL points as

$$\mathbf{D} = \frac{\Delta}{2} \mathbf{H} \mathbf{V}^{-1}, \quad (4.36)$$

where the pre-factor is necessary to pull back from the reference $[-1, 1]$ interval to the cell in consideration. Note that the same differentiation matrix applies regardless of the direction of the slice in Figure 4.1.

All of the necessary derivatives of ψ in the interior are thus obtained by slicing all values across a line of nodes, adjoining the boundary states, and applying the differenti-

ation matrix \mathbf{D} . With these derivatives known the full term $(\mathbf{u} \cdot \nabla \psi) \mathcal{S}_\psi$ can be calculated at every interior quadrature point, and summed to yield the cell-averaged source term $\langle (\mathbf{u} \cdot \nabla \psi) \mathcal{S}_\psi \rangle_h$. While the overall scheme presented in this dissertation is only applied to uniform Cartesian grids, care has been taken to leave extensions of the method to unstructured grids as a possibility. Discretizing this source term in this fashion does limit future applications to grids with quadrilateral cells in two dimensions and hexahedral cells in three dimensions.

4.5 Implementation details

A highly parallel multi-GPU code was written by the author to support the results in this chapter and the previous chapter, and has been made publicly available under a BSD 3 clause license [72]. Here, a brief overview of the code capabilities and design decisions is given. Shared memory parallelism is achieved through the Kokkos library [107, 26], and distributed memory parallelism is achieved through the message passing interface (MPI). While the use of MPI for parallelism is standard and familiar to many, Kokkos is quite new and will likely be unfamiliar. A short description of Kokkos and how it is used within the code supporting this research is given in the next section. Following this, a breakdown of all steps the solver goes through is presented.

4.5.1 The Kokkos library

Kokkos is a performance portability library that provides useful data structures, such as multidimensional arrays and unordered maps, and a set of abstractions for launching work in parallel over these data structures. A crucial observation is that with naive storage ordering many scientific workloads can easily become bottlenecked by the time spent fetching data from main memory. Unfortunately, different parallel architectures benefit from different storage patterns, and no one layout (e.g. row-major, column-major, or tiled layouts of multidimensional arrays) can possibly work well on all architectures.

The key innovation of Kokkos is to tie the storage layouts directly to the routines for launching parallel work, and perhaps more importantly, to provide the end programmer with a tidy interface to manage without needing to deal with low-level complexity. Moreover, the parallel work loads do not require any vendor-specific implementations from the end user. All results presented in this dissertation were run on NVidia GPUs, but the available code [72] does not contain any CUDA.

Another aspect of the implementation that is supported by Kokkos is worth deliberating on. The calculation of the smoothness indicators, eq. (2.46), and the reconstruction to the various quadrature points from each substencil, eq. (2.40), can both be done more efficiently with batched linear algebra subroutine calls. This allows all derivatives needed to form a single β_q to be calculated simultaneously. Similarly, by

gathering the reconstruction vectors $\mathbf{r}_{(q)}$ for each quadrature point together all Riemann states can be simultaneously produced for one substencil. In each case this is as simple as performing a small matrix-matrix product or matrix-vector product respectively (see the implementation for more details). This batching process is supported by the Kokkos-Kernels library [106].

4.5.2 Overview of the solver

The overall run of the solver proceeds in three main stages: (i) construction of the stencils and reconstruction vectors, (ii) evaluation of the initial conditions into cell average quantities, and (iii) advancing the cell averages through time. The first two stages are only performed once as setup for the run. The final stage is the most complex and forms the vast majority of the work.

Constructing the (sub)stencils and reconstruction vectors entails three steps:

1. Enumerate all cells in the full stencil, \mathcal{S}_0 , and the remaining substencils, \mathcal{S}_q , $q = 1, \dots, N_S$.
2. Form reconstruction vectors, $\mathbf{r}_{(q)}$, for each face and internal quadrature point \mathbf{x}_s relative to each (sub)stencil using eq. (2.37). The quadrature points on each face and within the cell are set using a tensor product Gauss-Legendre quadrature rule with R points per dimension, yielding $2R^{\text{th}}$ order accuracy. This results in $2dR^{d-1}$ reconstruction vectors associated with the faces, and R^d associated with

the interior.

3. Form reconstruction vectors for all partial derivatives needed to evaluate the smoothness indicators in eq. (2.46) relative to each (sub)stencil.

For MHD problems the internal reconstruction vectors mentioned in the second step are always needed. For Euler or Navier-Stokes problems without source terms these are elided.

The initial conditions are assumed to be provided by the user as point values of the conservative variables for any supplied coordinate location. The solver internally loops over all cells and applies a tensor product Gauss-Legendre rule on each cell to obtain the cell-average quantities to sufficient accuracy. As with this source terms this rule uses R points per dimension, thus querying the user-supplied initial conditions function R^d times per cell.

As mentioned in Section 3.5, this solver uses Runge-Kutta style method-of-lines time integrators. The majority of the time integration loop consists of evaluating the right hand side within each Runge-Kutta stage. One evaluation of the right side consists of the following steps:

1. Fill the ghost cells in accordance with the boundary conditions.
2. Calculate Riemann states (and internal states if needed) at each quadrature point (see below).

3. Apply the positivity preserving limiter on the Riemann states, following Sections 3.4 and 4.3.3.
4. Populate Riemann states outside of the physical domain in accordance with the boundary conditions.
5. Call an approximate Riemann solver at each flux quadrature point.
6. Evaluate physical viscosity terms if present.
7. Evaluate and average the source terms if present.
 - For MHD this is always done in accordance with Section 4.4.
 - User-supplied source terms are also incorporated at this stage.
8. Integrate fluxes and update cell averages in accordance with the chosen RK method.

Naturally, the second step of calculating the Riemann and internal states contains most of the contributions from this work. This proceeds as follows:

1. Reconstruct unlimited Riemann states on all cells using cell averages from the full stencils \mathcal{S}_0 .
2. Evaluate the KXRCF indicator from either Section 3.3 or 4.3.1, and flag cells needing WENO.

3. For all flagged cells do the following:
 - (a) Construct transformation matrix Φ from eq. (3.14) or eq. (4.15) as appropriate using the central cell average data as the reference state. Construct the corresponding inverse transformation matrix Φ^{-1} according to eq. (3.15) or eq. (4.17).
 - (b) Project the average state for each cell in \mathcal{S}_0 onto the linearized primitive variables via eq. (3.2).
 - (c) Reconstruct pointwise values of linearized primitive variables at all face quadrature points via the reconstruction detailed in Section 2.5.
 - (d) Project the pointwise linearized primitive values back to conservative variables via eq. (3.3), yielding the Riemann states.

Fitting all of these tasks and all supported governing equations into a single piece of software is no small task. The interested reader is encouraged to have a look at the available code [72].

4.6 Numerical results

The full method for numerically solving the ideal MHD-GLM equations with an adiabatic equation of state has now been described. What remains is to test the method on a variety of benchmark problems to assess its capabilities and merits. The suite of

test problems and the significance of each one is described below, but first there are a some generalities in the method that need to be fixed in place, most of which follow directly from those in Section 3.6.

The stencil radius, R , is the main parameter governing the achievable order of accuracy. Throughout this section, as in Chapter 2 and Chapter 3, radius $R = 2$ and $R = 3$ stencils are considered, which should give at least 4th and 6th order accuracy respectively. Similarly, the number of quadrature points per direction used for flux and source term integration is set equal to R to match in accuracy, and length scale ℓ in the kernel function eq. (2.3) is tied to the grid scale as $\ell = 5\Delta$ in all cases. Numerical fluxes will be formed from the reconstructed Riemann states using a local Lax-Friedrichs flux.

Initial conditions are integrated to find cell-averages using the same quadrature rule with R points per direction as would be used for treating any source terms. For simplicity initial conditions are listed below using primitive variables, though naturally these would need to be converted to conservative variables prior to averaging. Note that the ψ field is always initialized to zero, and will be omitted from the given initial conditions in all following sections.

Each problem solved in the subsequent sections is present to highlight different features of the proposed method. First, convergence studies using isodensity vortex problems [1, 74] are shown to validate that the method reaches the designed order of accuracy. The Brio-Wu shock tube problem [11] is a classic one dimensional test for

shock capturing, which is solved here in both grid aligned and tilted configurations. The Orszag-Tang problem [77], and its three dimensional generalization [53, 64], is a qualitative test that serves to demonstrate the correct representation of complicated shock structures arising from initially smooth data. The field loop advection test [40, 41, 64] is extremely sensitive to divergence errors in the magnetic field, and is an important test to demonstrate the efficacy of the GLM cleaning formulation. An astrophysical jet generation problem [68, 91] is solved to demonstrate the efficacy of the proposed method as a global solver. Finally, low- β magnetized versions of the astrophysical jets presented in Section 3.6.4 are solved to demonstrate the positivity preserving qualities of the method [113].

4.6.1 Isodensity vortex

Analogs of the isentropic vortex presented in Section 3.6.1 as applicable to the ideal MHD equations are available as isodensity vortices. These are again smooth, and fully nonlinear, exact solutions of the ideal MHD equations that describe the advection of an exponentially localized magnetic vortex on a background flow. The two dimensional case was presented by Balsara in [1], and the three dimensional case was presented by Mignone et al. in [74]. These problems are solved as in [74], and experimental orders of convergence determined. This is a crucial test to perform as the accuracy of the discretization of the GLM source terms presented in Section 4.4 must be validated.

In two dimensions the domain is set as $\Omega = [-5, 5]^2$ with periodic boundary conditions in both directions. The initial conditions are set as

$$\rho = 1, \quad (4.37)$$

$$u_1 = 1 - y\kappa e^{(1-r^2)}, \quad (4.38)$$

$$u_2 = 1 + x\kappa e^{(1-r^2)}, \quad (4.39)$$

$$u_3 = 0, \quad (4.40)$$

$$p_g = 1 + \frac{1}{4} (\mu^2 (1 - 2r^2) - \kappa^2) e^{2(1-r^2)}, \quad (4.41)$$

$$B_1 = -y\mu e^{(1-r^2)}, \quad (4.42)$$

$$B_2 = x\mu e^{(1-r^2)}, \quad (4.43)$$

$$B_3 = 0, \quad (4.44)$$

where $r^2 = (x^2 + y^2)$, the parameters are set to $\kappa = \mu = 1/(2\pi)$, and the ratio of specific heats is $\gamma = 5/3$. At the final time of $t = 10$ the magnetic vortex should return to its initial position, and a measure of the error is available by comparing the initial and final states. The magnetic field is the most interesting quantity present, and the aggregate error (commensurate with that in [74])

$$Err_B = \sqrt{\|B_1 - B_1^{exact}\|_{L_1}^2 + \|B_2 - B_2^{exact}\|_{L_1}^2 + \|B_3 - B_3^{exact}\|_{L_1}^2}, \quad (4.45)$$

is considered throughout this section.

In three dimensions the domain is set as $\Omega = [-5, 5]^3$ with periodic boundary con-

Table 4.1: Shown are the experimental orders of convergence for the described method with radius $R = 2$ and $R = 3$ stencils as tested on the two dimensional isodensity magnetic vortex problem.

Radius $R = 2$			Radius $R = 3$		
N_x	Err_B	EOC	N_x	Err_B	EOC
32	2.02×10^{-3}	–	32	6.69×10^{-4}	–
64	1.23×10^{-4}	4.03	64	1.00×10^{-5}	6.06
128	4.28×10^{-6}	4.85	128	8.80×10^{-8}	6.83
256	1.37×10^{-7}	4.97	256	7.15×10^{-10}	6.94

ditions in all directions. The initial conditions are set as

$$\rho = 1, \quad (4.46)$$

$$u_1 = 1 - y\kappa e^{(1-r^2)}, \quad (4.47)$$

$$u_2 = 1 + x\kappa e^{(1-r^2)}, \quad (4.48)$$

$$u_3 = 2, \quad (4.49)$$

$$p_g = 1 + \frac{1}{4} (\mu^2 (1 - 2(r^2 - z^2)) - \kappa^2) e^{2(1-r^2)}, \quad (4.50)$$

$$B_1 = -y\mu e^{(1-r^2)}, \quad (4.51)$$

$$B_2 = x\mu e^{(1-r^2)}, \quad (4.52)$$

$$B_3 = 0, \quad (4.53)$$

where $r^2 = (x^2 + y^2 + z^2)$, the parameters are set to $\kappa = \mu = 1$, and the ratio of specific heats is $\gamma = 5/3$.

Both the two and three dimensional cases are solved on a sequence of finer grids using the RK4(3)9_F[3S₊^{*}] time integrator with tolerances of $\delta_a = \delta_r = 10^{-6}$ and a max-

Table 4.2: Shown are the experimental orders of convergence for the described method with radius $R = 2$ and $R = 3$ stencils as tested on the three dimensional isodensity magnetic vortex problem.

Radius $R = 2$			Radius $R = 3$		
N_x	Err_B	EOC	N_x	Err_B	EOC
32	4.59×10^{-4}	–	32	8.08×10^{-5}	–
64	2.47×10^{-5}	4.21	64	1.49×10^{-6}	5.76
128	8.48×10^{-7}	4.87	128	–	–

imal CFL of 1.25. The experimental orders of convergence can be seen in table 4.1 and table 4.2 for the two dimensional case and three dimensional cases respectively, each having results for the radius $R = 2$ and $R = 3$ schemes. In most cases the observed order of convergence matches or exceeds the expected convergence rates, much like the isentropic vortex in Section 3.6.1. This is an exceptional result, and shows that the splitting made in Section 4.4.1 for the sake of upwinding retains high-order accuracy. The only exception is the radius $R = 3$ case in three dimensions. For reasons that are not totally clear this case must be run with the KXRCF indicator disabled. It is likely that the observed order would rise in passing to the next grid resolution.

4.6.2 Brio-Wu shock tube

The Brio-Wu shock tube problem [11] is a critical benchmark problem for any compressible ideal MHD solver. As in Section 3.6.2, this problem is solved in two configurations. First, in the standard grid-aligned formulation we take the domain to be $\Omega = [0, 1] \times [0, 0.015625]$ with outflow conditions in the x -direction and periodic

conditions in the y -direction. The initial conditions are piecewise constant with $\rho_L = 1$, $B_{2,L} = 1$ and $p_L = 1$ set for $x < 0.5$ and $\rho_R = 0.125$, $B_{2,R} = -1$ and $p_R = 0.1$ otherwise. The velocity is zero throughout, and the x -direction magnetic field is set to $B_1 = 0.75$. The ratio of specific heats is set to $\gamma = 2$.

Second, we consider a tilted configuration where the same solution evolves oblique to the grid by an angle of $\theta \approx 26.5651^\circ$ (or $\tan \theta = 1/2$). This tests both for any preference for grid-aligned phenomena and for the divergence cleaning capabilities of the solver near shocks. Following the ideas in [60, 66] and in Section 3.6.2, the domain is set as $\Omega = [0, \sqrt{5}] \times [0, 2\sqrt{5}]$ with periodic conditions in both directions. The tilted coordinate is set as $x_{\parallel} = \frac{1}{\sqrt{5}}(2x + y)$ with $0 \leq x_{\parallel} \leq 4$, and the *left* state is initialized in the regions $0 \leq x_{\parallel} \leq 0.5$, $1.5 < x_{\parallel} \leq 2.5$, and $3.5 < x_{\parallel} \leq 4$, while the *right* state is imposed elsewhere. The initial magnetic field must also be rotated with the grid, and takes the initial values

$$B_{1,L} = \frac{1}{\sqrt{5}}(2B_{\parallel} - B_{\perp}), \quad B_{1,R} = \frac{1}{\sqrt{5}}(2B_{\parallel} + B_{\perp}), \quad (4.54)$$

$$B_{2,L} = \frac{1}{\sqrt{5}}(B_{\parallel} + 2B_{\perp}), \quad B_{2,R} = \frac{1}{\sqrt{5}}(B_{\parallel} - 2B_{\perp}), \quad (4.55)$$

where $B_{\parallel} = 0.75$ and $B_{\perp} = 1$. In this configuration the final density along the line $0 \leq x_{\parallel} \leq 1$ will match that of the grid aligned configuration.

Both the aligned and tilted configurations are solved using the SSP(4,3,2) time integrator with tolerances of $\delta_a = \delta_r = 10^{-4}$. The aligned cases use a grid with spacing $\Delta = 1/512$, and the tilted cases use a grid with spacing $\sqrt{5}/1280$ which sets a matching

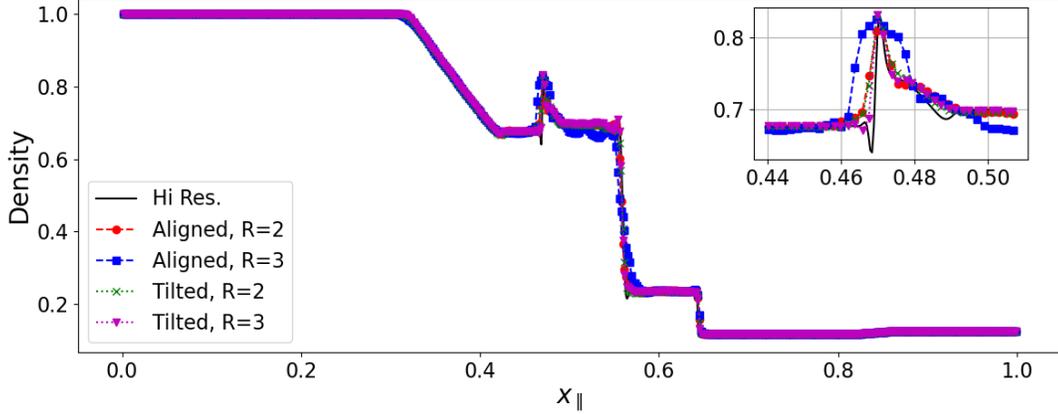


Figure 4.2: One dimensional traces of the density in the Brio-Wu shock tube problem are shown for four different configurations, and compared to a high resolution solution. Red circles and blue squares correspond to the grid-aligned cases with the radius $R = 2$ and $R = 3$ schemes respectively. Green crosses and purple triangles correspond to the grid-tilted cases with the radius $R = 2$ and $R = 3$ schemes respectively. The inset zooms in on the density spike to the right of the rarefaction.

number of cells along the line $0 \leq x_{\parallel} \leq 1, x_{\perp} = 0$. Traces of the density extracted from these results are visible in Figure 4.2. Apart from the grid-aligned radius $R = 3$ case, all results agree closely with each other, and the solver overall exhibits minimal preference between aligned and tilted phenomena. The excessive oscillations and broadened density peak in the aligned $R = 3$ results are troubling. Reverting this test to use the eight wave formulation suppresses this poor behavior, as does the application of more elaborate Riemann solvers such as an approximate Osher-Solomon scheme [12]. The tilted $R = 3$ case does not have these shortcomings, nor do any of the remaining $R = 3$ results.

4.6.3 Orszag-Tang vortex

The Orszag-Tang vortex, initially posed by Orszag and Tang in [77] and subsequently solved in many papers (see e.g. [40, 65, 101, 23]), evolves a set of staggered rotational velocity and magnetic fields. This initially smooth configuration rapidly generates complicated shock structures and makes an excellent test problem regarding the ability of a scheme to resolve both types of features. Additionally, failure to control divergence errors can result in spurious structures appearing as well as a total crash of the code [23]. The classic version of this problem is only two dimensional, but a three dimensional version was introduced in [53] which adds a small velocity perturbation that can be seen below.

This problem is solved on the domain $\Omega = [0, 1]^d$ with periodic boundary conditions

in all directions. The three dimensional initial conditions are

$$\rho = \gamma^2, \quad (4.56)$$

$$u_1 = -(1 + \varepsilon \sin(2\pi z)) \sin(2\pi y), \quad (4.57)$$

$$u_2 = (1 + \varepsilon \sin(2\pi z)) \sin(2\pi x), \quad (4.58)$$

$$u_3 = \varepsilon \sin(2\pi z), \quad (4.59)$$

$$p_g = \gamma, \quad (4.60)$$

$$B_1 = -\sin(2\pi y), \quad (4.61)$$

$$B_2 = \sin(4\pi x) \quad (4.62)$$

$$B_3 = 0, \quad (4.63)$$

where $\varepsilon = 0.2$ controls the strength of the perturbation, and the ratio of specific heats is $\gamma = 5/3$. Setting $z = 0$ reduces these to the traditional two dimensional initial conditions.

This problem is solved in two dimensions on a grid with spacing $\Delta = 1/1024$ by both the radius $R = 2$ and $R = 3$ schemes. The three dimensional version is solved on a grid with spacing $\Delta = 1/256$ by the radius $R = 2$ scheme. All results were obtained with the SSP(4,3,2) time integrator with tolerances of $\delta_a = \delta_r = 10^{-4}$ and a maximal CFL number of 1.25.

The density and magnetic pressure are visualized at times of $t = 0.5$ and $t = 1.0$, and can be seen for the two dimensional radius $R = 2$ scheme in Figure 4.3, the two

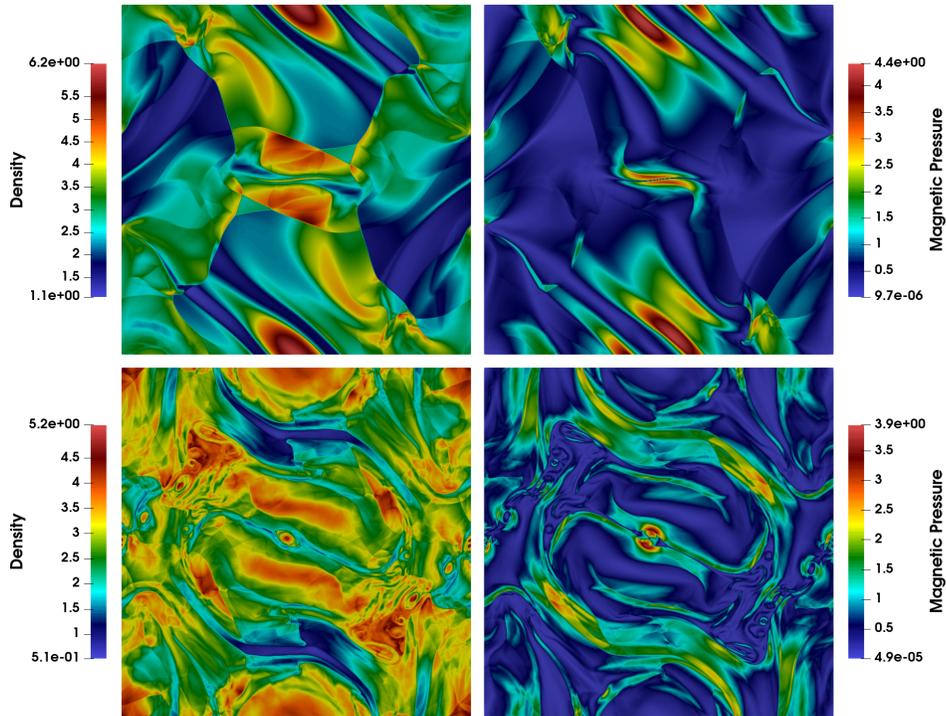


Figure 4.3: The density (left column) and magnetic pressure (right column) are shown for the two dimensional Orszag-Tang vortex at times of $t = 0.5$ (top row) and $t = 1$ (bottom row). The radius $R = 2$ scheme was used with a grid spacing of $\Delta = 1/1024$.

dimensional radius $R = 3$ scheme in Figure 4.4, and the three dimensional radius $R = 2$ scheme in Figure 4.5. Up to time $t = 0.5$ all results match very well with the literature (e.g. [40, 65, 101, 23] in two dimensions, and [53, 64] in three dimensions).

Running this problem to the longer time of $t = 1.0$ is more challenging and less reported on. Notably, the radius $R = 3$ results in Figure 4.4 exhibit broken symmetry in the center of the domain, which has been reported before [31]. This likely due to the non-associativity of floating point addition as observed in solvers for the compressible Euler equations [34]. In the three dimensional case at this longer time some fingering

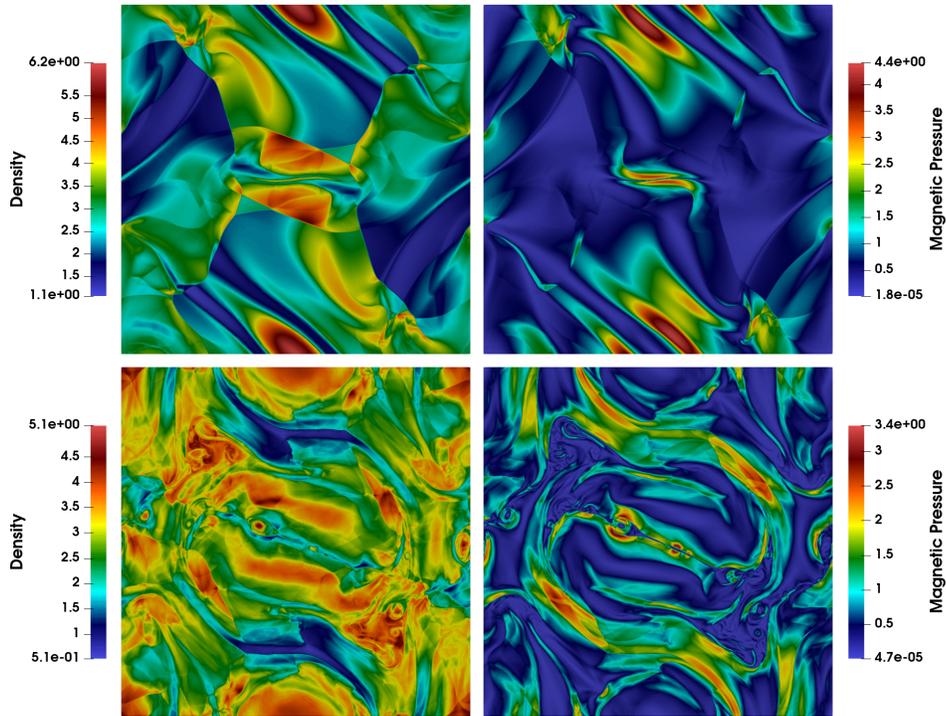


Figure 4.4: The density (left column) and magnetic pressure (right column) are shown for the two dimensional Orszag-Tang vortex at times of $t = 0.5$ (top row) and $t = 1$ (bottom row). The radius $R = 3$ scheme was used with a grid spacing of $\Delta = 1/1024$.

in the density and magnetic pressure are visible on the lower left face of the cube as viewed in Figure 4.5, which is similar to the result in [65], but more pronounced.

4.6.4 Field loop advection

The field loop advection problem provides a very sensitive test of a schemes ability to control divergence errors. A very weak circular magnetic field is placed into a small region within a uniformly advecting flow. The weak nature of the field means that this loop should interact minimally with the background flow, and maintaining the structure

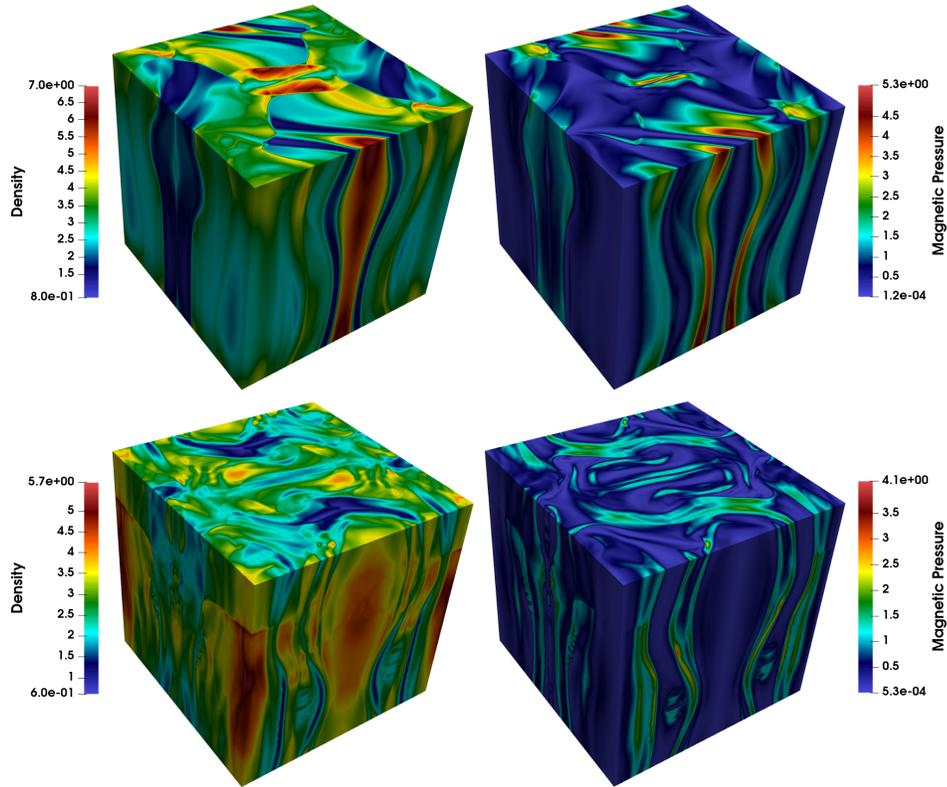


Figure 4.5: The density (left column) and magnetic pressure (right column) are shown for the three dimensional Orszag-Tang vortex at times of $t = 0.5$ (top row) and $t = 1$ (bottom row). The radius $R = 2$ scheme was used with a grid spacing of $\Delta = 1/256$.

of this loop over time is challenging. It is worth noting however that purely passive advection of this loop is not an exact solution of the ideal MHD equations, and some amount of interaction with the background flow is inevitable.

In two dimensions this problem is solved as in [40, 101]. The domain is set as $\Omega = [-1, 1] \times [-0.5, 0.5]$ with periodic boundary conditions in both directions. The uniform background flow is initialized with $\rho = 1$, $u_1 = 2$, $u_2 = 1$, $u_3 = 0$, and $p_g = 1$, which returns the field loop to its initial position at time $t = 1$. The magnetic field is

initialized as

$$B_1 = -A_0 \frac{y}{r}, \quad r < R, \quad (4.64)$$

$$B_2 = A_0 \frac{x}{r}, \quad r < R, \quad (4.65)$$

where $r = \sqrt{x^2 + y^2}$, $R = 0.3$ is the radius of the loop, and $A_0 = 10^{-3}$ controls the strength of the loop. For $r > R$ the magnetic field vanishes. This configuration results in an initial magnetic pressure of $p_b = A_0^2/2 = 5 \times 10^{-7}$ within the loop. The ratio of specific heats is $\gamma = 1.4$.

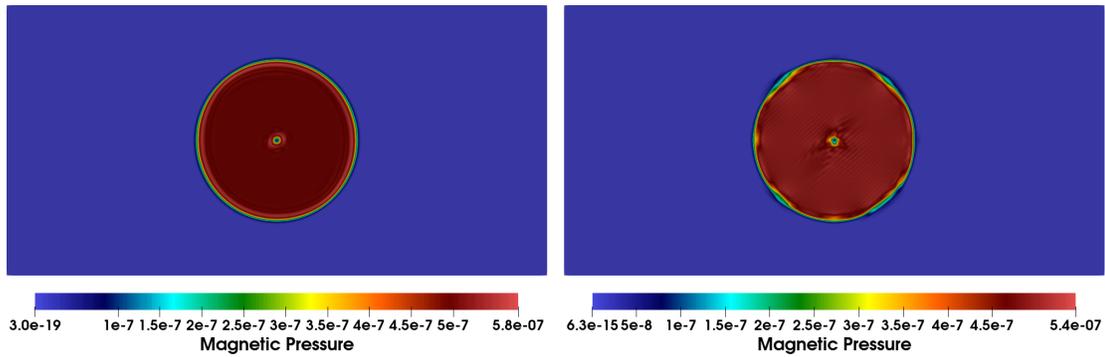


Figure 4.6: Shown is the magnetic pressure in the two dimensional field loop problem as obtained by the radius $R = 2$ scheme (left) and radius $R = 3$ scheme (right) at the final time of $t = 1$. Both cases utilize a grid spacing of $\Delta = 1/256$.

Figure 4.6 shows the magnetic pressure at the final time of $t = 1$ after the loop has traversed the domain once as obtained from the radius $R = 2$ and $R = 3$ schemes in two dimensions. In each case the grid spacing is set to $\Delta = 1/256$, and the $\text{RK3}(2)5_F[3S_+^*]$ time integrator was used with tolerances of $\delta_a = \delta_r = 10^{-4}$ and a maximal CFL number of 1. The radius $R = 3$ scheme requires disabling the KXRCF indicator, hence applying

WENO throughout the domain. In spite of this, this scheme exhibits a smaller deviation from the initial magnetic pressure.

An interesting additional challenge was posed in [101] where $u_3 = 1$ is set within the background flow. In this case $B_3 = 0$ can only be maintained if the remaining two components have exactly zero divergence. Though not reported here, this configuration was tested and B_3 only exhibited negligible fluctuations on the order of 10^{-18} .

In three dimensions this problem is solved as in [41, 64]. The domain is set as $\Omega = [-0.5, 0.5] \times [-0.5, 0.5] \times [-1, 1]$ and periodic boundary conditions are set in both directions. The uniform background flow is initialized with $\rho = 1$, $u_1 = 1$, $u_2 = 1$, $u_3 = 2$, and $p_g = 1$. As before, this background flow returns the field loop to its initial position at time $t = 1$. This version of the test initializes the field loop as a tube tilted about the y -axis. After performing the necessary change of variables, the initial field is

$$B_1 = -\frac{2A_0 y}{\sqrt{5} \tilde{r}}, \quad \tilde{r} < R, \quad (4.66)$$

$$B_2 = A_0 \frac{\tilde{x}}{\tilde{r}}, \quad \tilde{r} < R, \quad (4.67)$$

$$B_3 = -\frac{A_0 y}{\sqrt{5} \tilde{r}}, \quad \tilde{r} < R, \quad (4.68)$$

where now r requires more care to set. This is most easily written by considering three transformations of the x -coordinate, given by $x' = (2(x-1) + z)\sqrt{5}$, $x'' = (2x+z)\sqrt{5}$,

and $x''' = (2(x+1)+z)\sqrt{5}$. The radius is then set to

$$\tilde{r} = \min \left\{ \sqrt{x'^2 + y^2}, \sqrt{x''^2 + y^2}, \sqrt{x'''^2 + y^2} \right\}, \quad (4.69)$$

and \tilde{x} is set to the corresponding transformation of the x -coordinate. The radius and strength of the loop are maintained at $R = 0.3$ and $A_0 = 10^{-3}$. Once again, for $\tilde{r} > R$ the magnetic field vanishes, and the initial magnetic pressure is $p_b = A_0^2/2 = 5 \times 10^{-7}$ within the loop.

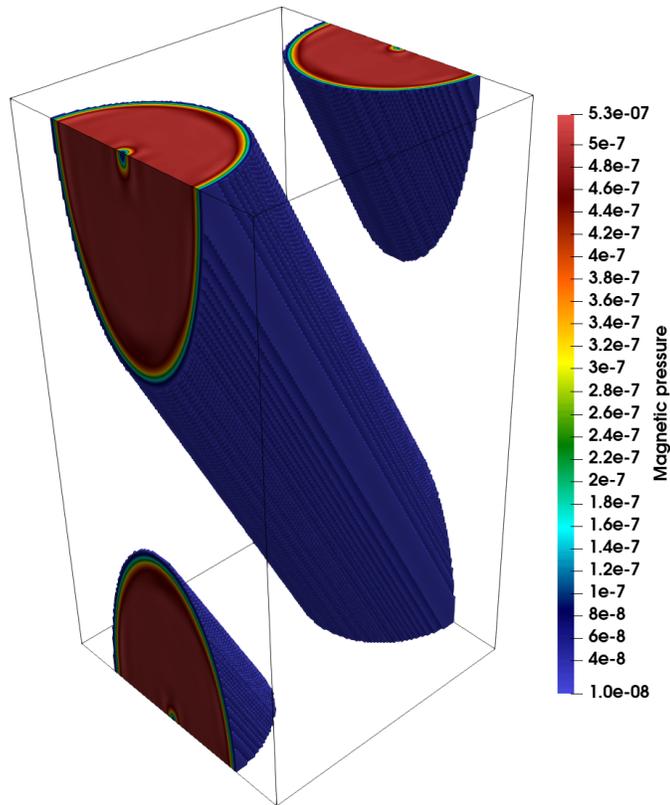


Figure 4.7: Shown is the magnetic pressure in the three dimensional field loop problem as obtained by the radius $R = 2$ scheme at the final time of $t = 1$ on a grid with spacing $\Delta = 1/256$. To more easily visualize the solution all cells with $p_b < 10^{-8}$ have been omitted.

Figure 4.7 shows the magnetic pressure at the final time of $t = 1$ after the loop has traversed the domain once as obtained from the radius $R = 2$ scheme in three dimensions. In each case the grid spacing is set to $\Delta = 1/128$, and as before the RK3(2)5_F[3S₊*] time integrator was used with tolerances of $\delta_a = \delta_r = 10^{-4}$ and a maximal CFL number of 1. Compared with the two dimensional case there is slightly greater dissipation of the magnetic field in the core of the loop (where it fails to be differentiable), and somewhat surprisingly there is an even smaller deviation in the magnetic pressure from its initial value.

4.6.5 Jet generation problem

The jet generation problem was posed by Li et al. [68] and more recently solved by Seo and Ryu [91]. The fluid is initially quiescent but has an overlaid magnetic field that is far from equilibrium. This induces strong outflowing jets from the origin and generates a complicated flow field.

The domain is set as $\Omega = [-12, 12]^3$ with outflow conditions on all faces. The initial conditions have constant density and gas pressure given by $\rho = 1$ and $p_g = 1$, with zero velocity. The initial magnetic field is

$$B_1 = (2xz - A_0y) e^{-r^2}, \quad (4.70)$$

$$B_2 = (2yz + A_0x) e^{-r^2}, \quad (4.71)$$

$$B_3 = 2(1 - x^2 - y^2) e^{-r^2}, \quad (4.72)$$

with $r^2 = (x^2 + y^2 + z^2)$, and $\gamma = 5/3$. This problem was solved with the radius $R = 2$ scheme on a grid with spacing $\Delta = 24/320$, using the SSP(4,3,2) time integrator with tolerances $\delta_a = \delta_r = 10^{-3}$ and a maximal CFL number of 1.25. Curiously, the use of the KXRCF indicator on this problem gives oscillations in the post-shock flow behind the roughly spherical outgoing shock. To suppress these oscillations the indicator is disabled, and WENO applied throughout the domain.

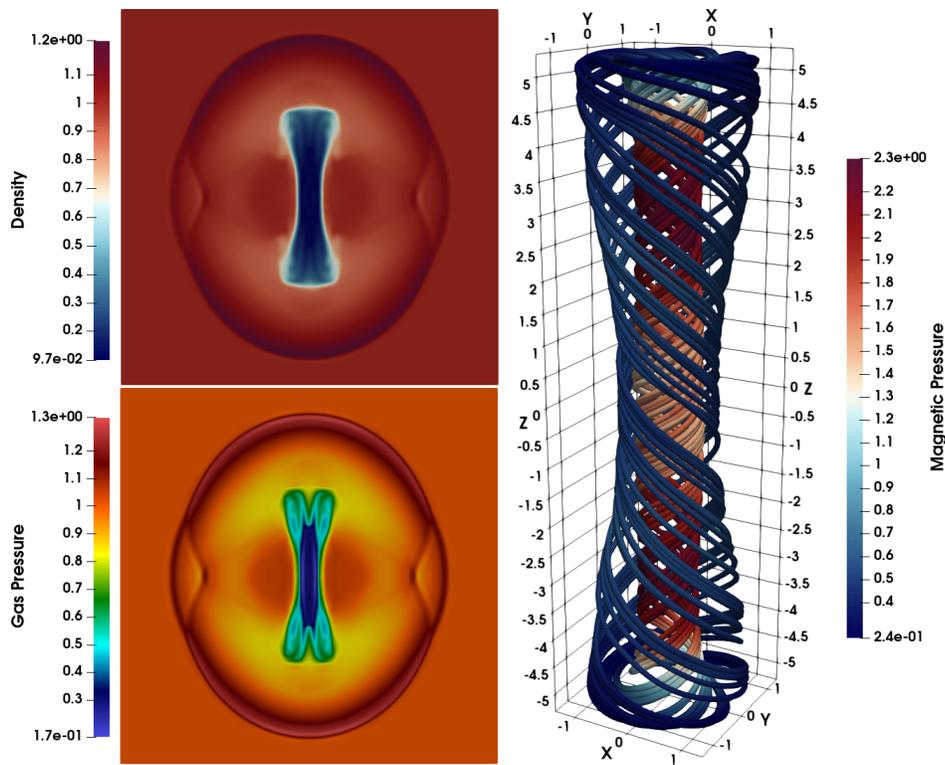


Figure 4.8: The density (top left) and gas pressure (bottom left) are shown along the $y = 0$ plane of the jet generation problem as solved by the $R = 2$ scheme to a final time of $t = 5$ on a grid with spacing $\Delta = 24/320$. The right panel shows one magnetic field line colored by magnetic pressure.

Slices of the density and gas pressure can be seen in Figure 4.8 at the final time

$t = 5$, and are directly comparable to the results in [91] despite using a lower resolution here. A single magnetic field line is plotted and is very similar to that shown in [68], exhibiting the same tight inner helix and loose counter-rotating outer helix. Being a global simulation, this problem exhibits a wide range of scales. The roughly spherical outgoing shock occupies much of the domain at the final time, but has a relatively simple structure with minimal second flows. The low density jets, particularly at their heads, exhibit more complicated structure that is certainly under-resolved here. The fact that useful results are generated is a testament to high order solvers. Nevertheless, this problem is an ideal candidate for future studies with adaptive mesh refinement.

4.6.6 Astrophysical jet

The final test problem consists of a magnetized version of the high density jet presented in Section 3.6.4. The two dimensional jet presented here follows the configuration from [113], and the three dimensional version is a natural extension thereof. In two dimensions the domain is set as $\Omega = [0, 0.5] \times [0, 1.5]$ with symmetry conditions on the lower x -face, outflow conditions on the upper x - and y -faces, and mixed inflow/outflow conditions on the lower y -face. The domain initially contains quiescent gas with an initial density of $\rho = \gamma$, and a gas pressure of $p_g = 1$. The ratio of specific heats is $\gamma = 1.4$. A background vertical magnetic field is set as $B_2 = \sqrt{2/\beta}$ with $\beta = 10^{-4}$. Inflowing gas is prescribed on the lower y -face for $x < 0.05$ with a density

of $\rho_{in} = 10\gamma$, gas pressure of $p_{g,in} = 1$, and velocity of $u_2 = 800$. The magnetic field is set to extrapolation over the entire face.

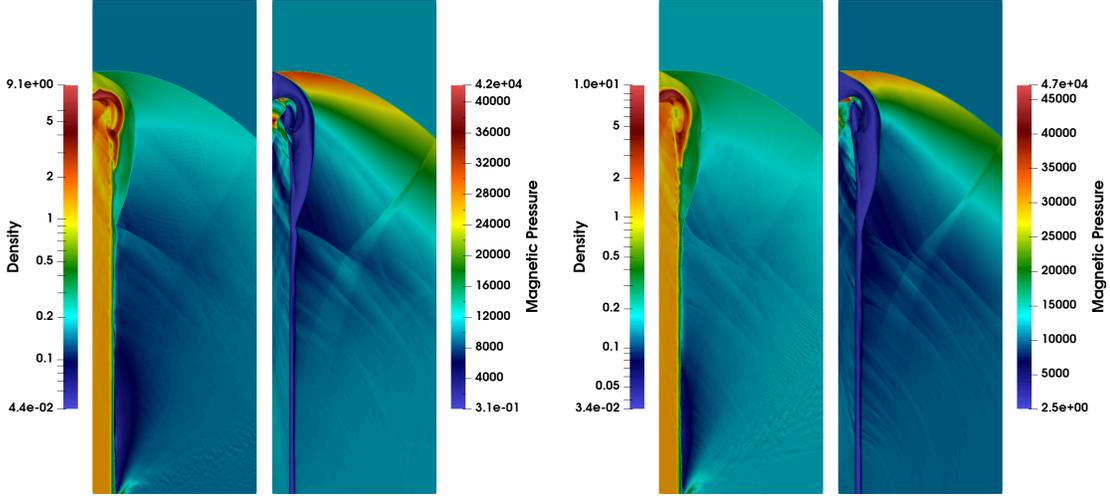


Figure 4.9: The left two panels show the logarithmic density and magnetic pressure fields in the magnetized astrophysical jet problem at $t = 0.002$ as obtained from the radius $R = 2$ scheme. The right two panels show the same results as obtained from the radius $R = 3$ scheme. Both schemes utilize a grid spacing of $\Delta = 1/1024$.

The three dimensional set up follows naturally from the two dimensional problem. The domain is now $\Omega = [0, 0.5] \times [0, 1.5] \times [0, 0.5]$, where the lower x - and z - faces both have symmetry conditions, the lower y -face has mixed inflow/outflow conditions, and all of the upper faces have outflow conditions. The initially quiescent gas is set exactly as before, as is the inflowing gas on the lower y -face which is now imposed for $\sqrt{x^2 + z^2} < 0.05$.

The two dimensional problem is solved by the radius $R = 2$ and $R = 3$ schemes on a grid with spacing $\Delta = 1/1024$, and the three dimensional version is solved by the radius $R = 2$ scheme on a grid with spacing $\Delta = 1/320$. All cases utilize the SSP(4,3,2) time

integrator with tolerances of $\delta_a = \delta_r = 10^{-2}$ and a maximal CFL of 1. These problems ultimately flag the majority of the domain as needing WENO, so the KXRCF indicator is disabled to apply WENO throughout.

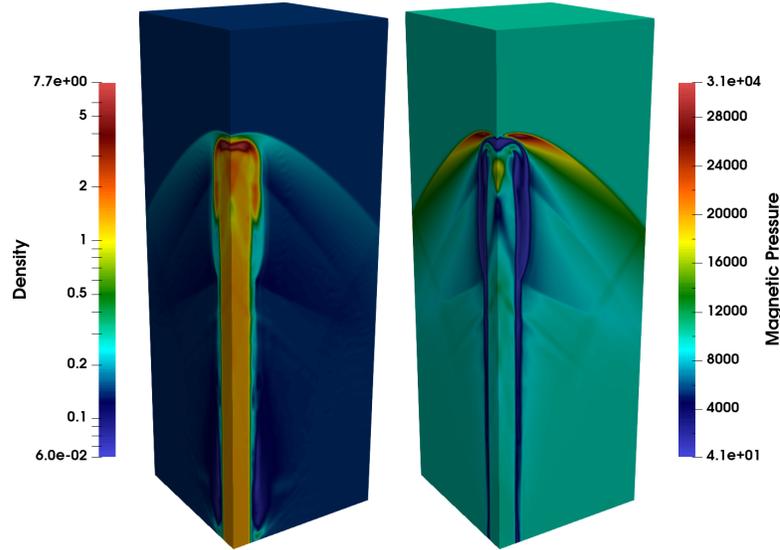


Figure 4.10: The logarithmic density (left) and magnetic pressure (right) are shown for the three dimensional magnetized astrophysical jet as obtained with the radius $R = 2$ scheme at the final time $t = 0.002$ on a grid with spacing $\Delta = 1/320$.

Plots of the logarithmic density and the magnetic pressure for the two dimensional problem are visible in Figure 4.9. These plots as shown are directly comparable to those presented in [113]. Corresponding results for the three dimensional case are shown in Figure 4.10. These magnetized jets have a richer morphology than their unmagnetized counterparts, shown in Section 3.6.4. There are now two bow shocks launched from the head of the jet as it pushes into the ambient fluid. The inner shock is strongly confined by the magnetic field, and rejoins the sheath of the jet part way down. The outer bow shock is of a magnetosonic nature and is unconfined by the strong vertical mag-

netic field. Just as in [113], a spurious reflection of this shock can be seen originating from the upper x -face of the domain in two dimensions. The three dimensional jet demonstrates nearly identical morphology.

Chapter 5

Conclusion

This dissertation presented kernel-based weighted essentially non-oscillatory schemes for the compressible Euler equations, the compressible Navier-Stokes equations, and the ideal magnetohydrodynamics equations. These schemes are applicable in two and three space dimensions, and their description is supported by a highly parallel multi-GPU code written by the author. The ultimate result is an efficient and general purpose solver applicable to a wide variety of flow phenomena, from nearly incompressible turbulence to flows dominated by extreme shocks.

Chapter 2 began with a discussion of reproducing kernel Hilbert spaces and linear recovery processes therein. These abstract notions were applied to the reconstruction problem. This allowed reconstruction to be posed in a fashion that is largely independent of the number of space dimensions or the specific layout of the given data. This

flexibility greatly simplified the development and presentation of a fully multidimensional reconstruction scheme. More importantly, this flexibility allowed for the application of this multidimensional reconstruction scheme to weighted essentially non-oscillatory (WENO) reconstruction. The latter half of this chapter deliberates on this adaptation, and forms the foundation of the following chapters.

Chapter 3 incorporates this reconstruction scheme into finite volume method for the compressible Euler equations closed by an adiabatic equation of state. Chapter 2 considers only the reconstruction of scalar valued functions. Systems of reconstruction variables are discussed to extend this reconstruction method to the treatment of multicomponent data. The standard choice of characteristic variables is recalled and used to motivate the use of alternate sets of variables for reconstruction. A few key drawbacks to these variables are highlighted, and a novel set of so-called linearized primitive variables are introduced to ameliorate these issues while still providing high quality reconstructions. On the other hand, the elaborate WENO reconstruction is observed to be unnecessary for many cells in a typical run of a finite volume solver. Troubled cell indicators were adapted from the discontinuous Galerkin community which allow for WENO to be avoided in most of the domain. This makes the reconstruction process cheaper and provides valuable memory savings. A self-adjusting positivity preserving limiter, adapted from existing finite volume methods, was described and allows for the solution of problems with extremely strong shocks. Choices of time integration

methods were discussed, particularly as relevant for use on GPUs. The chapter closed with a variety of challenging benchmark problems, including extensions to handle the compressible Navier-Stokes equations.

Chapter 4 extends these ideas further to handle the equations of ideal magnetohydrodynamics. These equations bring a dramatically new requirement in the form of the divergence-free constraint on the magnetic field. A few standard methods for satisfying this constraint were reviewed, and the method of generalized Lagrange multipliers (GLM) ultimately selected for the present solver. Reconstruction over linearized primitive variables, flagging cells for WENO with troubled cell indicators, and limitation of the reconstructed states for positivity preservation, were all adapted from the previous chapter. Of particular interest is the introduction of source terms necessary for the GLM method which depend on derivatives of the state variables. The accurate discretization of these source terms was deliberated on, and a few useful approximations made. With all of the supported physics and numerical schemes fully described, an overview of the developed software to support this work could be given. As before, this chapter closed with the application of the proposed scheme to a variety of stringent benchmark problems.

5.0.1 Directions for future study

As complete as the work presented here is, there are always possibilities for future study. A few such possibilities are discussed here. These fall roughly in the categories of extensions to the implementation, physics additions, and improvements of the scheme itself.

Two directions that have already been mentioned throughout the dissertation would be to embed these methods into an adaptive mesh refinement framework, or to extend the reconstruction scheme to unstructured meshes. The former is likely a more productive avenue for the investigation of astrophysical phenomena, while the latter would be more suited to handling problems with manifestly complicated geometries like turbo-machinery. Other algorithmic improvements could come from a more thorough investigation of time integrators, and particularly the application of error-adaptive integrators to problems with very strong shocks.

Regarding additional physics, one could consider incorporating other equations of state, supporting multi-fluid evolution and reaction terms, or adding non-ideal MHD terms. Whole new sets of governing equations could be considered, and in fact the available code does have a very preliminary implementation for special relativistic hydrodynamics.

Finally, the scheme itself leaves some room for improvement. More optimal tagging criteria for WENO reconstruction could give improved memory savings. For MHD in

particular, it would be interesting to see an investigation of different indicator variables. Given an improved tagging system, the WENO reconstruction could be simplified and made cheaper by not trying to match the accuracy of the full stencil reconstruction at all. On a similar note, there is plenty of room to investigate other smoothness indicators and formulations of the nonlinear weights.

It is the author's hope that by providing a full implementation these avenues for future work can be more easily explored.

Bibliography

- [1] Dinshaw S. Balsara. Second-order-accurate schemes for magnetohydrodynamics with divergence-free reconstruction. *The Astrophysical Journal Supplement Series*, 151(1):149, 2004.
- [2] Dinshaw S. Balsara. Self-adjusting, positivity preserving high order schemes for hydrodynamics and magnetohydrodynamics. *Journal of Computational Physics*, 231(22):7504–7517, 2012.
- [3] Dinshaw S Balsara, Christoph Altmann, Claus-Dieter Munz, and Michael Dumbser. A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG+HWENO schemes. *Journal of Computational Physics*, 226(1):586–620, 2007.
- [4] Dinshaw S Balsara, Sudip Garain, and Chi-Wang Shu. An efficient class of weno schemes with adaptive order. *Journal of Computational Physics*, 326:780–804, 2016.
- [5] Dinshaw S Balsara, Tobias Rumpf, Michael Dumbser, and Claus-Dieter Munz. Efficient, high accuracy ader-weno schemes for hydrodynamics and divergence-free magnetohydrodynamics. *Journal of Computational Physics*, 228(7):2480–2516, 2009.
- [6] P. Batten, N. Clarke, C. Lambert, and D. M. Causon. On the choice of wavespeeds for the hllc riemann solver. *SIAM Journal on Scientific Computing*, 18(6):1553–1570, 1997.
- [7] 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.
- [8] 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.

- [9] Rémi Bourgeois and Dongwook Lee. GP-MOOD: A positivity-preserving high-order finite volume method for hyperbolic conservation laws. *Journal of Computational Physics*, 471:111603, 2022.
- [10] J.U. Brackbill and DC Barnes. The effect of nonzero product of magnetic gradient and B on the numerical solution of the magnetohydrodynamic equations. *Journal of Computational Physics*, 35:426–430, 1980.
- [11] Moysey Brio and Cheng Chin Wu. An upwind differencing scheme for the equations of ideal magnetohydrodynamics. *Journal of computational physics*, 75(2):400–422, 1988.
- [12] Manuel J. Castro, José M. Gallardo, and Antonio Marquina. Approximate osher–solomon schemes for hyperbolic systems. *Applied Mathematics and Computation*, 272:347–368, 2016. Recent Advances in Numerical Methods for Hyperbolic Partial Differential Equations.
- [13] Marcos Castro, Bruno Costa, and Wai Sun Don. High order weighted essentially non-oscillatory WENO-Z schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 230(5):1766–1792, 2011.
- [14] Marcos Castro, Bruno Costa, and Wai Sun Don. High order weighted essentially non-oscillatory weno-z schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 230(5):1766–1792, 2011.
- [15] Li Li Chen and Cong Huang. One dimensional hybrid weno-ao method using improved troubled cell indicator based on extreme point. *Computers & Fluids*, 225:104976, 2021.
- [16] Shusheng Chen, Boxi Lin, Yansu Li, and Chao Yan. Hllc+: Low-mach shock-stable hllc-type riemann solver for all-speed flows. *SIAM Journal on Scientific Computing*, 42(4):B921–B950, 2020.
- [17] Alina Chertock, Shaoshuai Chu, and Alexander Kurganov. Adaptive high-order A-WENO schemes based on a new local smoothness indicator. *East Asian Journal on Applied Mathematics*, 13(3):576–609, 2023.
- [18] Stéphane Clain, Steven Diot, and Raphaël Loubère. A high-order finite volume method for systems of conservation laws—multi-dimensional optimal order detection (MOOD). *Journal of computational Physics*, 230(10):4028–4050, 2011.
- [19] I. Cravero, G. Puppo, M. Semplice, , and G. Visconti. CWENO: Uniformly accurate reconstructions for balance laws. *Mathematics of Computation*, 87:1689–1719, 2018.

- [20] Isabella Cravero and Matteo Semplice. On the accuracy of WENO and CWENO reconstructions of third order on nonuniform meshes. *Journal of Scientific Computing*, 67(3):1219–1246, 2016.
- [21] G Dal Maso, P. G LeFloch, and F Murat. Definition and weak stability of nonconservative products. *Journal de mathématiques pures et appliquées*, 74(6):483–548, 1995.
- [22] A. Dedner, F. Kemm, D. Kröner, C.-D. Munz, T. Schnitzer, and M. Wesenberg. Hyperbolic divergence cleaning for the mhd equations. *Journal of Computational Physics*, 175(2):645–673, 2002.
- [23] Dominik Derigs, Andrew R. Winters, Gregor J. Gassner, Stefanie Walch, and Marvin Bohm. Ideal glm-mhd: About the entropy consistent nine-wave magnetic field divergence diminishing ideal magnetohydrodynamics equations. *Journal of Computational Physics*, 364:420–467, 2018.
- [24] Steven Diot, Stéphane Clain, and Raphaël Loubère. Improved detection criteria for the multi-dimensional optimal order detection (MOOD) on unstructured meshes with very high-order polynomials. *Computers & Fluids*, 64:43–63, 2012.
- [25] T.A. Driscoll and B. Fornberg. Interpolation in the limit of increasingly flat radial basis functions. *Computers & Mathematics with Applications*, 43(3):413–422, 2002.
- [26] H. Carter Edwards, Christian R. Trott, and Daniel Sunderland. Kokkos: Enabling manycore performance portability through polymorphic memory access patterns. *Journal of Parallel and Distributed Computing*, 74(12):3202 – 3216, 2014. Domain-Specific Languages and High-Level Frameworks for High-Performance Computing.
- [27] C.R. Evans and J.F. Hawley. Simulation of magnetohydrodynamic flows – a constrained transport method. *The Astrophysical Journal*, 332:659–677, 1988.
- [28] Gregory Fasshauer and Michael McCourt. *Kernel-based Approximation Methods using MATLAB*. WORLD SCIENTIFIC, 2015.
- [29] Gregory E Fasshauer and Michael J McCourt. *Kernel-based approximation methods using Matlab*, volume 19. World Scientific Publishing Company, 2015.
- [30] Imre Fekete, Sidafa Conde, and John N. Shadid. Embedded pairs for optimal explicit strong stability preserving runge–kutta methods. *Journal of Computational and Applied Mathematics*, 412:114325, 2022.

- [31] Kyle Gerard Felker and James M. Stone. A fourth-order accurate finite volume method for ideal mhd via upwind constrained transport. *Journal of Computational Physics*, 375:1365–1400, 2018.
- [32] Hui Feng, Fuxing Hu, and Rong Wang. A new mapped weighted essentially non-oscillatory scheme. *Journal of Scientific Computing*, 51(2):449–473, 2012.
- [33] Hui Feng, Cong Huang, and Rong Wang. An improved mapped weighted essentially non-oscillatory scheme. *Applied Mathematics and Computation*, 232:453–468, 2014.
- [34] Nico Fleischmann, Stefan Adami, and Nikolaus A Adams. Numerical symmetry-preserving techniques for low-dissipation shock-capturing schemes. *Computers & Fluids*, 189:94–107, 2019.
- [35] Natasha Flyer, Bengt Fornberg, Victor Bayona, and Gregory A. Barnett. On the role of polynomials in rbf-fd approximations: I. interpolation and accuracy. *Journal of Computational Physics*, 321:21–38, 2016.
- [36] Bengt Fornberg, Erik Lehto, and Collin Powell. Stable calculation of gaussian-based rbf-fd stencils. *Computers and Mathematics with Applications*, 65(4):627–637, 2013.
- [37] Richard Franke. Scattered data interpolation: Tests of some methods. *Mathematics of computation*, 38(157):181–200, 1982.
- [38] K. O. Friedrichs and P. D. Lax. Systems of conservation equations with a convex extension. *Proceedings of the National Academy of Sciences*, 68(8):1686–1688, 1971.
- [39] Guosheng Fu and Chi-Wang Shu. A new troubled-cell indicator for discontinuous galerkin methods for hyperbolic conservation laws. *Journal of Computational Physics*, 347:305–327, 2017.
- [40] Thomas A Gardiner and James M Stone. An unsplit Godunov method for ideal MHD via constrained transport. *Journal of Computational Physics*, 205(2):509–539, 2005.
- [41] Thomas A Gardiner and James M Stone. An unsplit Godunov method for ideal MHD via constrained transport in three dimensions. *Journal of Computational Physics*, 227(8):4123–4141, 2008.
- [42] Giorgio Giangaspero, Edwin van der Weide, Magnus Svärd, Mark H Carpenter, and Ken Mattsson. Case c3. 3: Taylor-green vortex. In *Case summary for 3rd International Workshop on Higher-Order CFD Methods*, 2015.

- [43] Francis X Giraldo. *An Introduction to Element-Based Galerkin Methods on Tensor-Product Bases*. Springer Cham, 2020.
- [44] Sergei Godunov and I Bohachevsky. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. *Matematičeskij sbornik*, 47(3):271–306, 1959.
- [45] Sergei K Godunov. Symmetric form of the equations of magnetohydrodynamics. *Numerical Methods for Mechanics of Continuum Medium*, 1:26–34, 1972.
- [46] Sergei Konstantinovich Godunov. A difference method for numerical calculation of discontinuous solutions of the equations of hydrodynamics. *Matematicheskii Sbornik*, 47(89)(3):271–306, 1959.
- [47] Youngsoo Ha and Carl L Gardner. Positive scheme numerical simulation of high mach number astrophysical jets. *Journal of Scientific Computing*, 34(3):247–259, 2008.
- [48] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II*. Springer Berlin, Heidelberg, 1996.
- [49] Ami Harten. Eno schemes with subcell resolution. *Journal of Computational Physics*, 83(1):148–184, 1989.
- [50] Ami Harten. Adaptive multiresolution schemes for shock computations. *Journal of Computational Physics*, 115(2):319–338, 1994.
- [51] Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, iii. In *Upwind and high-resolution schemes*, pages 218–290. Springer, 1987.
- [52] Amiram Harten, Peter D. Lax, and Bram van Leer. On upstream differencing and godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25(1):35–61, 1983.
- [53] Christiane Helzel, James A. Rossmanith, and Bertram Taetz. An unstaggered constrained transport method for the 3d ideal magnetohydrodynamic equations. *Journal of Computational Physics*, 230(10):3803–3829, 2011.
- [54] 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.

- [55] Jan S. Hesthaven, Fabian Mönkeberg, and Sara Zaninelli. RBF based CWENO method. In Spencer J. Sherwin, David Moxey, Joaquim Peiró, Peter E. Vincent, and Christoph Schwab, editors, *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2018*, pages 191–201, Cham, 2020. Springer International Publishing.
- [56] Xiangyu Y. Hu, Nikolaus A. Adams, and Chi-Wang Shu. Positivity-preserving method for high-order conservative schemes solving compressible euler equations. *Journal of Computational Physics*, 242:169–180, 2013.
- [57] International Workshop on High-Order CFD Methods. International Workshop on High-Order CFD Methods, October 2021.
- [58] P. Janhunen. A positive conservative method for magnetohydrodynamics based on hll and roe methods. *Journal of Computational Physics*, 160(2):649–661, 2000.
- [59] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted eno schemes. *Journal of computational physics*, 126(1):202–228, 1996.
- [60] Soshi Kawai. Divergence-free-preserving high-order schemes for magnetohydrodynamics: An artificial magnetic resistivity method. *Journal of Computational Physics*, 251:292–318, 2013.
- [61] D.I. Ketcheson. Highly efficient strong stability-preserving Runge-Kutta methods with low-storage implementations. *SIAM Journal on Scientific Computing*, 30(4):2113–2136, 2008.
- [62] J. F. B. M. Kraaijevanger. Contractivity of runge-kutta methods. *BIT Numerical Mathematics*, 31(3):482–528, Sep 1991.
- [63] L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon, and J.E. Flaherty. Shock detection and limiting with discontinuous galerkin methods for hyperbolic conservation laws. *Applied Numerical Mathematics*, 48(3):323–338, 2004. Workshop on Innovative Time Integrators for PDEs.
- [64] Dongwook Lee. A solution accurate, efficient and stable unsplit staggered mesh scheme for three dimensional magnetohydrodynamics. *Journal of Computational Physics*, 243:269–292, 2013.
- [65] Dongwook Lee and Anil E Deane. An unsplit staggered mesh scheme for multidimensional magnetohydrodynamics. *Journal of Computational Physics*, 228(4):952–975, 2009.

- [66] Youngjun Lee, Dongwook Lee, and Adam Reyes. A recursive system-free single-step temporal discretization method for finite difference methods. *Journal of Computational Physics: X*, 12:100098, 2021.
- [67] Randall J LeVeque. *Finite volume methods for hyperbolic problems*, volume 31. Cambridge university press, 2002.
- [68] Hui Li, Giovanni Lapenta, John M. Finn, Shengtai Li, and Stirling A. Colgate. Modeling the large-scale structures of astrophysical jets in the magnetically dominated limit. *The Astrophysical Journal*, 643(1):92, may 2006.
- [69] Ruo Li and Wei Zhong. A modified adaptive improved mapped weno method. *arXiv preprint arXiv:2011.03916*, 2020.
- [70] Ruo Li and Wei Zhong. Locally order-preserving mapping for weno methods. *Journal of Computational and Applied Mathematics*, page 115004, 2022.
- [71] Xu-Dong Liu, Stanley Osher, and Tony Chan. Weighted essentially non-oscillatory schemes. *Journal of computational physics*, 115(1):200–212, 1994.
- [72] Ian May. slug-cfd/kfvm-kokkos: hydro-apj, January 2024.
- [73] Ian C. T. May and Dongwook Lee. Kfvm-weno: A high-order accurate kernel-based finite volume method for compressible hydrodynamics, 2024.
- [74] A. Mignone, P. Tzeferacos, and G. Bodo. High-order conservative finite difference GLM-MHD schemes for cell-centered MHD. *Journal of Computational Physics*, 229(17):5896 – 5920, 2010.
- [75] M. L. Muñoz-Ruiz, C. Parés, and M. J. Castro Díaz. Convergence of path-conservative numerical schemes for hyperbolic systems of balance laws. In Gunilla Kreiss, Per Lötstedt, Axel Målqvist, and Maya Neytcheva, editors, *Numerical Mathematics and Advanced Applications 2009*, pages 675–682, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
- [76] Aronszajn N. Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404, 1950.
- [77] Steven A. Orszag and Cha-Mei Tang. Small-scale structure of two-dimensional magnetohydrodynamic turbulence. *Journal of Fluid Mechanics*, 90(1):129–143, 1979.
- [78] K.G. Powell. An approximate Riemann solver for magnetohydrodynamics (that works more than one dimension). Technical report, DTIC Document, 1994.

- [79] Hendrik Ranocha, Lisandro Dalcin, Matteo Parsani, and David I. Ketcheson. Optimized runge-kutta methods with automatic step size control for compressible computational fluid dynamics. *Communications on Applied Mathematics and Computation*, 4(4):1191–1228, Dec 2022.
- [80] C.E. Rasmussen and C.K.I. Williams. *Gaussian Processes for Machine Learning*. Adaptive Computation And Machine Learning. MIT Press, 2005.
- [81] Steven I Reeves, Dongwook Lee, Adam Reyes, Carlo Graziani, and Petros Tzeferacos. An application of gaussian process modeling for high-order accurate adaptive mesh refinement prolongation. *Communications in Applied Mathematics and Computational Science*, 17(1):1–41, 2022.
- [82] Adam Reyes, Dongwook Lee, Carlo Graziani, and Petros Tzeferacos. A new class of high-order methods for fluid dynamics simulations using gaussian process modeling: One-dimensional case. *Journal of Scientific Computing*, 76(1):443–480, 2018.
- [83] Adam Reyes, Dongwook Lee, Carlo Graziani, and Petros Tzeferacos. A variable high-order shock-capturing finite difference method with GP-WENO. *Journal of Computational Physics*, 381:189–217, 2019.
- [84] Adam Reyes, Dongwook Lee, Carlo Graziani, and Petros Tzeferacos. A variable high-order shock-capturing finite difference method with gp-weno. *Journal of Computational Physics*, 381:189–217, 2019.
- [85] P. L. Roe and D. S. Balsara. Notes on the eigensystem of magnetohydrodynamics. *SIAM Journal on Applied Mathematics*, 56(1):57–67, 1996.
- [86] Ravi Samtaney. Suppression of the Richtmyer–Meshkov instability in the presence of a magnetic field. *Physics of Fluids*, 15(8):L53–L56, 06 2003.
- [87] Robert Schaback. Native hilbert spaces for radial basis functions i. In Manfred W. Müller, Martin D. Buhmann, Detlef H. Mache, and Michael Felten, editors, *New Developments in Approximation Theory*, pages 255–282, Basel, 1999. Birkhäuser Basel.
- [88] Robert Schaback. Small errors imply large evaluation instabilities. *Advances in Computational Mathematics*, 49(2):25, Mar 2023.
- [89] Robert Schaback and Holger Wendland. Kernel techniques: From machine learning to meshless methods. *Acta Numerica*, 15:543–639, 2006.

- [90] M. Semplice, A. Coco, and G. Russo. Adaptive mesh refinement for hyperbolic systems based on third-order compact weno reconstruction. *Journal of Scientific Computing*, 66(2):692–724, Feb 2016.
- [91] Jeongbhin Seo and Dongsu Ryu. How-mhd: A high-order weno-based magnetohydrodynamic code with a high-order constrained transport algorithm for astrophysical applications. *The Astrophysical Journal*, 953(1):39, aug 2023.
- [92] Varun Shankar, Grady B Wright, Robert M Kirby, and Aaron L Fogelson. A radial basis function (RBF)-finite difference (FD) method for diffusion and reaction–diffusion equations on surfaces. *Journal of Scientific Computing*, 63(3):745–768, 2015.
- [93] 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.
- [94] Jing Shi, Yong-Tao Zhang, and Chi-Wang Shu. Resolution of high order weno schemes for complicated flow structures. *Journal of Computational Physics*, 186(2):690–696, 2003.
- [95] 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.
- [96] Chi-Wang Shu. High order weighted essentially nonoscillatory schemes for convection dominated problems. *SIAM review*, 51(1):82–126, 2009.
- [97] 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.
- [98] Thomas Sonar. Optimal recovery using thin plate splines in finite volume methods for the numerical solution of hyperbolic conservation laws. *IMA Journal of Numerical Analysis*, 16(4):549–581, 10 1996.
- [99] Seth C. Spiegel, HT Huynh, and James R. DeBonis. *A Survey of the Isentropic Euler Vortex Problem using High-Order Methods*. 2015.
- [100] Raymond J. Spiteri and Steven J. Ruuth. A new class of optimal high-order strong-stability-preserving time discretization methods. *SIAM Journal on Numerical Analysis*, 40(2):469–491, 2002.

- [101] James M Stone, Thomas A Gardiner, Peter Teuben, John F Hawley, and Jacob B Simon. Athena: a new code for astrophysical MHD. *The Astrophysical Journal Supplement Series*, 178(1):137, 2008.
- [102] James M Stone, Kengo Tomida, Christopher J White, and Kyle G Felker. The athena++ adaptive mesh refinement framework: Design and magnetohydrodynamic solvers. *The Astrophysical Journal Supplement Series*, 249(1):4, 2020.
- [103] Geoffrey Ingram Taylor and Albert Edward Green. Mechanism of the production of small eddies from large ones. *Proceedings of the Royal Society of London. Series A-Mathematical and Physical Sciences*, 158(895):499–521, 1937.
- [104] Eleuterio F Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.
- [105] G. Tóth. The $\nabla \cdot B = 0$ Constraint in Shock-Capturing Magnetohydrodynamics Codes. *Journal of Computational Physics*, 161(2):605–652, 2000.
- [106] Christian Trott, Luc Berger-Vergiat, David Poliakoff, Sivasankaran Rajamanickam, Damien Lebrun-Grandie, Jonathan Madsen, Nader Al Awar, Milos Gligoric, Galen Shipman, and Geoff Womeldorff. The kokkos ecosystem: Comprehensive performance portability for high performance computing. *Computing in Science Engineering*, 23(5):10–18, 2021.
- [107] Christian R. Trott, Damien Lebrun-Grandié, Daniel Arndt, Jan Ciesko, Vinh Dang, Nathan Ellingwood, Rahulkumar Gayatri, Evan Harvey, Daisy S. Hollman, Dan Ibanez, Nevin Liber, Jonathan Madsen, Jeff Miles, David Poliakoff, Amy Powell, Sivasankaran Rajamanickam, Mikael Simberg, Dan Sunderland, Bruno Turcksin, and Jeremiah Wilke. Kokkos 3: Programming model extensions for the exascale era. *IEEE Transactions on Parallel and Distributed Systems*, 33(4):805–817, 2022.
- [108] Bram Van Leer. Upwind and high-resolution methods for compressible flow: From donor cell to residual-distribution schemes. In *16th aiaa computational fluid dynamics conference*, page 3559, 2006.
- [109] Wim M Van Rees, Anthony Leonard, Dale I Pullin, and Petros Koumoutsakos. A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high reynolds numbers. *Journal of Computational Physics*, 230(8):2794–2805, 2011.
- [110] Wim M. van Rees, Anthony Leonard, D.I. Pullin, and Petros Koumoutsakos. A comparison of vortex and pseudo-spectral methods for the simulation of periodic

- vortical flows at high reynolds numbers. *Journal of Computational Physics*, 230(8):2794–2805, 2011.
- [111] Rong Wang, Hui Feng, and Cong Huang. A new mapped weighted essentially non-oscillatory method using rational mapping function. *Journal of Scientific Computing*, 67(2):540–580, 2016.
- [112] Holger Wendland. *Scattered Data Approximation*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2004.
- [113] Kailiang Wu and Chi-Wang Shu. Provably positive high-order schemes for ideal magnetohydrodynamics: analysis on general meshes. *Numerische Mathematik*, 142(4):995–1047, 2019.
- [114] Kailiang Wu and Chi-Wang Shu. Geometric quasilinearization framework for analysis and design of bound-preserving schemes. *SIAM Review*, 65(4):1031–1073, 2023.
- [115] Xiangxiong Zhang and Chi-Wang Shu. On positivity-preserving high order discontinuous galerkin schemes for compressible euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23):8918–8934, 2010.
- [116] Zhuang Zhao, Jun Zhu, Yibing Chen, and Jianxian Qiu. A new hybrid weno scheme for hyperbolic conservation laws. *Computers & Fluids*, 179:422–436, 2019.
- [117] Ye Zhou. Rayleigh–Taylor and Richtmyer–Meshkov instability induced flow, turbulence, and mixing. I. *Physics Reports*, 720-722:1–136, 2017. Rayleigh–Taylor and Richtmyer–Meshkov instability induced flow, turbulence, and mixing. I.