

Towards a New-generation Numerical Scheme for the Compressible Navier-Stokes Equations with the Active Flux Method

by

Fanchen He

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Applied and Interdisciplinary Mathematics)
in the University of Michigan
2021

Doctoral Committee:

Professor Smadar Karni, Co-Chair
Emeritus Professor Philip Roe, Co-Chair
Associate Professor Krzysztof J. Fidkowski
Professor Robert Krasny

Fanchen He
wzfche@umich.edu
ORCID iD: 0000-0002-9479-2410

© Fanchen He 2021
All Rights Reserved

DEDICATION

To My Parents.

ACKNOWLEDGMENTS

I find myself extremely lucky to have the best advisers Prof Philip Roe and Prof Smadar Karni. I would like to express my sincere gratitude to them. Prof Roe's remarkable insight, impressive intuition, and great erudition in the area of computational fluid dynamics make my journey towards the PhD degree exciting and enjoyable. The guidance and support from Prof Karni always inspire me and give me infinite strength and determination to move forward. I appreciate the freedom and trust they grant me to grow my own ideas. I also thank my dissertation committee members, Prof Krzysztof Fidkowski and Prof Robert Krasny, for their helpful comments and feedback.

Teaching was an important part of my life in graduate school. I am grateful to the Department of Mathematics for the opportunity of teaching undergraduate courses since my first year, which helps me establish confidence and valuable skills for my career, e.g., public speaking, communication, leadership.

Finally, I want to thank my parents for their unwavering support and encouragement over the years.

This work was supported in part by NSF Grant DMS 1417053 and by a University of Michigan Rackham Dissertation Fellowship.

TABLE OF CONTENTS

Dedication	ii
Acknowledgments	iii
List of Figures	vi
List of Tables	viii
List of Appendices	ix
Abstract	x

CHAPTER

1	Introduction	1
	1.1 A Brief Survey of the Active Flux Method	3
	1.2 Thesis Overview	4
2	The Active Flux Method and the Extension to a Family of Arbitrary-order Schemes	6
	2.1 The General Paradigm of the Active Flux Method	6
	2.2 Introduction: Why Should We Eliminate the Bubble Functions?	7
	2.3 The Numerical Discretization	8
	2.3.1 One-dimensional Space	8
	2.3.2 Two-dimensional Space	11
	2.4 Some Analysis on the Idea of Discrepancy Distribution	14
	2.4.1 Fourier Analysis	14
	2.4.2 Analysis Based on the Taylor Expansion: What Information Does the Discrepancy Carry?	17
	2.5 Numerical Experiments	19
	2.5.1 One-dimensional Space	19
	2.5.1.1 Linear Advection	20
	2.5.1.2 The Burgers Equation	20
	2.5.1.3 The Euler Equations	21
	2.5.2 Two-dimensional Space	22
	2.5.2.1 Linear Advection	23
	2.5.2.2 Linear Acoustics	23
	2.5.2.3 Pressureless Euler Equations	26
3	Operator Splitting and Treatment of Spatial Derivatives	30
	3.1 The First Nontrivial Example: the Euler Equations	30
	3.2 Treatment of Spatial Derivatives: Ideas Inspired by the Solution to the Wave Equation	33

3.3	Numerical Results	37
4	Boundary Conditions, Domain of Complicated Geometry and Stability in the General Active Flux Method	41
4.1	Boundary Conditions	41
4.1.1	Predictor	42
4.1.1.1	Treatment of Complicated Geometry	42
4.1.1.2	Imposing the Boundary Conditions	44
4.1.2	Corrector	44
4.2	Stability	44
4.3	Numerical Results for Advanced Test Cases	46
4.3.1	Gaussian Bump	47
4.3.2	Inviscid Flow around the Cylinder	50
4.3.3	Inviscid Flow around the NACA0012 Airfoil	51
5	Including Viscosity I: A Traditional Strategy	56
5.1	A Benchmark Scheme for Viscosity and Diffusion	56
5.1.1	Approximation of Derivatives	57
5.1.2	The Benchmark Numerical Scheme for the Heat Equation	59
5.1.3	The Benchmark Numerical Scheme for the Navier-Stokes Equations	59
5.2	Numerical Experiments	60
5.2.1	Transient Problems of the Heat Equation	60
5.2.2	Steady-State Problems of the Heat Equation	61
5.2.3	The Steady Viscous Flow around A Cylinder	62
6	Including Viscosity II: The Hyperbolic Reformulations	64
6.1	Hyperbolic Reformulations of Equations	65
6.1.1	The Hyperbolic Heat Equations	65
6.1.2	The Hyperbolic Viscosity and the Decomposition of Waves	67
6.1.3	The Hyperbolic Navier-Stokes Equations	69
6.2	Numerical Results	70
6.2.1	Steady State of the Heat Equation	70
6.2.2	Steady Viscous Flow around a Cylinder	71
6.2.3	Steady Viscous Flow around the NACA0012 Airfoil	72
7	Conclusions and Future Work	74
	Appendices	75
	Bibliography	98

LIST OF FIGURES

2.1	The discretization in the Active Flux method of different order of accuracy in one-dimensional space. p represents the degree of the reconstructed polynomial on each cell.	8
2.2	The discretization in the Active Flux method of different accuracy in the two-dimensional space. p represents the degree of the reconstructed polynomial on each cell.	12
2.3	The restriction on the size of time step Δt . ($p = 2$)	13
2.4	The reference element in the two-dimensional space. p represents the degree of the reconstructed polynomial on each cell.	14
2.5	Plots of $ \lambda ^{1/\nu}$: the change of amplitude after the wave travels over a unit distance Δx ($p = 1$)	16
2.6	Plots of $ \lambda_i ^{1/\nu}$: the change of amplitude after the wave travels over a unit distance Δx ($p = 2$)	16
2.7	Plots of $ \lambda ^{1/\nu}$: examples of the distributions weights that lead to instability ($p = 1$)	17
2.8	Plots of $ \lambda_i ^{1/\nu}$: examples of the distributions weights that lead to instability ($p = 2$)	17
2.9	The convergence plots of the Active Flux schemes for linear advection equation with different order of accuracy.	20
2.10	The convergence plots of the Active Flux schemes for the Burgers equation with different order accuracy.	21
2.11	The convergence plots of the Active Flux schemes for the Euler equations with different order of accuracy.	22
2.12	The vertices in the regular mesh (a) are perturbed by $0.2h(\sin(10x + 1)^{10y+1}, \cos(10y + 1)^{10x+1})$ in (b).	22
2.13	Convergence rates of $p + 1$ -th-order schemes for the linear advection equation with different direction $\mathbf{a} = (\cos \theta, \sin \theta)$ on the unperturbed mesh.	24
2.14	Convergence rates of $p + 1$ -th-order schemes for the linear advection equation with different direction $\mathbf{a} = (\cos \theta, \sin \theta)$ on the perturbed mesh.	25
2.15	Convergence rates of $p + 1$ -th-order schemes for the equations of linear acoustics on the unperturbed mesh and the perturbed mesh.	27
2.16	Convergence rates of $p + 1$ -th-order schemes for the pressureless Euler equations on the unperturbed mesh and the perturbed mesh.	29
3.1	Accuracy of acoustic-formula-based derivative approximation strategies used in $p + 1$ -th-order schemes on the unperturbed mesh and the perturbed mesh.	39
3.2	Convergence rates of $p + 1$ -th-order schemes for the Euler equations on the unperturbed mesh and the perturbed mesh.	40

4.1	Numerical domain of dependence for evaluating derivatives at Node v	42
4.2	43
4.3	The sample mesh for the flow through a channel with smooth bump with $2 \times N \times 3N$ cells ($N = 16$).	47
4.4	The steady state solutions for the flow through a channel with smooth bump on a mesh of $6N^2$ cells ($N = 64$).	48
4.5	Flow through a channel with smooth bump: The steady-state pressure coefficient profiles on the bottom wall generated by different meshes.	49
4.6	Flow through a channel with smooth bump: Convergence plot of errors for $p = 1$	50
4.7	Mesh 1 for the cylinder case with 441 nodes and 829 cells	50
4.8	The steady state solutions for the flow around a cylinder on Mesh 3.	52
4.9	Flow around a cylinder: Convergence plot of entropy errors for $p = 1$	53
4.10	A sample mesh for the NACA0012 Airfoil case with 801 nodes and 1468 cells	53
4.11	Subsonic flow around the NACA0012 Airfoil with $M_\infty = 0.5$, $\alpha = 1^\circ$	54
4.12	Subsonic flow around the NACA0012 ($M_\infty = 0.5$, $\alpha = 1^\circ$): Convergence plot of entropy errors for $p = 1$	54
4.13	Transonic flow around the NACA0012 Airfoil with $M_\infty = 0.8$, $\alpha = 1.25^\circ$	54
4.14	Transonic flow around the NACA0012 Airfoil with $M_\infty = 0.85$, $\alpha = 1^\circ$	55
5.1	Accuracy of the alternative derivative approximation strategy for $p = 1$ on the unperturbed mesh and the perturbed mesh.	58
5.2	Convergence rates of the benchmark scheme for the transient solutions to the heat equation on the unperturbed mesh and the perturbed mesh.	61
5.3	Convergence rates of the benchmark scheme for the steady-state solutions to the heat equation on the unperturbed mesh and the perturbed mesh.	62
5.4	The steady state solutions for the viscous flow around a cylinder on Mesh 3.	63
6.1	Convergence rates of the steady-state solutions to the hyperbolic heat equation on the unperturbed mesh and the perturbed mesh.	71
6.2	The steady-state solutions for the viscous flow around a cylinder on Mesh 3 generated by the Hyperbolic Navier-Stokes equations.	72
6.3	The steady-state solutions for the viscous flow around the NACA0012 Airfoil on Mesh 3 generated by the Hyperbolic Navier-Stokes equations.	73
D.1	Refinement of an interior cell.	92
D.2	Refinement of a cell on the boundary.	92
E.1	Numerical solutions to the Buckley-Leverett equations generated by the Active Flux schemes with simple limiters.	95
E.2	Numerical solutions to the modified Sod's test generated by the Active Flux schemes with simple limiters.	97
E.3	Numerical solutions to Toro's test 3 generated by the Active Flux schemes with simple limiters.	97

LIST OF TABLES

2.1	Comparison of the number of DOFs per cell required by the AF method and the DG method. (AF = Active Flux, DG = Discontinuous Galerkin)	12
6.1	Comparison of the running time of the benchmark scheme and the one based on hyperbolic reformulation in solving the steady-state problem of the heat equation.	71
E.1	The initial values of test cases for the Euler equations	96

LIST OF APPENDICES

A The Governing Equations of Fluid Dynamics 75
B Exact Solutions 78
C Elementary Analysis of the Hyperbolic Heat Equations in the One-dimensional Space . 87
D Notes about My Code 91
E Archive of Miscellaneous Investigation 93

ABSTRACT

Taking underlying physics into consideration is essential in the design of numerical schemes for fluid dynamics simulations. The Active Flux (AF) method is a novel class of fully-discrete numerical schemes that aim to truly reflect the multidimensional physics in gas dynamics. The original scheme has successfully achieved third-order accuracy for the Euler equations. In this project, we overhaul the original scheme and formulate a general paradigm for the Active Flux method, which allows a flexible and systematic extension to a family of schemes as well as a tidy treatment of general boundary conditions. We propose a novel and versatile notion of discrete conservation laws for the general AF method. We also apply the schemes to solving the Navier-Stokes equations. Numerical strategies for approximating derivatives and improving stability are discussed along the way, which significantly expands the capacity of the AF method.

Throughout the dissertation, we view the AF method as a predictor-corrector method. In the original AF scheme, assigning degrees of freedom to the cell boundaries and reconstructing continuous piecewise functions are key to capture the multidimensional physics in the inviscid flows at the prediction stage. However, having degrees of freedom (DOFs) shared by neighboring cells poses difficulty in enforcing conservation laws in the correction stage. We introduce a novel treatment of discrete conservation laws for the AF method. Instead of enforcing strict local conservation for each cell, we distribute to the nodal values a “discrepancy” equal to the difference between conservative and non-conservative updates, which replaces expensive and inconsistent bubble functions that were used in the original 3rd-order AF scheme. This notion of conservation

is compatible with a family of arbitrary-order spatial discretization and independent of the predictor. Thus we are able to systematically extend the AF method to a family of schemes of various order of accuracy and apply them to different problems governed by conservation laws. Analysis in one space dimension and abundant test cases, including the Euler equations, in both one and two space dimensions, are presented to verify the accuracy.

When we extend the Euler solver to the Navier-Stokes equations, we face difficulty from the stiffness caused by viscosity and heat conduction as well as approximating second-order derivatives. We introduce relaxed variables to reformulate the equations as a first-order hyperbolic system. The hyperbolic model has the same steady-state solutions as the original one. Our numerical experiments indicate that, compared with the benchmark scheme that directly approximates second-order derivatives, using the hyperbolic reformulation in solving the steady-state problems significantly reduces the simulation time while yielding better accuracy.

In practice, most problems in fluid dynamics simulations come with domains that contain complex geometry. Within the framework that we propose for the AF method, we discuss a simple but effective implementation of general boundary conditions in predictors and correctors. Results generated by the second-order scheme from the AF family are presented for both inviscid and viscous flows on domains with curved boundaries.

CHAPTER 1

Introduction

As computational power advances rapidly, computational fluid dynamics (CFD) plays an increasingly important role in fundamental research of fluid mechanics and simulation-based engineering design processes. It provides an affordable and useful replacement of expensive or even impossible experiments in the real world. For example, with the aid of CFD simulation, the aerospace design process is accelerated while the cost and risk of testing are significantly reduced [42]. Valuable physical insight for modeling complicated flow phenomena, e.g., turbulence, can be extracted from the data generated by direct numerical simulation (DNS) [28]. These tasks require a large amount of computation to get physical details resolved as much as possible. Thus robust numerical algorithms that have a good balance of accuracy and efficiency as well as compatibility with exascale-computing, are critical for the practical use of CFD.

In this project, we focus on developing new-generation numerical tools for simulation of flows by the compressible Navier-Stokes equations, which consist of the conservation laws of mass, momentum, and energy, plus constitutive equations derived from assumptions in thermodynamics. (See Appendix A for the derivation of the equations.) When designing numerical schemes for fluids simulations, fully capturing the underlying physics is vital to success. The conservation laws are among the most apparent and important ones. The Lax-Wendroff theorem states that the solution to which numerical solutions converge is a weak solution to the partial differential equation [23].

A discrete version of the conservation laws can be defined directly on cell averages when numerical schemes use discontinuous representation of solutions, e.g., Finite Volume (FV) method

and Discontinuous Galerkin (DG) method. Indeed, the current mainstream commercial CFD code for conservation laws is based on 2nd-order finite volume methods, which are robust and versatile. High-order (≥ 3) FV methods do exist but have to use large stencils and thus they are not ideal for parallelization. High-order accuracy could be also achieved with discontinuous Galerkin methods. However, introducing too many degrees of freedom per element not only increases the amount of computation, but also brings about challenges from memory requirements, especially for three-dimensional cases. There are also strict restrictions on the time step size for explicit methods when spatial accuracy is very high. High-order-accurate temporal evolution also incurs more computational cost.

The physics in multidimensional gas dynamics is much richer than conservation laws. However, numerical schemes using discontinuous representation of solutions are flawed in evaluating fluxes that pass from each cell to its neighbors. Due to the discontinuity at the cell boundaries, a Riemann solver is required. These Riemann problems reflect only the one-dimensional wave structures. For example, when considering the acoustic system, we will be led to d'Alembert's solution consisting of two waves moving with equal speed in opposite directions, which are no different from advective waves [36]. However, in the multidimensional setup, the acoustic wave propagates in all directions, which is essentially distinguished from the advective wave.

So, why not turn to numerical schemes that use continuous representation of solutions? Researchers pursued numerical schemes based on continuous reconstruction, e.g., the Cell Vertex method [22], the Residual Distribution (RD) method [2], the Multi-moment Finite Volume method [21], hoping to lower the cost of computation. Among them, the Active Flux (AF) method developed by Roe and his students Eymann, Maeng, Fan in [36, 13, 26, 16] is a promising candidate designed to fully reflect multidimensional physics.

1.1 A Brief Survey of the Active Flux Method

The original Active Flux (AF) method is a novel 3rd-order fully-discrete explicit scheme introduced to capture multidimensional physics in flows governed by the Euler equations by splitting the evolution of the solution into advective and acoustic components and treating them separately. In two-dimensional space, the AF scheme works on triangle meshes where DOFs are assigned to the vertices, midpoints of edges, and cell average. Each triangular cell has access to 7 DOFs. Continuous piecewise quadratic functions plus bubbles¹ are reconstructed from the discrete data to represent the numerical solutions. Since all DOFs that represent nodal values on the cell boundaries are shared by adjacent cells, on average, each cell has $3/2 + 3/6 + 1 = 3$ DOFs. The potential extension of such type spatial discretization to three-dimensional space would significantly mitigate the curse of dimensionality².

In the evolution of numerical solutions from time t to $t + \Delta t$, there are two stages. The first step is to update the nodal values³ of variables on the cell boundaries by using a proper combination of the third-order numerical schemes developed for advection [26] and acoustics [16]. With the nodal values of variables, the temporal integral⁴ of fluxes can be computed to update the cell average. In the early development [26, 13, 16, 27, 14, 40], this is how the conservation laws were enforced. After constructing piecewise quadratic functions plus bubbles from the updated data, the scheme is ready for the next iteration. The stencil remains compact in temporal evolution. Numerical experiments on problems with simple boundary conditions [27, 14, 40, 36] indicate that the decomposition of physics plus the time marching strategy was successful. The Euler solver based on the Active Flux method is more accurate than the prevailing second-order finite volume methods in practical use and more computationally efficient than other comparable third-order

¹The bubble function is a polynomial of degree 3 that vanishes on the cell boundary. In the reference cell, the bubble function is given by $xy(1 - x - y)$.

²Sharing DOFs with neighboring cells does not change the fact that the number of DOFs has asymptotic exponential growth ($\sim O(p^d)$) with the number of dimensions d when the degrees p of reconstructed polynomials are large. However, in practical use, p is usually small, e.g., $p < 5$, where the reduction of the number of DOFs is manifest.

³Since the nodal values on cell boundaries are directly predicted instead of being estimated by Riemann solvers, this method gets the name “Active Flux”.

⁴To achieve 3rd-order accuracy, the nodal values at $t + \Delta t/2$ are also required for the integral.

schemes [39].

This is not the end of the story. No robust boundary treatment has been developed with the original AF scheme; The discrete conservation law achieved via the bubble functions has its limitation. The extension to viscous compressible flows needs to be figured out. These are the major problems to be addressed by this dissertation.

In recent years, there are also other branches developed from the idea of the AF method [5, 20, 6, 4, 9]. Their investigation, which is more rigorous theoretically, focuses on one-dimensional space and the extension to two-dimensional space uses Cartesian meshes. Since their topics are not closely related to the main concerns of this dissertation, we omit the details here.

1.2 Thesis Overview

In this project, we overhaul the original scheme and formulate a general paradigm for the Active Flux method, which allows a flexible and systematic extension of the AF scheme and a tidy treatment of general boundary conditions. The major contributions of this dissertation include:

- A novel notion of discrete conservation laws that extends the AF method to a family of arbitrary-order schemes,
- A general treatment for common boundary conditions in compressible viscous flows and a numerical strategy to stabilize the 2nd-order scheme from the family,
- Extending the AF scheme for the Navier-Stokes equations via hyperbolic reformulation.

In Chapter 2, we formulate the AF scheme as a predictor-corrector method. We explain why it is necessary to get rid of the bubble functions and introduce a systematic treatment for the discrete conservation law for a family of arbitrary-order schemes. Analysis and numerical experiments are presented. Not all the problems have exact solutions to serve as the predictor. In Chapter 3, we dissect the operator splitting from the original AF scheme to seek a proper approximation of derivatives for gas dynamics simulations. The full Euler equations are used as a test problem to verify the high-order accuracy. Then we discuss numerical strategies to enforce boundary condi-

tions and stabilize the scheme, which are essential to expanding the capacity of the AF schemes in Chapter 4. The high-order treatment of curved boundary is not available yet, so we present the results generated by the 2nd-order scheme from the AF family. In Chapter 5, to include viscosity and heat conduction, we introduce a simple benchmark solver based on piecewise linear reconstruction that directly approximates 2nd-order derivatives. In Chapter 6, we illustrate the hyperbolic reformulation as a numerical strategy to speed up simulations for steady-state solutions. Numerical schemes that use piecewise linear reconstruction are used for validation. Most of the ideas from Chapter 4 to 6 are extendable for higher-order schemes. We discuss the conclusions and future work in Chapter 7. We also include the documentation of some miscellaneous investigations in the Appendix. Note that almost all of the numerical schemes presented in the thesis relate to either one- or two-dimensional cases. The three-dimensional case is expected to be very similar since our numerical discretization is systematic.

CHAPTER 2

The Active Flux Method and the Extension to a Family of Arbitrary-order Schemes

2.1 The General Paradigm of the Active Flux Method

The numerical evolution in AF is generally described as a predictor-corrector method. The predictor estimates the nodal values at $t^n + \Delta t$ from the data at t^n . For example, the exact solutions or method of characteristics can serve as a predictor. Usually the predictor is non-conservative. Thus we need the corrector to reconcile with the conservation laws. It could be simply adding bubble functions or a new notion of discrete conservation laws that we introduce in this chapter.

Algorithm 2.1 The general paradigm of the Active Flux method

- 1: Initialize discrete data U^0 .
 - 2: $n = 0$.
 - 3: **while** $t^n < T$ **do**
 - 4: $t^{n+1} = t^n + \Delta t$.
 - 5: Reconstruct continuous piecewise polynomials $u^{h,n}$ from the discrete data U^n .
 - 6: Evaluate nodal values U at $t^{n+k/p}$ with proper methods ($1 \leq k \leq p$). (**Predictor**)
 - 7: Numerically integrate the fluxes from t^n to t^{n+1} .
 - 8: Reconciliation with the conservation laws. (**Corrector**)
 - 9: $n = n + 1$.
 - 10: **end while**
-

2.2 Introduction: Why Should We Eliminate the Bubble Functions?

Our experiments show that a proper form of discrete conservation laws is essential to prevent local truncation error from accumulating. In the original 3rd-order AF method, an independent DOF for each cell is required to enforce the strict local conservation laws. For triangular meshes (2D) or tetrahedral meshes (3D), this entails introducing an extra DOF per cell and enriching the continuous piecewise quadratic reconstruction with bubble functions[26], which considerably multiplies the cost of computation. Furthermore, the bubble function approach is not robust: mesh alignment issues and instability near discontinuities have been reported [26, 19, 20]. The construction of bubble functions is also dimension-dependent and there is no systematic extension to numerical schemes of different order of accuracy.

We go back to basics and ask, is a strict local conservation law really necessary? The one-dimensional model we built for third-order schemes in our initial investigation [19] indicates that we can replace the cell-wise balance of flux by a flexible discrepancy distribution strategy without loss of accuracy. Several numerical experiments in [19] indicate this is also true for third-order schemes in two-dimensional space. Thus, constructing bubble functions is no longer necessary. In this chapter, we extend the idea of discrepancy distribution further and present a family of arbitrary-order Active Flux schemes. With reconstruction of continuous piecewise polynomials of degree p , we achieve global $p + 1$ -th-order accuracy. However, because degrees of freedom may relate to more than one element, the storage requirements are greatly reduced. Besides, the flexibility of discrepancy distribution enables us to design limiters when the solutions are almost discontinuous. See Appendix E.1. The new schemes generate satisfying solutions in shock tube tests, where the original AF-based Euler solver based on bubble functions was unstable.

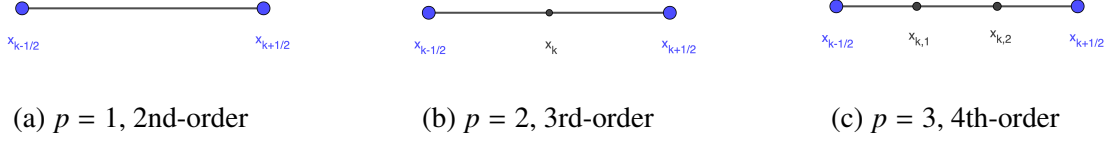


Figure 2.1: The discretization in the Active Flux method of different order of accuracy in one-dimensional space. p represents the degree of the reconstructed polynomial on each cell.

2.3 The Numerical Discretization

2.3.1 One-dimensional Space

For simplicity, we use a domain $\Omega = [0, 1]$ with periodic boundary conditions in this section to illustrate a novel treatment of discrete conservation laws and a family of numerical schemes built on top of it. We divide the domain $\Omega = [0, 1]$ into N cells $I_i = [x_{i-1/2}, x_{i+1/2}]$. To reconstruct a continuous piecewise polynomial of degree p for a given function u , we store $U_{i-1/2}$, the nodal value of u at the interface $x_{i-1/2}$, and $U_{i,k} = u(x_{i-1/2} + k/p\Delta x)$, $\Delta x = x_{i+1/2} - x_{i-1/2}$, for $1 \leq k \leq p-1$. See Figure 2.1. Then on each cell, we reconstruct polynomials of degree p via interpolation.

For $p = 1$, $\xi = (x - x_{i-1/2})/\Delta x$,

$$u_i^h(x) = U_{i-1/2}(1 - \xi) + U_{i+1/2}\xi.$$

For $p = 2$, $\xi = 2(x - x_{i-1/2})/\Delta x$,

$$u_i^h(x) = U_{i-1/2} \frac{(\xi - 1)(\xi - 2)}{2} + U_{i,1} \xi(2 - \xi) + U_{i+1/2} \frac{(\xi - 1)\xi}{2}.$$

For $p = 3$, $\xi = 3(x - x_{i-1/2})/\Delta x$,

$$u_i^h(x) = -U_{i-1/2} \frac{(\xi - 1)(\xi - 2)(\xi - 3)}{6} + U_{i,1} \frac{\xi(\xi - 2)(\xi - 3)}{2} - U_{i,2} \frac{\xi(\xi - 1)(\xi - 3)}{2} + U_{i+1/2} \frac{\xi(\xi - 1)(\xi - 2)}{6}.$$

To illustrate the discrete numerical evolution, we take the hyperbolic equation with a conservation law as an example.

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0.$$

Note that along the characteristics $(x(s), s)$ where $x'(s) = f'(u(x(s), s))$, u remains constant. That is, $u(x(s), s) \equiv u(x(0), 0)$ and $x(s) - x(0) = sf'(u(x(s), s))$. In the predictor stage, the nodal values at $(x, t + \Delta t)$ are estimated by the method of characteristics:

$$U^* = u^h(x - a\Delta t, t), \quad (2.1)$$

where $a = f'(\tilde{u})$ is the estimated wave speed of p -th-order accuracy. \tilde{u} is some approximation of $u(x, t + \Delta t)$. For linear advection, a is a constant independent of \tilde{u} . For nonlinear equations like the Burgers equation, \tilde{u} can be found by a few iterations of similar characteristics tracing. The CFL condition is given by $\nu = |a|\frac{\Delta t}{\Delta x} \leq \frac{1}{p}$.

Usually there is discrepancy between the cell average of reconstructed $U^{h,*}$ and one estimated by the local conservation law. We define the discrepancy as follows. Suppose we have predicted the U^* from the data at U^n , with $\Delta t = t^{n+1} - t^n$.

$$d\bar{U}_i = \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} u^h(x, t^n) - u^h(x, t^{n+1}) dx + \frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} f(u^h(x_{i-1/2}, t)) - f(u^h(x_{i+1/2}, t)) dt, \quad (2.2)$$

where the integrals are evaluated by numerical quadrature of sufficient accuracy. For $p + 1$ -th-order scheme, we use the Newton-Cotes formula of degree p . For the temporal integral of the flux $f(u)$ at interfaces $x = x_{i\pm 1/2}$, we need to estimate $u(x, t^n + k/p\Delta t)$, for $1 \leq k \leq p - 1$ with the same predictor.

In the corrector stage, we distribute the discrepancy to the nodal values with weights $(\alpha_1, \dots, \alpha_{p-1}, \beta_1, \beta_2)$.

$$\begin{aligned} U_{i,k}^{n+1} &= U_i^* + \alpha_k d\bar{U}_i, \quad 1 \leq k \leq p - 1, \\ U_{i+1/2}^{n+1} &= U_{i+1/2}^* + \frac{\beta_2 d\bar{U}_i \Delta x_i + \beta_1 d\bar{U}_{i+1} \Delta x_{i+1}}{\Delta x_i + \Delta x_{i+1}}. \end{aligned} \quad (2.3)$$

The conservation laws require the total discrepancy over the whole domain to be fully absorbed so that $\int_{\Omega} u^{h,n+1} dx = \int_{\Omega} u^{h,n}$. Let $\{A_j\}$ denote the coefficients in the Newton-Cotes formula. The corrector changes the integral of the numerical solution at $t = t^{n+1}$ by

$$\begin{aligned}
& \int_{\Omega} (u^{h,n+1} - u^{h,*}) dx \\
&= \sum_i \Delta x_i \left((U_{i-1/2}^{n+1} - U_{i-1/2}^*) A_0 + \sum_{k=1}^{p-1} (U_{i,k}^{n+1} - U_{i,k}^*) A_k + (U_{i+1/2}^{n+1} - U_{i+1/2}^*) A_p \right) \\
&= \sum_i \Delta x_i d\bar{U}_i \left(\beta_1 A_0 + \sum_{k=1}^{p-1} \alpha_k A_k + \beta_2 A_p \right).
\end{aligned} \tag{2.4}$$

Thus, we have the following restriction on the distribution weights.

$$\beta_1 A_0 + \sum_{k=1}^{p-1} \alpha_k A_k + \beta_2 A_p = 1. \tag{2.5}$$

For example,

when $p = 1, \beta_1 + \beta_2 = 1$;

when $p = 2, \beta_1 + 4\alpha + \beta_2 = 6$;

when $p = 3, \beta_1 + 3\alpha_1 + 3\alpha_2 + \beta_2 = 8$.

Later analysis and numerical experiments will show that the condition Eq. (2.5) is essential to achieve global $p + 1$ -th-order accuracy. It is worth noting that, unless $\beta_1 = \beta_2 = 0$, the strict local balance of flux is not satisfied for a specific cell because the local discrepancy is not fully

absorbed, although the conservation law holds for the whole domain.

$$\begin{aligned}
& \int_{x_{i-1/2}}^{x_{i+1/2}} (u^{h,n+1} - u^{h,*}) dx \\
&= \Delta x_i \left(\left(\frac{\beta_2 d\bar{U}_{i-1} \Delta x_{i-1} + \beta_1 d\bar{U}_i \Delta x_i}{\Delta x_{i-1} + \Delta x_i} \right) A_0 + \sum_{k=1}^{p-1} (\alpha_k d\bar{U}_i) A_k + \left(\frac{\beta_2 d\bar{U}_i \Delta x_i + \beta_1 d\bar{U}_{i+1} \Delta x_{i+1}}{\Delta x_i + \Delta x_{i+1}} \right) A_p \right) \\
&= \Delta x_i d\bar{U}_i \left(\beta_1 A_0 + \sum_{k=1}^{p-1} \alpha_k A_k + \beta_2 A_p \right) \\
&+ \Delta x_i \left(\left(\frac{(\beta_2 d\bar{U}_{i-1} - \beta_1 d\bar{U}_i) \Delta x_{i-1}}{\Delta x_{i-1} + \Delta x_i} \right) A_0 + \left(\frac{(\beta_1 d\bar{U}_{i+1} - \beta_2 d\bar{U}_i) \Delta x_{i+1}}{\Delta x_i + \Delta x_{i+1}} \right) A_p \right).
\end{aligned} \tag{2.6}$$

Thus we also call the discrepancy distribution strategy as a loose form of discrete conservation law.

The original Active Flux method with the bubble function is the case with $p = 2$, $\alpha = 1.5$, $\beta_1 = \beta_2 = 0$. We note that many numerical schemes for one-dimensional linear advection are actually equivalent although the interpretation is different. The original third-order Active Flux method in one-dimensional space is identical to CIP-CSL2 [21], an ADER method [20], as well as Scheme V by van Leer [45]. But we believe our interpretation is more systematic and flexible for further extension of the Active Flux method.

2.3.2 Two-dimensional Space

For multidimensional space, we decompose the domain into cells of simplices and reconstruct piecewise multivariate polynomials from nodal values. We consider a triangular mesh in two-dimensional space as an example. The reconstruction of a bivariate polynomial of degree p over a triangular cell requires $(p + 1)(p + 2)/2$ DOFs. Thus we assign the nodes as Figure 2.2 shows. It is worth noting that nodes on vertices are shared by 6 cells on average; nodes on edges are shared by 2 cells. The average number of DOFs per cell required by the AF method is considerably lower than the DG method when we approximate the numerical solutions on each cell by polynomials of

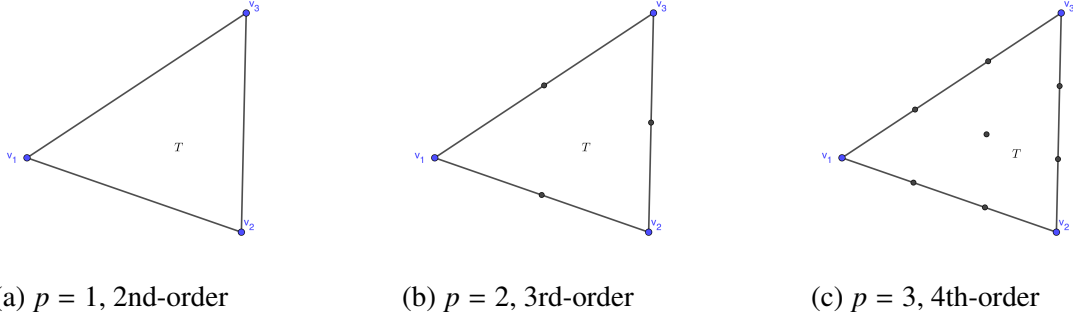


Figure 2.2: The discretization in the Active Flux method of different accuracy in the two-dimensional space. p represents the degree of the reconstructed polynomial on each cell.


the same degree p . See Table 2.1¹. The discretization of three-dimensional space is a natural and straightforward extension of the two-dimension example. For fixed p , a higher proportion of DOFs are shared and thus the gain in the number of DOFs per cell is even greater in three-dimensional space.

	$p = 1$	$p = 2$	$p = 3$
AF (2D)	$\approx \frac{3}{6} = 0.5$	$\approx \frac{3}{6} + \frac{3}{2} = 2$	$\approx \frac{3}{6} + 2(\frac{3}{2}) + 1 = 4.5$
DG (2D)	3	6	10
AF (3D) ²	$\approx (\frac{8}{8})/5 = 0.2$	$\approx (\frac{8}{8} + \frac{12}{4} + \frac{6}{2})/5 = 1.4$	$\approx (\frac{8}{8} + 2(\frac{12}{4} + \frac{6}{2}) + 4 + \frac{12}{2})/5 = 4.6$
DG (3D)	4	10	20

Table 2.1: Comparison of the number of DOFs per cell required by the AF method and the DG method. (AF = Active Flux, DG = Discontinuous Galerkin)

Like the one-dimensional case, each step of time evolution is described as predictor and corrector. The predictor step estimates the nodal values U^* at $t = t^{n+1}$ with $p + 1$ -th-order accuracy for the continuous piecewise polynomial u^h reconstructed from the data at U^n . For example, formulas of exact solutions are used in predictors for acoustics [15] and hyperbolic diffusion [18]; The method of characteristics is used for advection [26]. Let a denote the maximum wave speed. To make the scheme compact and ensure the physical domain of dependence covered by the numerical one, we

¹ Although DG methods may show superconvergence of rates like $2p + 1$ in norms that are defined “on the basis of the cell-averaged solution” [46], generally, numerical schemes that approximate the solutions with piecewise polynomials of degree p have order of accuracy no higher than $p + 1$ when the errors are measured by general function norms like L_2 -norm $\|u - u^h\|_2$. In this work, we focus on errors defined by general function norms.

²We consider a cube decomposed into 5 tetrahedrons. 

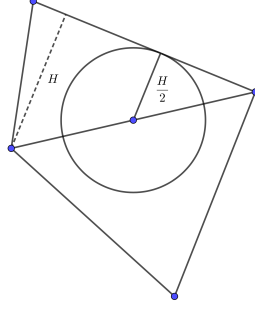


Figure 2.3: The restriction on the size of time step Δt . ($p = 2$)

restrict the time step by $\Delta t = \frac{1}{ap} \min_{T,e} H_{T,e}$, where $H_{T,e}$ represents the height of cell T rising from edge e . See Figure 2.3.

In the corrector step, we first evaluate the discrepancy of cell average on triangular cell T by Eq. (2.7)

$$d\bar{U}_T = \frac{-1}{S_T} \left(\int_{\partial T} \int_{t^n}^{t^{n+1}} \mathbf{f}(u^h) \cdot \mathbf{n} dt dl + \iint_T u^h(\mathbf{x}, t^{n+1}) - u^h(\mathbf{x}, t^n) d\sigma \right), \quad (2.7)$$

where $u^h(x, t^{n+1})$ is reconstructed from U^* and all integrals are evaluated with proper numerical quadrature. Then $d\bar{U}_T$ is distributed to the nodes v on the cell T with weights $w_{v,T}$.

$$U_v^{n+1} = U_v^* + \frac{\sum_{\{T: v \in T\}} w_{v,T} d\bar{U}_T S_T}{\sum_{\{T: v \in T\}} S_T}. \quad (2.8)$$

Let $A_{v,T}$ ($v \in T$) represent the contribution of nodal value at node v to cell average of T .

$$\frac{1}{S_T} \iint_T u^h(\mathbf{x}, t) d\sigma = \sum_{v \in T} A_{v,T} U_v.$$

In fact, $A_{v,T}$ only depends on v . We simply denote it by A_v . Without loss of generality, we map the cell T to the reference element whose three vertices are $(0, 0)$, $(0, 1)$, and $(1, 0)$. See Figure 2.4.

When $p = 1$, $A_0 = A_1 = A_2 = 1/3$;

when $p = 2$, $A_0 = A_2 = A_5 = 0$, $A_1 = A_3 = A_4 = 1/3$;

when $p = 3$, $A_0 = A_3 = A_9 = 1/30$, $A_1 = A_2 = A_4 = A_6 = A_7 = A_8 = 3/40$, $A_5 = 1/2$.

The conservation law requires $d\bar{U}_T$ to be fully absorbed into nodal values. Thus for any fixed

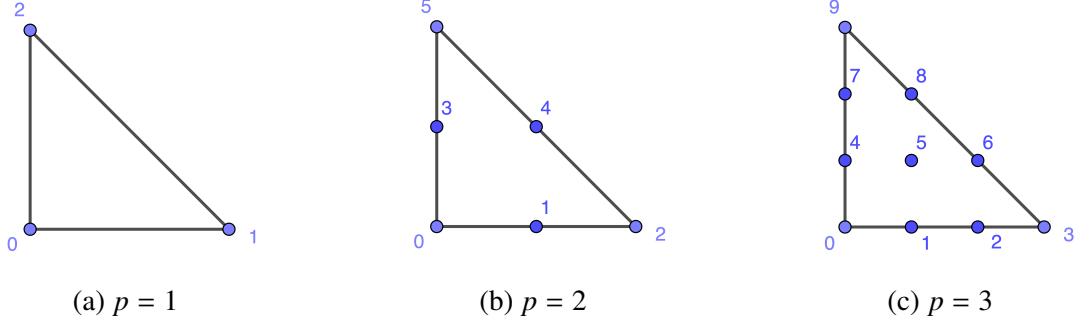


Figure 2.4: The reference element in the two-dimensional space. p represents the degree of the reconstructed polynomial on each cell.

cell T , $\{w_{v,T}\}$ is supposed to satisfy

$$\sum_{v \in T} w_{v,T} A_v = 1.$$

The simplest choice is $w_{v,T} = w = 1$.

2.4 Some Analysis on the Idea of Discrepancy Distribution

As pointed out in [37], the accuracy reflects the low-frequency properties of a scheme. In this section, we focus on the accuracy of the numerical schemes built on discrepancy distribution.

2.4.1 Fourier Analysis

To reveal the role played by the conservation law Eq. (2.5), we conduct Fourier Analysis for the numerical schemes for linear advection,

$$\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0.$$

Here we assume that each cell is of the same length Δx . For $p = 1$, we assume numerical solution at time t^n takes the form $U_{k+1/2}^n = e^{i\omega k \Delta x} r^n$. The numerical evolution is given by

$$U_{k+1/2}^{n+1} = \lambda U_{k+1/2}^n = \lambda^{n+1} e^{i\omega k \Delta x} r^0. \tag{2.9}$$

We write λ in the polar form $\lambda = |\lambda|e^{-i\omega\tilde{a}\Delta t}$, where \tilde{a} is a function of $\omega\Delta x$. If $\tilde{a} = a$, $U_{k+1/2}^{n+1}$ remains exact for all n . We expand $|\lambda|$ and \tilde{a}/a for small $\omega\Delta x$, which is corresponding to a fine mesh or smooth initial values.

$$\begin{aligned} |\lambda| &= 1 - \frac{\nu(\nu-1)(\beta_1 + \beta_2 - 2)}{2}(\omega\Delta x)^2 + O(|\omega\Delta x|^4), \\ \frac{\tilde{a}(\omega\Delta x)}{a} &= \frac{\arg \lambda}{-\nu\omega\Delta x} = 1 + \frac{(\nu-1)(3(\beta_1 + \beta_2)\nu - 4\nu - 3\beta_2 + 2)}{12}(\omega\Delta x)^2 + O(|\omega\Delta x|^4). \end{aligned} \quad (2.10)$$

The condition required by the conservation law, $\beta_1 + \beta_2 = 2$, corrects the amplitude of the wave propagation. Both the amplitude $|\lambda|$ and argument $-i\omega\Delta t\tilde{a}$ of λ have errors of $O(|\omega\Delta x|^3)$. Thus the numerical scheme is of global second-order accuracy when $\nu = a\Delta t/\Delta x$ satisfies the CFL condition, i.e., $0 < \nu < 1$.

For $p = 2$, we quote the results we presented in [19]. We assume that $(U_k^n, U_{k+1/2}^n) = e^{i\omega k\Delta x}(r_1^n, r_2^n)$.

Then the numerical solution at $t = t^{n+1}$ can be written as

$$\begin{pmatrix} U_k^{n+1} \\ U_{k+1/2}^{n+1} \end{pmatrix} = \mathbf{G} \begin{pmatrix} U_k^n \\ U_{k+1/2}^n \end{pmatrix} = e^{i\omega k\Delta x} \mathbf{G}^{n+1} \begin{pmatrix} r_1^0 \\ r_2^0 \end{pmatrix}. \quad (2.11)$$

Note that there are two eigenvalues of \mathbf{G} (See Figure 2.6). The primary eigenvalue is defined as the one closer to 1 as $\omega\Delta x$ approaches 0. Similarly, we expand the amplitude and the argument of the primary eigenvalue for small $\omega\Delta x$.

$$\begin{aligned} |\lambda| &= 1 + c_1(\omega\Delta x)^4 + O(|\omega\Delta x|^6), \\ \frac{\tilde{a}(\omega\Delta x)}{a} &= \frac{\arg \lambda}{-\nu\omega\Delta x} = 1 - \frac{1}{24} \frac{(2\nu-1)(2\nu-2)(\beta_1 + \beta_2 + 4\alpha - 6)}{(6\nu-2)(-\beta_1 - \beta_2 + 2\alpha) - 12} (\omega\Delta x)^2 + c_2(\omega\Delta x)^4 + O(|\omega\Delta x|^6), \end{aligned} \quad (2.12)$$

where c_1 and c_2 are complicated functions about α , β_1 , β_2 , ν . We notice that the condition Eq. (2.5) $4\alpha + \beta_1 + \beta_2 = 6$ is essential to correct the dispersion relation in the numerical scheme. The errors in the amplitude and argument are of fourth order and give us global third-order accuracy when the scheme is stable.

Generally, even-order schemes, e.g., the second-order scheme with $p = 1$, the corrector (the

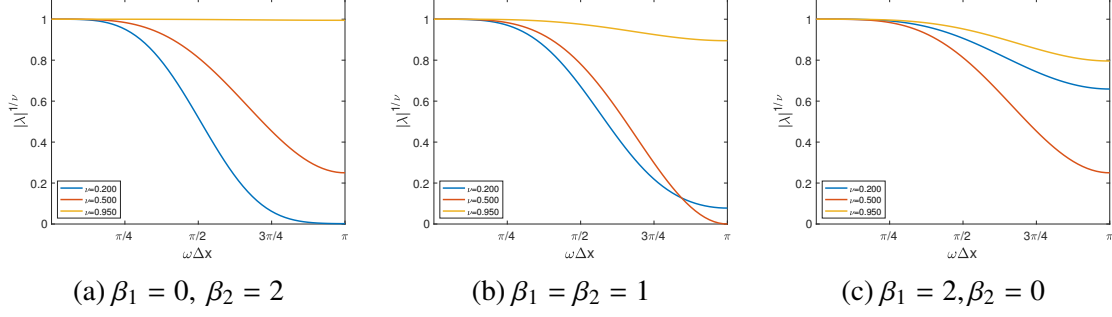


Figure 2.5: Plots of $|\lambda|^{1/\nu}$: the change of amplitude after the wave travels over a unit distance Δx ($p = 1$)

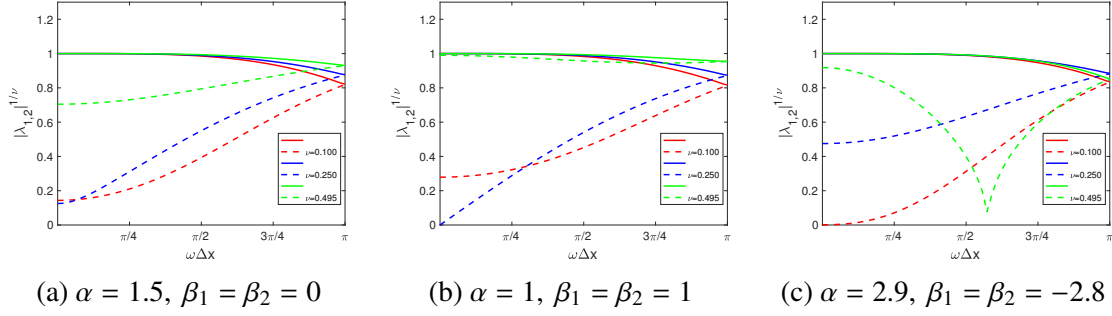


Figure 2.6: Plots of $|\lambda_i|^{1/\nu}$: the change of amplitude after the wave travels over a unit distance Δx ($p = 2$)

discrepancy distribution step) improves the accuracy by correcting the amplitude; odd-order schemes, like the case $p = 2$, the increased accuracy is due to the correction on dispersion relation.

Note that the discrete conservation law Eq. (2.5) does not guarantee stability. There are further restrictions on the distribution weights. Intuitively, we do not want the correction to significantly increase errors on nodal values. We present the change of amplitude after the wave travels over a distance of mesh size Δx for $p = 1$ and $p = 2$, i.e., $|\lambda|^{1/\nu}$. Figure 2.5 suggests that when $p = 1$ the numerical schemes are stable for all $0 < \nu < 1$ with $0 \leq \beta_1 \leq 2$. Figure 2.6 originally appeared in [19] where we concluded from an idea inspired by the Residual Distribution method [3] that we require $1 \leq \alpha \leq 3$ to make the numerical scheme stable for all allowed CFL number ν , i.e., $0 < \nu < 1/2$. We also attach the same plots for different distribution weights which make the schemes unstable. See Figure 2.7 and Figure 2.8³.

³We originally published Figure 2.8 in [19].

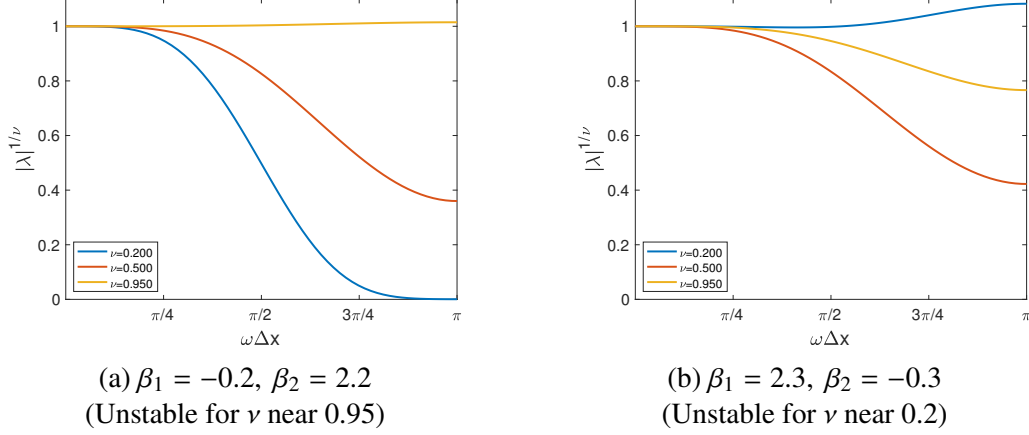


Figure 2.7: Plots of $|\lambda|^{1/\nu}$: examples of the distributions weights that lead to instability ($p = 1$)

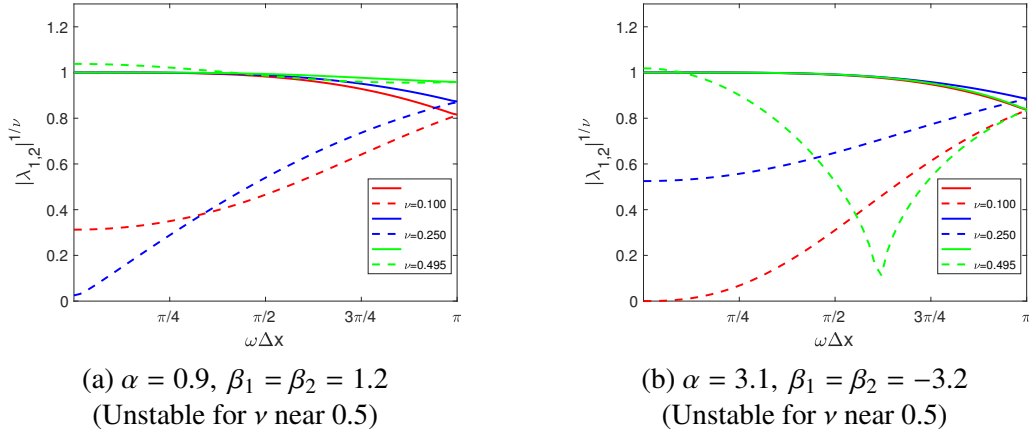


Figure 2.8: Plots of $|\lambda_i|^{1/\nu}$: examples of the distributions weights that lead to instability ($p = 2$)

2.4.2 Analysis Based on the Taylor Expansion: What Information Does the Discrepancy Carry?

For more general cases, we conduct analysis based on the Taylor expansion. We decompose the discrepancy defined by Eq. (2.2) into two parts: the error caused by the discrete form of the discrepancy being apply on the exact solutions, which is similar to the local truncation error, and the contribution from the error contained in u^h that was reconstructed from the predicted values.

First, we notice that evaluating the discrete discrepancy Eq. (2.2) on the exact solution $u(x, t)$ gives higher-order and thus negligible error. The Newton-Cotes formula of degree p has error of $O(h^{k(p)})$, where $k(p) = p + 2$ for odd p , $k(p) = p + 3$ for even p . Suppose $\Delta t \sim \Delta x$. When we plug

the exact solution $u(x, t)$ into the formula Eq. (2.2), for some η between t^n and t^{n+1} , ξ between $x_{i-\frac{1}{2}}$ and $x_{i+\frac{1}{2}}$,

$$\begin{aligned} d\bar{U}_i &= \frac{C\Delta x^{k(p)}}{\Delta x} \left(\frac{\partial^{k(p)-1}u}{\partial x^{k(p)-1}} \Big|_{(\xi(t^n), t^n)} - \frac{\partial^{k(p)-1}u}{\partial x^{k(p)-1}} \Big|_{(\xi(t^{n+1}), t^{n+1})} \right) \\ &\quad + \frac{C\Delta t^{k(p)}}{\Delta x} \left(\frac{\partial^{k(p)-1}u}{\partial t^{k(p)-1}} \Big|_{(x_{i-\frac{1}{2}}, \eta(x_{i-\frac{1}{2}}))} - \frac{\partial^{k(p)-1}u}{\partial t^{k(p)-1}} \Big|_{(x_{i+\frac{1}{2}}, \eta(x_{i+\frac{1}{2}}))} \right) \\ &= O(\Delta x^{k(p)}). \end{aligned}$$

Then we consider the numerical solution u^h reconstructed from the prediction that uses the method of characteristics. If the prediction U^* has error e , then the error of flux is of the same order if we assume f' is bounded.

$$f(U^*) - f(u) = f(u + e) - f(u) = f'(u)e + O(e^2).$$

Suppose the discrete data U^n are accurate at $t = t^n$. Then in the numerical evolution to t^{n+1} , all the prediction based on the method of characteristics has error of $O(\Delta x^{p+1})$ when piecewise polynomial of degree p reconstruction of U^n is used. Let e^h be the reconstruction from errors of the predicted nodal values. e^h vanishes at t^n . It is not hard to show that in smooth cases, $e^h(x, t)$ satisfies a Lipschitz condition. Thus we have

$$\begin{aligned} d\bar{U}_i &= \frac{1}{\Delta x} \int_{t_n}^{t^{n+1}} f'(u(x_{i-1/2}, t))e^h(x_{i-1/2}, t) - f'(u(x_{i-1/2}, t))e^h(x_{i+1/2}, t) dt, \\ &\quad + \frac{1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} -e^h(x, t^{n+1}) dx + O(\Delta x^{k(p)}) \\ &= \frac{-1}{\Delta x} \int_{x_{i-1/2}}^{x_{i+1/2}} e^h(x, t^{n+1}) dx + O(\Delta x^{p+1}). \end{aligned} \tag{2.13}$$

This means the discrepancy is actually the cell average of the error that was brought in by the predictor. We interpret the result as that the corrector step catches the high-order error in the prediction stage and projects it into a piecewise constant function with the least-square loss of information. The discrepancy distribution is equivalent to assimilating the projected error into the discrete data. Our numerical experiments indicate that this is the key step to preventing the local

truncation error accumulation. Thus we achieve global $p + 1$ -th-order accuracy. It is worth noting that $d\bar{U}_i$ also satisfies a Lipschitz condition when the exact solution is smooth. When we correct nodal values $U_{i-1/2}$ at interfaces, it does not matter whether we take more information from the left cell or the right cell. In other words, the difference between β_1 and β_2 does not have much influence on the accuracy in the smooth cases. However, when the solution contains a large slope, by allowing the distribution weights to vary for different cells, there is extra space to refine the discrepancy distribution process, which enables the design of limiters (see the next Appendix E).

2.5 Numerical Experiments

2.5.1 One-dimensional Space

We conduct numerical experiments to verify the global $p + 1$ -th-order accuracy of the family of numerical schemes. In the following examples the domain is $\Omega = [0, 1]$ with periodic boundary conditions. For linear advection and the Burgers equation, where the analytical solutions are known, the error is measured in the L_1 -norm:

$$E = \frac{1}{pN} \sum_{i=1}^N (|U_{i-1/2} - u(x_{i-1/2})| + \sum_{k=1}^{p-1} |U_{i,k} - u(x_{i,k})|). \quad (2.14)$$

For the Euler equations with smooth solutions, we compare the numerical solutions with the reference solutions generated by a second-order Finite Volume method on a mesh of 1200 cells. The errors are measured by the L_1 -norm of cell average.

$$E = \frac{1}{N} \sum_{i=1}^N (|\bar{U}_i - \bar{U}_i^{\text{ref}}|). \quad (2.15)$$

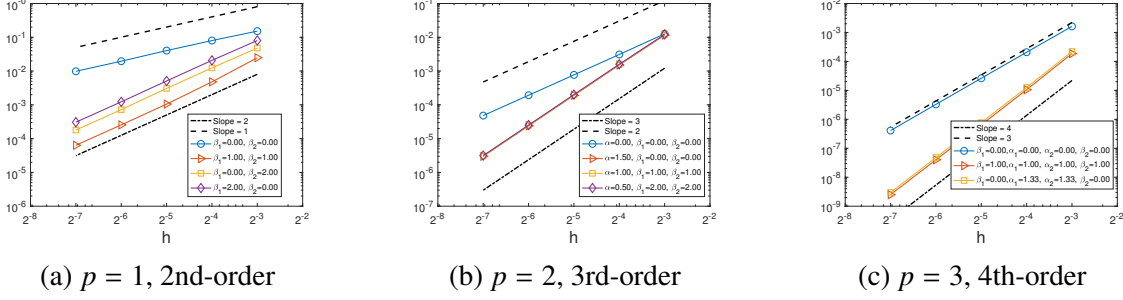


Figure 2.9: The convergence plots of the Active Flux schemes for linear advection equation with different order of accuracy.

2.5.1.1 Linear Advection

Consider a linear advection equation,

$$\begin{cases} \frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0, \\ u(x, 0) = \sin(2\pi x). \end{cases} \quad (2.16)$$

We set $a = 1$, and simulate the equation with step size $\Delta t = 0.8\Delta x/a/p$ until $T = 1$. We generate the error convergence with different p and different discrepancy distribution weights. Setting all weights zero is equivalent to skipping the corrector step. Figure 2.9 shows that enforcing the loose form of discrete conservation Eq. (2.5) in the corrector step gives a global $p + 1$ -th-order accuracy.

2.5.1.2 The Burgers Equation

We consider the Burgers equation as an example of nonlinear equations.

$$\begin{cases} \frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} = 0, \\ u(x, 0) = \sin(2\pi x) + 1. \end{cases} \quad (2.17)$$

In the predictor stage, we initialize $a = \tilde{u}$ by setting $\tilde{u} = U(1 - \frac{\partial u^h}{\partial x} \Delta t)$, where $\frac{\partial u^h}{\partial x}$ take values from the left for nodes at interfaces. Then the prediction is made by tracing the characteristics: $U^* = u^h(x - a\Delta t)$. For $p = 3$, we iterate one more time with $a = U^*$. We set $\Delta t = 0.7\Delta x/2/p$. The

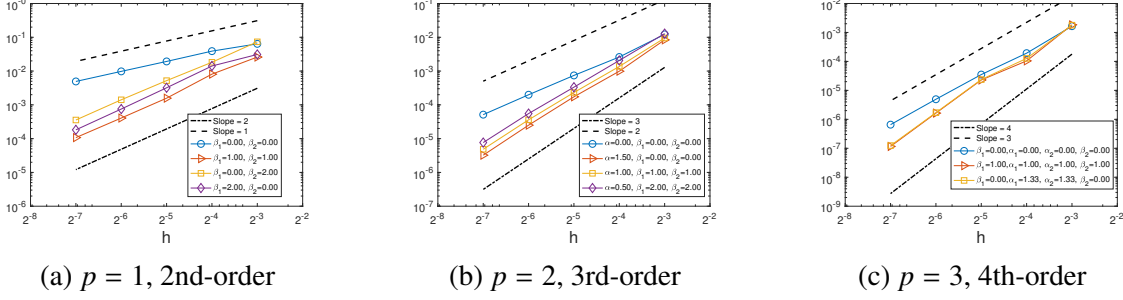


Figure 2.10: The convergence plots of the Active Flux schemes for the Burgers equation with different order accuracy.

error is measured at $T = 0.1$, where no shock wave forms. Figure 2.10 confirms the $p + 1$ -th-order accuracy.

2.5.1.3 The Euler Equations

We further test the numerical schemes with the Euler equations, a nonlinear system of the conservation laws.

$$\begin{cases}
 \frac{\partial \rho}{\partial t} + v \frac{\partial \rho}{\partial x} + \rho \frac{\partial v}{\partial x} = 0 \\
 \rho \frac{\partial v}{\partial t} + \rho v \frac{\partial v}{\partial x} + \frac{\partial p}{\partial x} = 0 \\
 \frac{\partial p}{\partial t} + v \frac{\partial p}{\partial x} + \gamma p \frac{\partial v}{\partial x} = 0
 \end{cases} \quad (2.18)$$

IC: $\rho(x, 0) = 0$, $v(x, 0) = \sin(2\pi x) + 2$, $p(x, 0) = 0.1$.

In the predictor stage, an operator splitting strategy similar to [39] is used. (See Section 3.1). The corrector is the same as the scalar cases. We set all weights of discrepancy distribution equal 1 for all p^4 .

We simulate the system until $T = 0.08$ with step size $\Delta t = 0.8\Delta x/p/3$. Figure 2.11 shows that the errors measured by cell-average L_1 -norm converge with rate $p + 1$ as we refine the mesh. When the total numbers of DOFs are same, third-order scheme has much lower errors, e.g., $\Delta x = h = 1/80$ for $p = 1$ versus $\Delta x = h = 1/40$ for $p = 2$.

⁴Here p represents the degree of polynomials that are used in reconstruction. In the differential equations Eq. (2.18), p means pressure.

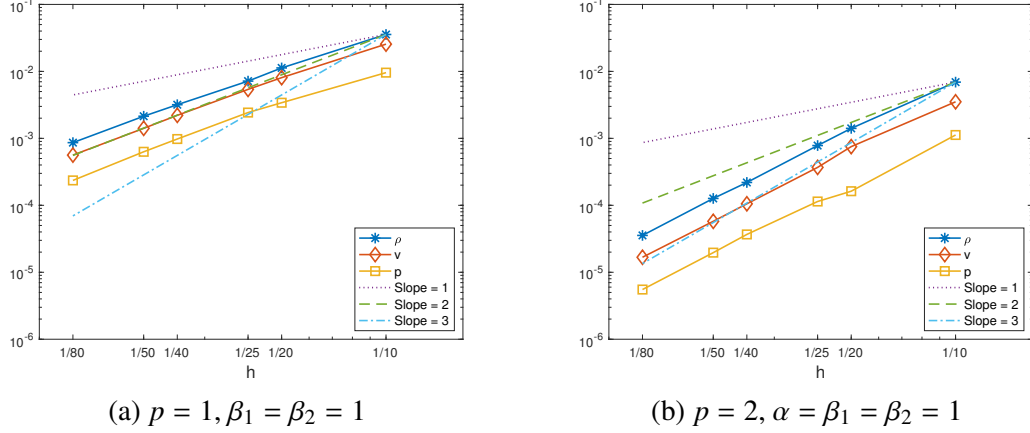


Figure 2.11: The convergence plots of the Active Flux schemes for the Euler equations with different order of accuracy.

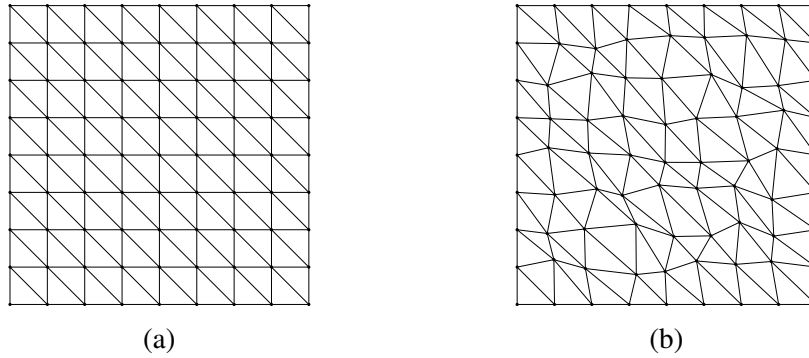


Figure 2.12: The vertices in the regular mesh (a) are perturbed by $0.2h(\sin(10x + 1)^{10y+1}, \cos(10y + 1)^{10x+1})$ in (b).

2.5.2 Two-dimensional Space

To verify the accuracy of the family of numerical schemes for the two-dimensional space, we test them on unperturbed and perturbed meshes (see Figure 2.12). The unperturbed mesh is generated by dividing the domain $[0, 1] \times [0, 1]$ into $N \times N$ squares and each square is split into two triangles. $h = 1/N$ reflects the mesh size. We measure the error by L_1 -norm

$$E = \frac{1}{N_{\text{DOFs}}} \sum_v |U_v - u(\mathbf{x}(v))|. \quad (2.19)$$

2.5.2.1 Linear Advection

Consider the linear advection equation with periodic boundary conditions on $[0, 1] \times [0, 1]$:

$$\begin{cases} \frac{\partial u}{\partial t} + \mathbf{a} \cdot \nabla u = 0, \\ u(x, y, 0) = \sin(2\pi x) \sin(4\pi y), \end{cases} \quad (2.20)$$

with $\mathbf{a} = (\cos \theta, \sin \theta)$, where $\theta = 0, \pi/6, \pi/4, 3\pi/4$.

We set the time step as the 90% of the maximum of allowed step size, i.e., $\Delta t = 0.9 \frac{1}{p} \min_{T,e} H_{T,e}$. In all of the schemes we test, by setting all weights to be 1, the discrepancy $d\bar{U}_T$ is evenly distributed to nodal values. The numerical solutions are compared with the exact ones at $T = 1$.

Figure 2.13 and Figure 2.14 show that with reconstruction of continuous piecewise polynomial of degree p , the numerical schemes generally achieve the expected $p + 1$ -th-order global accuracy. But we notice that, when $\theta = 0$ or $3\pi/4$, which means the direction of advection aligns with the unperturbed mesh, the error convergence rate of the scheme with $p = 3$ drops by 1. This is the so-called mesh alignment issue caused by the low dissipation across the cell boundaries. It also appeared in the original third-order AF method with quadratic reconstruction plus bubble functions [26]. In our new scheme with $p = 3$, the nodal value inside the cell has a weight of 0.5 when we compute the cell average, which is much higher than the contribution from the nodes on vertices or edges. Thus the correction on the cell boundaries has limited effect. Luckily we can alleviate the issues by adjusting the weight of discrepancy distribution, while there is no simple remedy for the original AF scheme.

2.5.2.2 Linear Acoustics

Consider the equations of linear acoustics with periodic boundary conditions on $[0, 1] \times [0, 1]$:

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho_0} \nabla p = 0, \\ \frac{\partial p}{\partial t} + \gamma p_0 \nabla \cdot \mathbf{u} = 0. \end{cases} \quad (2.21)$$

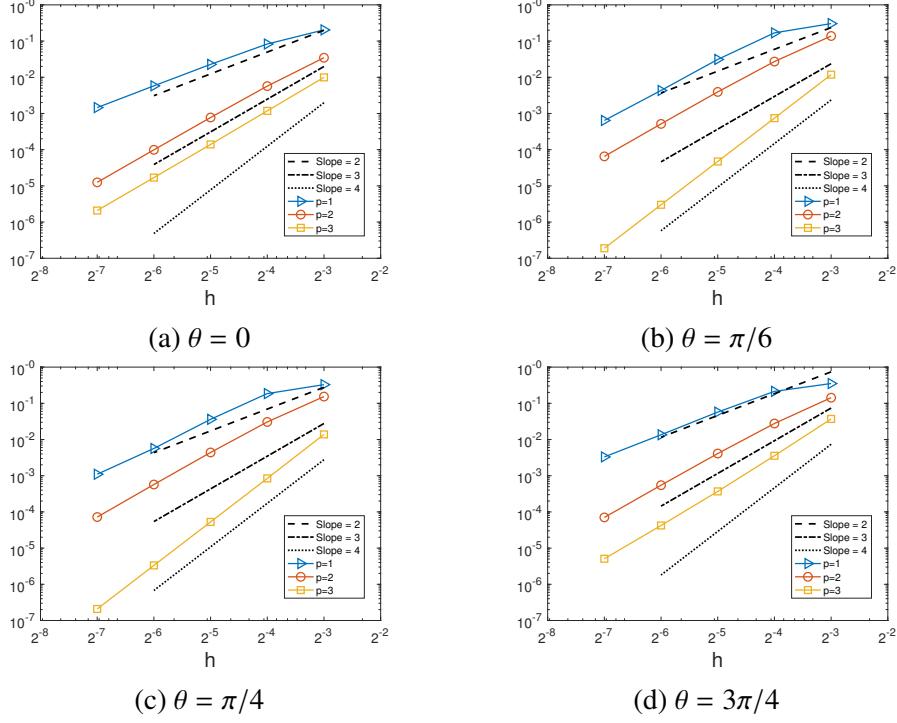


Figure 2.13: Convergence rates of $p + 1$ -th-order schemes for the linear advection equation with different direction $\mathbf{a} = (\cos \theta, \sin \theta)$ on the unperturbed mesh.

The system has analytical solutions [15, 16, 18]. Without loss of generality, we assume $\gamma p_0 = \rho_0 =$

1. Define

$$F_\psi(\mathbf{x}, t) = \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} Q_\psi(\mathbf{x}, \rho) d\rho, \quad (2.22)$$

where,

$$Q_\psi(\mathbf{x}, \rho) = \frac{1}{2\pi} \int_0^{2\pi} \psi(x_1 + \rho \cos \theta, x_2 + \rho \sin \theta) d\theta.$$

The solutions are given by

$$\begin{aligned} p(\mathbf{x}, t_0 + \Delta t) &= \frac{\partial}{\partial t} F_{p(\cdot, t_0)}(\mathbf{x}, \Delta t) - F_{\nabla \cdot \mathbf{u}(\cdot, t_0)}(\mathbf{x}, \Delta t), \\ \mathbf{u}(\mathbf{x}, t_0 + \Delta t) &= Q_{\mathbf{u}(\cdot, t_0)}(\mathbf{x}, 0) - F_{\nabla p(\cdot, t_0)}(\mathbf{x}, \Delta t) \\ &\quad + \int_0^{\Delta t} \left(\frac{\Delta t}{\sqrt{(\Delta t)^2 - \rho^2}} \right) \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\nabla \cdot \mathbf{u}) \Big|_{\mathbf{x} + \rho \mathbf{n}} d\theta d\rho, \end{aligned} \quad (2.23)$$

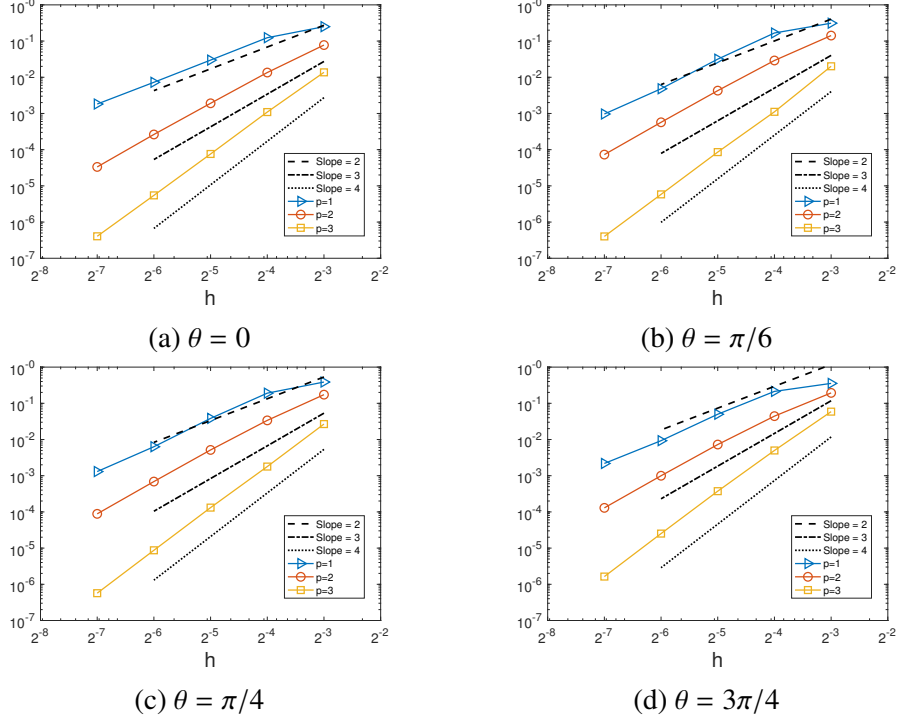


Figure 2.14: Convergence rates of $p + 1$ -th-order schemes for the linear advection equation with different direction $\mathbf{a} = (\cos \theta, \sin \theta)$ on the perturbed mesh.

where $\mathbf{n} = (\cos \theta, \sin \theta)$. For general parameters (ρ_0, p_0) , the exact solutions can be derived with proper transformation of variables. In the following numerical experiments, the predictor in the numerical scheme uses the formula of the exact solutions. All the spatial derivatives involved are no more than first order. Thus we evaluate them directly.

The acoustic process in the fluid does not change the vorticity. We construct solutions that take the following forms

$$\begin{pmatrix} u_1(x, y, t) \\ u_2(x, y, t) \\ p(x, y, t) \end{pmatrix} = \begin{pmatrix} \tilde{u}_1(t) \cos(2\pi k_1 x) \sin(2\pi k_2 y) \\ \tilde{u}_2(t) \sin(2\pi k_1 x) \cos(2\pi k_2 y) \\ \tilde{p}(t) \sin(2\pi k_1 x) \sin(2\pi k_2 y) \end{pmatrix} + \begin{pmatrix} k_2 \sin(2\pi k_1 x) \sin(2\pi k_2 y) \\ k_1 \cos(2\pi k_1 x) \cos(2\pi k_2 y) \\ 0 \end{pmatrix}. \quad (2.24)$$

Note that the velocity in first term in Eq. (2.24) corresponds to the curl-free part of the velocity field while the second represents the divergence-free (or rotational) part which is supposed to remain constant in the acoustic process. It is easy to see that \tilde{u}_1 , \tilde{u}_2 , \tilde{p} is actually governed by a system of

ODEs

$$\frac{\partial}{\partial t} \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{p} \end{pmatrix} = \begin{pmatrix} 0 & 0 & -2\pi k_1 \frac{1}{\rho_0} \\ 0 & 0 & -2\pi k_2 \frac{1}{\rho_0} \\ -2\pi k_1 \gamma p_0 & -2\pi k_2 \gamma p_0 & 0 \end{pmatrix} \begin{pmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{p} \end{pmatrix} \quad (2.25)$$

which can be solved numerically with high accuracy for reference.

The parameters are set as $(\rho_0, p_0) = (10, 5)$, $\gamma = 1.4$, $k_1 = 3$, $k_2 = 2$. We choose initial conditions $\tilde{u}_1 = k_1$, $\tilde{u}_2 = k_2$, $\tilde{p} = 1$ and simulate the acoustic equations until $T = 0.15$ with schemes of different order of accuracy. $\Delta t = 0.9 \frac{1}{ap} \min_{T,e} H_{T,e}$ where a is the speed of sound $a = \sqrt{\frac{\gamma p_0}{\rho_0}}$. Figure 2.15 presents the plots of the errors measured by L_1 -norm. The numerical solutions of pressure do converge with rate $p + 1$ when we use polynomial of degree p for reconstruction. However, the converge rates of errors in u_1 and u_2 is just p . The errors on the perturbed mesh are generally greater than ones on the unperturbed mesh. Further investigation on linear acoustics can be found in Section 3.2.

2.5.2.3 Pressureless Euler Equations

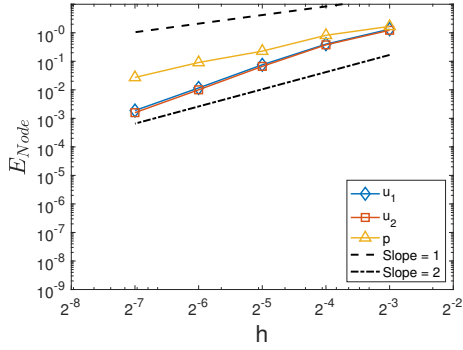
In the two-dimensional space we use the pressureless Euler equations Eq. (2.26) as a test problem for nonlinear systems.

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \\ \frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = 0. \end{cases} \quad (2.26)$$

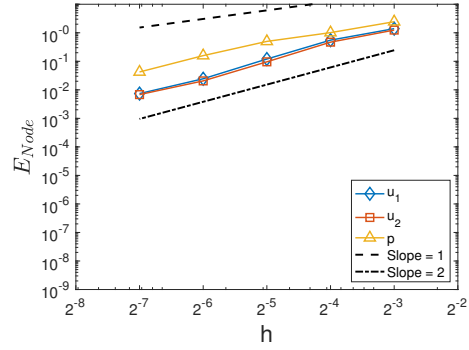
In the early development stage of the Active Flux method, this system had been thoroughly studied by Maeng [26] as the model problem of the nonlinear advection in the Euler equations. But Maeng's numerical scheme was restricted to 3rd-order accuracy, which was achieved by using the piecewise quadratic reconstruction enriched by the bubble functions.

The second equation is actually the Burgers equation in the two-dimensional space. The velocity \mathbf{v} remains constant along the characteristics. The exact solutions are implicitly given by the following equations.⁵

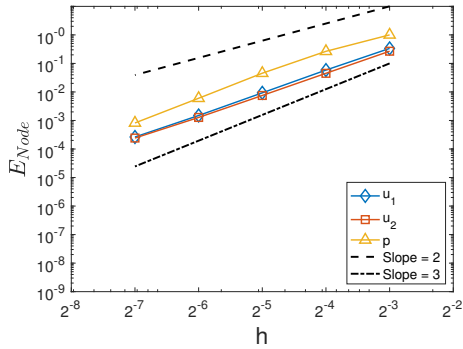
⁵I also provide an alternate concise derivation in the Appendices B.1.



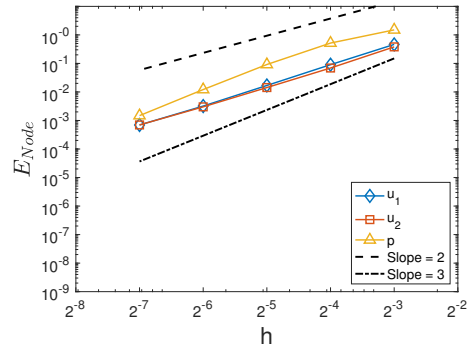
(a) $p = 1$ Unperturbed Mesh



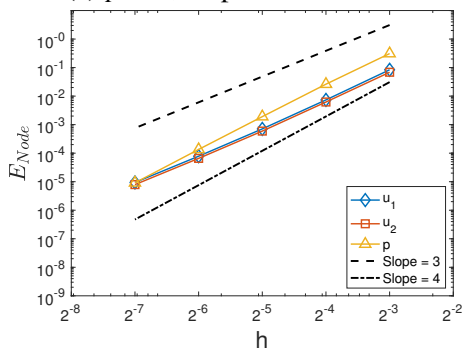
(b) $p = 1$, Perturbed Mesh



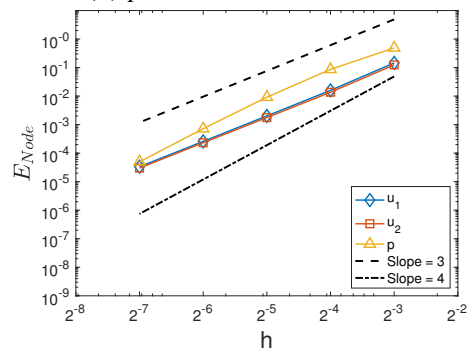
(c) $p = 2$, Unperturbed Mesh



(d) $p = 2$, Perturbed Mesh



(e) $p = 3$, Unperturbed Mesh



(f) $p = 3$, Perturbed Mesh

Figure 2.15: Convergence rates of $p + 1$ -th-order schemes for the equations of linear acoustics on the unperturbed mesh and the perturbed mesh.

$$\begin{aligned}
\mathbf{x}^* &= \mathbf{x} - \Delta t \mathbf{v}(\mathbf{x}, t_0 + \Delta t), \\
\mathbf{v}(\mathbf{x}, t_0 + \Delta t) &= \mathbf{v}(\mathbf{x}^*, t_0), \\
\rho(\mathbf{x}, t_0 + \Delta t) &= \frac{\rho(\mathbf{x}^*, t_0)}{\det(1 + \Delta t \nabla \mathbf{v}) \Big|_{\mathbf{x}=\mathbf{x}^*, t=t_0}}.
\end{aligned} \tag{2.27}$$

In the evolution from $t = t^n$ to $t = t^{n+1}$, the numerical predictor initially estimates \mathbf{x}^* by

$$\mathbf{x}^* = \mathbf{x} - \Delta t (\mathbf{v} - \Delta t \mathbf{v}^h \cdot \nabla \mathbf{v}^h).$$

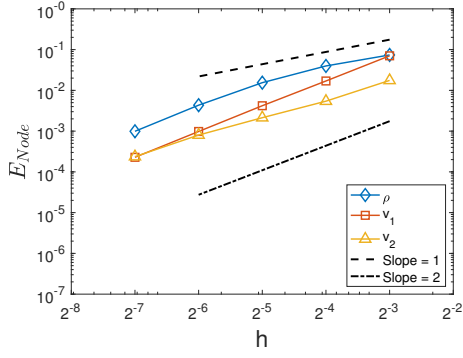
Then \mathbf{v} and ρ are updated by the formulas Eq. (2.27), using the reconstructed solutions from $t = t^n$. If $p > 2$, to achieve local truncation error of $p + 1$ -th order, the tracing-back step needs to be repeated for another $p - 2$ times, with \mathbf{x}^* estimated by using the latest predicted \mathbf{v} at (\mathbf{x}, t^{n+1}) : $\mathbf{x}^* = \mathbf{x} - \Delta t \mathbf{v}$.

Note that the conserved variables are ρ and $\rho \mathbf{v}$ in the pressureless Euler equations.

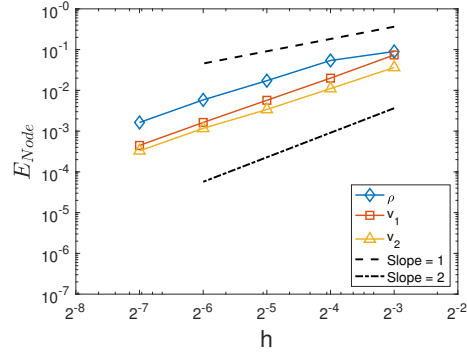
$$\begin{aligned}
\int_T \frac{\partial \rho}{\partial t} d\sigma + \oint_{\partial T} \rho (\mathbf{v} \cdot \mathbf{n}) dl &= 0, \\
\int_T \frac{\partial \rho \mathbf{v}}{\partial t} d\sigma + \oint_{\partial T} \rho \mathbf{v} (\mathbf{v} \cdot \mathbf{n}) dl &= 0.
\end{aligned} \tag{2.28}$$

We evaluate the nodal values of the conserved variables directly from the primitive variables ρ and \mathbf{v} once we complete the predictor step. Piecewise polynomial reconstruction of the conserved variables is assumed (although we do not actually compute it) when we compute integrals in the discrepancy defined by Eq. (2.7). Then the discrepancy is distributed to the nodal values of the conserved variables as described in Eq. (2.8). Transforming the corrected conserved variables ρ and $\rho \mathbf{v}$ back to the primitive variables ρ and \mathbf{v} completes the corrector stage.

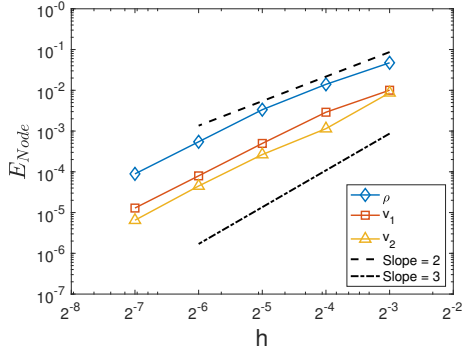
In the numerical experiments, we use periodic boundary conditions on the domain $[0, 1] \times [0, 1]$. Initial values are set as $\rho(\mathbf{x}, 0) = 1$, $v_1 = \sin(2\pi x) + 2$, $v_2 = \cos(2\pi y) + 1$. $\Delta t = 0.95 \frac{1}{v_{max} p} \min_{T,e} H_{T,e}$, where $v_{max} = 4$. We measure the errors of numerical solutions at $T = 0.1$. The convergence plots in Figure 2.16 verify the $p + 1$ -th-order accuracy.



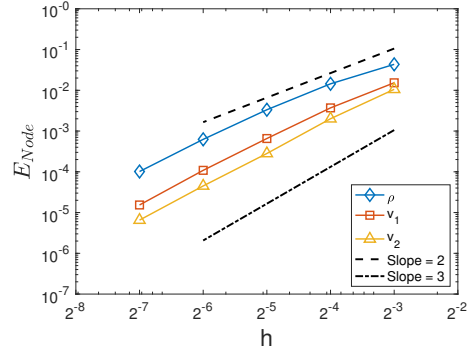
(a) $p = 1$ Unperturbed Mesh



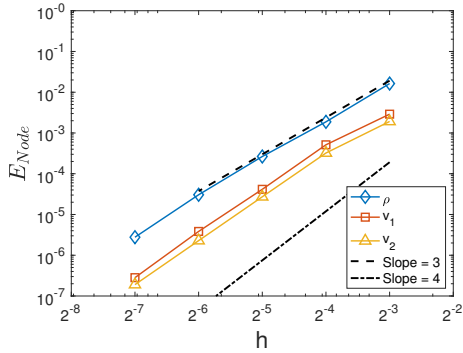
(b) $p = 1$ Perturbed Mesh



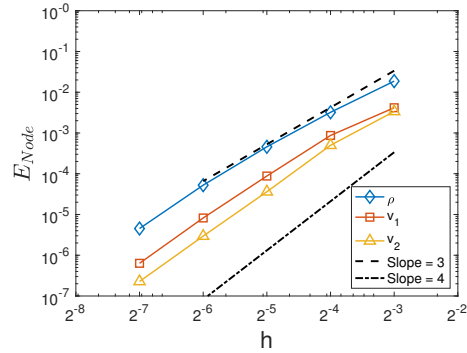
(c) $p = 2$ Unperturbed Mesh



(d) $p = 2$ Perturbed Mesh



(e) $p = 3$ Unperturbed Mesh



(f) $p = 3$ Perturbed Mesh

Figure 2.16: Convergence rates of $p + 1$ -th-order schemes for the pressureless Euler equations on the unperturbed mesh and the perturbed mesh.

CHAPTER 3

Operator Splitting and Treatment of Spatial Derivatives

The test problems used in Chapter 2 all have exact solutions. Thus, the predictor based on the formulas does not bring in errors except for the one caused by spatial discretization. Besides, each test case examines only one type of physical process and the optimal numerical domain of dependence is used in the predictor. For example, the problems of advective nature mainly use the information from the cell where the characteristics originate, while the acoustic problem uses a symmetric domain of dependence by taking the integral over a disk centered at the current node.

However, in real applications, the complicated governing equations of flows do not have analytical formulas, and physical processes of different natures are mixed up. For instance, the Euler equations contain both advection and acoustics. To fully use the nature of the physics, an operator splitting method is adopted for the predictor. Several different operator splitting strategies for the Euler equations had been studied by Maeng [26] and Fan [16]. In the following sections, we are going to revisit the Euler equations and formulate the idea in a more mathematical way.

3.1 The First Nontrivial Example: the Euler Equations

The accuracy of the AF scheme mainly depends on the accuracy of the predictor. (The corrector is more about preventing the local truncation errors from accumulating.) We consider a third-order

Lax-Wendroff-type predictor as an example.

$$\mathbf{u}(x_0, t_0 + \Delta t) = \mathbf{u} + \Delta t \frac{\partial \mathbf{u}}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 \mathbf{u}}{\partial t^2} + O(\Delta t^3) \quad (3.1)$$

By repeatedly using the first-order system of PDEs, all the temporal derivatives can be written as spatial derivatives at (x_0, t_0) .

The terms on the right-hand side of the Euler equations are categorized either as advective terms (marked by underbrace $\underbrace{\quad}$) or acoustic terms (marked by underline $\underline{\quad}$).

$$\begin{aligned} \rho_t &= -\underbrace{\mathbf{u} \cdot \nabla \rho}_{\text{advective}} - \underline{\rho \nabla \cdot \mathbf{u}}_{\text{acoustic}} \\ \mathbf{u}_t &= -\underbrace{\mathbf{u} \cdot \nabla \mathbf{u}}_{\text{advective}} - \underline{\frac{1}{\rho} \nabla p}_{\text{acoustic}} \\ p_t &= -\underbrace{\mathbf{u} \cdot \nabla p}_{\text{advective}} - \underline{\gamma p \nabla \cdot \mathbf{u}}_{\text{acoustic}} \end{aligned} \quad (3.2)$$

When we pursue any solvers that have order of accuracy higher than second order, the nonlinear interaction between the advective terms and acoustic terms as well as other nonlinear effect are no longer negligible. The second-order derivatives of the primitive variables ρ , \mathbf{u} , p are complicated.

$$\begin{aligned} \rho_{tt} &= -\mathbf{u}_t \cdot \nabla \rho - \rho_t \nabla \cdot \mathbf{u} - \underbrace{\mathbf{u} \cdot \nabla \rho}_{\text{advective}} - \underline{\rho \nabla \cdot \mathbf{u}_t}_{\text{acoustic}} \\ &= \left(\underbrace{\mathbf{u} \cdot \nabla \rho}_{\text{advective}} + \underline{\rho \nabla \cdot \mathbf{u}}_{\text{acoustic}} \right) \cdot \nabla \rho + \left(\underbrace{\mathbf{u} \cdot \nabla \rho}_{\text{advective}} + \underline{\rho \nabla \cdot \mathbf{u}}_{\text{acoustic}} \right) \nabla \cdot \mathbf{u} \\ &\quad + \underbrace{\mathbf{u} \cdot \nabla}_{\text{advective}} \left(\underbrace{\mathbf{u} \cdot \nabla \rho}_{\text{advective}} + \underline{\rho \nabla \cdot \mathbf{u}}_{\text{acoustic}} \right) + \underline{\rho \nabla \cdot \left(\underbrace{\mathbf{u} \cdot \nabla \mathbf{u}}_{\text{advective}} + \underline{\frac{1}{\rho} \nabla p}_{\text{acoustic}} \right)}_{\text{acoustic}} \end{aligned} \quad (3.3)$$

$$\begin{aligned} \mathbf{u}_{tt} &= -\mathbf{u}_t \cdot \nabla \mathbf{u} - \frac{-\rho_t \nabla p}{\rho^2} - \underbrace{\mathbf{u} \cdot \nabla \mathbf{u}_t}_{\text{advective}} - \underline{\frac{1}{\rho} \nabla p_t}_{\text{acoustic}} \\ &= \left(\underbrace{\mathbf{u} \cdot \nabla \mathbf{u}}_{\text{advective}} + \underline{\frac{1}{\rho} \nabla p}_{\text{acoustic}} \right) \cdot \nabla \mathbf{u} + \frac{-1}{\rho^2} \left(\underbrace{\mathbf{u} \cdot \nabla \rho}_{\text{advective}} + \underline{\rho \nabla \cdot \mathbf{u}}_{\text{acoustic}} \right) \nabla p \\ &\quad + \underbrace{\mathbf{u} \cdot \nabla}_{\text{advective}} \left(\underbrace{\mathbf{u} \cdot \nabla \mathbf{u}}_{\text{advective}} + \underline{\frac{1}{\rho} \nabla p}_{\text{acoustic}} \right) + \underline{\frac{1}{\rho} \nabla \left(\underbrace{\mathbf{u} \cdot \nabla p}_{\text{advective}} + \underline{\gamma p \nabla \cdot \mathbf{u}}_{\text{acoustic}} \right)}_{\text{acoustic}} \end{aligned} \quad (3.4)$$

$$\begin{aligned}
p_t &= -\mathbf{u}_t \cdot \underbrace{\nabla p} - \underbrace{\gamma p_t \nabla \cdot \mathbf{u}} - \underbrace{\mathbf{u} \cdot \nabla p_t} - \underbrace{\gamma p \nabla \cdot \mathbf{u}_t} \\
&= \left(\underbrace{\mathbf{u} \cdot \nabla \mathbf{u}} + \frac{1}{\rho} \nabla p \right) \cdot \underbrace{\nabla p} + \gamma \left(\underbrace{\mathbf{u} \cdot \nabla p} + \underbrace{\gamma p \nabla \cdot \mathbf{u}} \right) \underbrace{\nabla \cdot \mathbf{u}} \\
&\quad + \underbrace{\mathbf{u} \cdot \nabla} \left(\underbrace{\mathbf{u} \cdot \nabla p} + \underbrace{\gamma p \nabla \cdot \mathbf{u}} \right) + \underbrace{\gamma p \nabla \cdot} \left(\underbrace{\mathbf{u} \cdot \nabla \mathbf{u}} + \frac{1}{\rho} \nabla p \right)
\end{aligned} \tag{3.5}$$

After we expand the expressions with the distributive law, the all-underbraced terms correspond to advection. We define the update of a variable $\mathbf{\blacksquare}$ from advection as follows.

$$\mathbf{\blacksquare}^{\text{ad}} = \mathbf{\blacksquare} - \Delta t \mathbf{u} \cdot \nabla \mathbf{\blacksquare} + \frac{\Delta t^2}{2} \left((\mathbf{u} \cdot \nabla \mathbf{u}) \cdot \nabla \mathbf{\blacksquare} + \mathbf{u} \cdot \nabla (\mathbf{u} \cdot \nabla \mathbf{\blacksquare}) \right) + O(\Delta t^3). \tag{3.6}$$

We do not actually evaluate those terms in the numerical solver. Instead, we use the method of characteristics where all advective terms upto second order are included. Set

$$\mathbf{x}^* = \mathbf{x} - \Delta t (\mathbf{u} - \Delta t \mathbf{u} \cdot \nabla \mathbf{u}).$$

$$\mathbf{\blacksquare}^{\text{ad}} = \mathbf{\blacksquare}^h(\mathbf{x}^*), \tag{3.7}$$

where $\mathbf{\blacksquare}^h$ represents the reconstructed solution.

The products of underbraced terms and underlined terms represent the nonlinear interaction of the advection and the acoustics. The all-underlined terms are for the pure acoustic process. Fortunately, most interaction terms and nonlinear acoustic terms in the expressions of the Lax-Wendroff predictor of ρ can be neatly absorbed [40],

$$\rho^* = \frac{\rho^{\text{ad}}}{\det(1 + \Delta t \nabla \mathbf{u}) \Big|_{\mathbf{x}=\mathbf{x}^*}} + \frac{\Delta t^2}{2} \Delta p. \tag{3.8}$$

p^* has a similar simplification,

$$p^* = \frac{p^{\text{ad}}}{(\det(1 + \Delta t \nabla \mathbf{u}))^\gamma} + \frac{\Delta t^2}{2} \left(\frac{\gamma p}{\rho} \Delta p + \frac{1}{\rho} \nabla p \cdot \nabla p - \frac{\gamma p}{\rho^2} \nabla \rho \cdot \nabla p \right). \tag{3.9}$$

But there is no convincing way to absorb nonlinear acoustics and the interaction terms for the velocity.

$$\mathbf{u}^* = \mathbf{u}^{\text{ad}} - \frac{\Delta t}{\rho} \nabla p + \frac{\Delta t^2}{2} \left(\frac{1}{\rho} \nabla p \cdot \nabla \mathbf{u} + \frac{-1}{\rho^2} (\mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u}) \nabla p + \mathbf{u} \cdot \nabla \left(\frac{1}{\rho} \nabla p \right) + \frac{1}{\rho} \nabla (\mathbf{u} \cdot \nabla p + \gamma p \nabla \cdot \mathbf{u}) \right). \quad (3.10)$$

For terms like $\mathbf{u} \cdot \nabla \left(\frac{1}{\rho} \nabla p \right)$, we further expand them by using the product rule.

Mathematically, all the predictors of the same order of accuracy are equal. In practice, some predictors are more equal than others. This is because different leading terms of errors have different influence on stability. The belief is that proper use of the domain of dependence will make the numerical schemes more efficient.

Recall that the acoustic process has a symmetric domain of dependence. Thus all the derivatives in the acoustic terms and the interaction terms should be evaluated in a way that reflects this fact (see the next section). This completes the operator splitting. With all the first- and second- order spatial derivatives of the primitive variables evaluated, the predictor simply needs to add up the terms prescribed by the Lax-Wendroff method Eq. (3.1).¹

3.2 Treatment of Spatial Derivatives: Ideas Inspired by the Solution to the Wave Equation

In the AF-based numerical schemes introduced in Chapter 2, piecewise polynomials of C^0 continuity are reconstructed from nodal values at $t = t^n$ for the temporal evolution to $t = t^{n+1}$. Within a cell, the reconstructed function u^h is differentiable. The accuracy of the first-order derivatives evaluated by direct differentiation is one order lower than that of u^h . Note that ∇u^h is discontinuous at the cell boundaries. For the first-order derivatives, simply taking some average value is a satisfying practical treatment. However, the discontinuity of ∇u^h leaves the second-order derivatives

¹The predictor described here is different from Maeng's [26] or Fan's work [16]. For example, before incorporating the acoustic terms, their methods require another reconstruction step after getting the update from the advection. But we do not need it in our predictor.

undefined.

The analytical solutions Eq. (2.23) to the linear acoustic system Eq. (2.21) give us some hints to treat the spatial derivatives if we interpret the formulas as a Lax-Wendroff method.

Recall the linear acoustic system,

$$\begin{cases} \frac{\partial \mathbf{u}}{\partial t} + \frac{1}{\rho_0} \nabla p = 0, \\ \frac{\partial p}{\partial t} + \gamma p_0 \nabla \cdot \mathbf{u} = 0. \end{cases}$$

For simplicity we assume $\gamma p_0 = \rho_0 = 1$. Note the following mathematical facts.

Fact 1:

$$\frac{\partial^2 p}{\partial t^2} = \Delta p, \quad \frac{\partial^2 \mathbf{u}}{\partial t^2} = \nabla(\nabla \cdot \mathbf{u}). \quad (3.11)$$

Fact 2:

Define $\mathbf{n} = (\cos \theta, \sin \theta)$.

$$\begin{aligned} \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\nabla \cdot \mathbf{u}) \Big|_{(x_1, x_2) + \rho(\cos \theta, \sin \theta)} d\theta &= \frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{u})(x_1 + r_1, x_2 + r_2) dr_1 dr_2 \\ &= \frac{\rho}{2\pi\rho^2} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{u})(x_1 + r_1, x_2 + r_2) dr_1 dr_2. \end{aligned} \quad (3.12)$$

Fact 3:

By the definition of $Q_\psi(\mathbf{x}, \rho)$,

$$Q_\psi(\mathbf{x}, \rho) = \frac{1}{2\pi} \int_0^{2\pi} \psi(x_1 + \rho \cos \theta, x_2 + \rho \sin \theta) d\theta.$$

we have

$$Q_\psi(\mathbf{x}, 0) = \psi(\mathbf{x}). \quad (3.13)$$

$$\frac{\partial Q_\psi(\mathbf{x}, \rho)}{\partial \rho} = \frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \Delta \psi(x_1 + r_1, x_2 + r_2) dr_1 dr_2. \quad (3.14)$$

Fact 4:

Apply integration by part on the definition of $F_\psi(\mathbf{x}, t)$:

$$F_\psi(\mathbf{x}, t) = \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} Q_\psi(\mathbf{x}, \rho) d\rho = tQ_\psi(\mathbf{x}, 0) + \int_0^t \sqrt{t^2 - \rho^2} \frac{\partial Q_\psi(\mathbf{x}, \rho)}{\partial \rho} d\rho.$$

$$\frac{\partial}{\partial t} F_\psi(\mathbf{x}, t) = Q_\psi(\mathbf{x}, 0) + \int_0^t \frac{t}{\sqrt{t^2 - \rho^2}} \frac{\partial Q_\psi(\mathbf{x}, \rho)}{\partial \rho} d\rho. \quad (3.15)$$

We define the following two types of average for a function over a disk of radius t centered at

\mathbf{x} .

$$H_\psi(\mathbf{x}, t) = \frac{1}{t} F_\psi(\mathbf{x}, t) = \frac{1}{t} \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} Q_\psi(\mathbf{x}, \rho) d\rho,$$

$$G_\psi(\mathbf{x}, t) = \frac{1}{t} \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} \left(\frac{1}{\pi \rho^2} \iint_{r^2 \leq \rho^2} \psi(x_1 + r_1, x_2 + r_2) dr_1 dr_2 \right) d\rho. \quad (3.16)$$

Then the analytical solutions to the acoustic system

$$p(\mathbf{x}, t_0 + \Delta t) = \frac{\partial}{\partial t} F_{p(\cdot, t_0)}(\mathbf{x}, \Delta t) - F_{\nabla \cdot \mathbf{u}(\cdot, t_0)}(\mathbf{x}, \Delta t),$$

$$\mathbf{u}(\mathbf{x}, t_0 + \Delta t) = Q_{\mathbf{u}(\cdot, t_0)}(\mathbf{x}, 0) - F_{\nabla p(\cdot, t_0)}(\mathbf{x}, \Delta t) + \int_0^{\Delta t} \left(\frac{\Delta t}{\sqrt{(\Delta t)^2 - \rho^2}} \right) \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\nabla \cdot \mathbf{u}) d\theta d\rho.$$

can be written into a unified form:

$$p(\mathbf{x}, t_0 + \Delta t) = p(\mathbf{x}, t_0) + \Delta t H_{p_t(\cdot, t_0)}(\mathbf{x}, \Delta t) + \frac{\Delta t^2}{2} G_{p_{tt}(\cdot, t_0)}(\mathbf{x}, \Delta t),$$

$$\mathbf{u}(\mathbf{x}, t_0 + \Delta t) = \mathbf{u}(\mathbf{x}, t_0) + \Delta t H_{\mathbf{u}_t(\cdot, t_0)}(\mathbf{x}, \Delta t) + \frac{\Delta t^2}{2} G_{\mathbf{u}_{tt}(\cdot, t_0)}(\mathbf{x}, \Delta t). \quad (3.17)$$

The performance of the AF-based numerical schemes in simulating linear acoustics (see 2.5.2.2) meets our requirement of accuracy for practical purposes. Thus for the derivatives that have acoustic nature or reflect the interaction of advection and acoustics in the operator splitting Eq. (3.8), Eq. (3.9), Eq. (3.10), we estimate the first-order spatial derivative at each node by H , second-order derivatives by G . Since the sound wave speed is no longer a constant in the Euler equations, the radius of the disk from which we take the averages should equal the local sound wave speed $\sqrt{\gamma p/\rho}$ at (\mathbf{x}, t_0) times Δt .

Let w be a continuous function of C^2 approximated by piecewise continuous function w^h . The

sound wave speed at (\mathbf{x}, t) is c . w_x^h and w_y^h are evaluated directly, thus they are piecewise-defined but discontinuous at the cell boundaries. Then we have the following approximation of the spatial derivatives at (\mathbf{x}, t) .

$$\frac{\partial w}{\partial x} = H_{w_x^h}(\mathbf{x}, c\Delta t), \quad \frac{\partial w}{\partial y} = H_{w_y^h}(\mathbf{x}, c\Delta t). \quad (3.18)$$

As for the second-order derivatives, when evaluating the integrals in the definition of G , we need the following tricks.

$$\begin{aligned} \iint_{r^2 \leq \rho^2} \frac{\partial^2 w}{\partial x^2} dr_1 dr_2 &= \iint_{r^2 \leq \rho^2} \nabla \cdot \left(\frac{\partial w}{\partial x}, 0 \right) dr_1 dr_2 = \oint_{r=\rho} \mathbf{n} \cdot \left(\frac{\partial w^h}{\partial x}, 0 \right) dl, \\ \iint_{r^2 \leq \rho^2} \frac{\partial^2 w}{\partial x \partial y} dr_1 dr_2 &= \iint_{r^2 \leq \rho^2} \frac{1}{2} \nabla \cdot \left(\frac{\partial w}{\partial y}, \frac{\partial w}{\partial x} \right) dr_1 dr_2 = \oint_{r=\rho} \frac{1}{2} \mathbf{n} \cdot \left(\frac{\partial w^h}{\partial y}, \frac{\partial w^h}{\partial x} \right) dl, \\ \iint_{r^2 \leq \rho^2} \frac{\partial^2 w}{\partial y^2} dr_1 dr_2 &= \iint_{r^2 \leq \rho^2} \nabla \cdot \left(0, \frac{\partial w}{\partial y} \right) dr_1 dr_2 = \oint_{r=\rho} \mathbf{n} \cdot \left(0, \frac{\partial w^h}{\partial y} \right) dl. \end{aligned} \quad (3.19)$$

$$\frac{\partial^2 w}{\partial x^2} = G_{w_{xx}}(\mathbf{x}, c\Delta t), \quad \frac{\partial^2 w}{\partial x \partial y} = G_{w_{xy}}(\mathbf{x}, c\Delta t), \quad \frac{\partial^2 w}{\partial y^2} = G_{w_{yy}}(\mathbf{x}, c\Delta t). \quad (3.20)$$

Consider a function $u(x, y) = \sin(6\pi x) \sin(4\pi y)$ on $[0, 1] \times [0, 1]$. In Figure 3.1, we measure the accuracy of derivative approximation based on such strategies on the meshes described by Figure 2.12, for various mesh sizes h and the degree p of polynomials that are used in the reconstruction. The radius of the disk is given by $R = 0.95 \frac{1}{p} \min_{T,e} H_{T,e}$. The treatment for derivatives on the boundary is described in Section 4.1.1.1. We compare the numerical values of u_x , u_y , u_{xx} , u_{xy} , u_{yy} and the exact values. The errors are measured by L_1 -norm defined by Eq. (2.19). Interestingly, we notice that the approximation of first-order derivatives is always around 2nd order for all p because we take the average over a disk and thus lose the high resolution brought by the high-order polynomial; the second-order derivatives have $p - 1$ -th-order accuracy.

Notes:

1. The average of ψ defined by $H_\psi(\mathbf{x}, t)$ is the so-called spherical mean that was introduced to solve the three-dimensional wave equations. (The functions in two-dimensional space can be regarded as functions independent of z in three-dimensional space.) This derivative approximation strategy is extendable to three-dimensional space.

2. Although the predictor based on the Lax-Wendroff method seems of third-order local-truncation error (Eq. (3.1)), when being applied to the linear acoustic equations with derivatives approximated by the strategies described above, it is identical to the exact solution. Thus we can expect that the actual accuracy is better. (See the numerical experiments on the Euler equations 3.2.)
3. A similar treatment was used by Fan [16]. But in that work, the description was vague. It seems that only the average H (the spherical mean) was used for derivatives in the interaction terms and nonlinear acoustic terms.

3.3 Numerical Results

$$\begin{cases} \rho_t + \mathbf{u} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{u} = 0, \\ \mathbf{u}_t + \mathbf{u} \cdot \nabla \mathbf{u} + \frac{1}{\rho} \nabla p = 0, \\ p_t + \mathbf{u} \cdot \nabla p + \gamma p \nabla \cdot \mathbf{u} = 0. \end{cases} \quad (3.21)$$

To verify the accuracy, we consider the vortex transport problem governed by the Euler equations. The analytical solutions are given by the following equations.

$$\begin{aligned} \rho(x, y, t) &= \rho_\infty f_1^{1/(\gamma-1)}, \\ u_1(x, y, t) &= U_\infty - f_2(y - V_\infty t), \\ u_2(x, y, t) &= V_\infty + f_2(x - U_\infty t), \\ p(x, y, t) &= p_\infty f_1^{\gamma/(\gamma-1)}. \end{aligned} \quad (3.22)$$

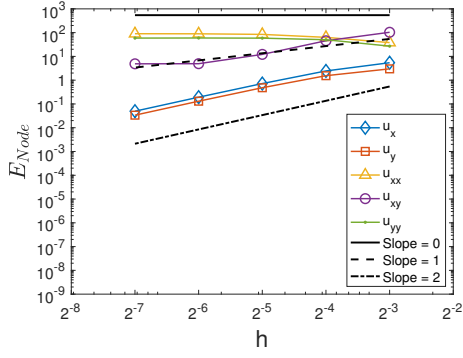
where,

$$f_0 = 1 - \frac{(x - U_\infty t)^2 + (y - V_\infty t)^2}{r_c^2}, \quad f_1 = 1 - \epsilon^2 (\gamma - 1) M_\infty^2 \frac{e^{f_0}}{8\pi^2}, \quad f_2 = \epsilon \frac{\sqrt{U_\infty^2 + V_\infty^2}}{2\pi r_c} e^{f_0/2}.$$

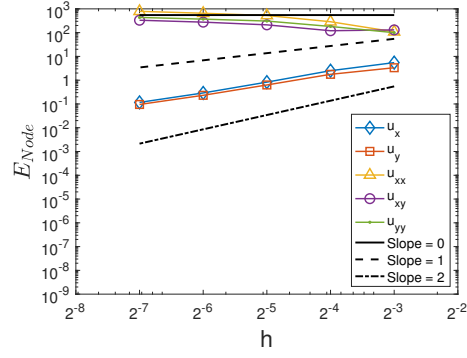
We use following values: $\gamma = 1.4$, $\epsilon = 0.3$, $r_c = 1$; $(U_\infty, V_\infty) = (1, 1)/\sqrt{2}$, Mach number $M_\infty = 0.5$,

$\rho_\infty = 1, p_\infty = 1/(\gamma M_\infty^2)$. If we assign periodic boundary conditions to the domain $[-5, 5] \times [-5, 5]$, the vortex will return to the origin at $T = 10\sqrt{2}$.

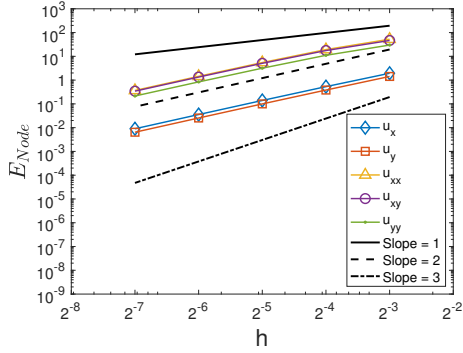
The predictor is based on the operator splitting discussed in Section 3.1. $\rho, \rho \mathbf{u}, \frac{1}{2}\rho \mathbf{u} \cdot \mathbf{u} + \frac{p}{\gamma-1}$ are the conserved variables. The step of reconciliation with conservation law is similar to the one for the pressureless Euler equations. Figure 3.2 confirms that the convergence rates of the L_1 errors defined by Eq. (2.19) on the nodal values are $p + 1$ when polynomials of degree p are used.



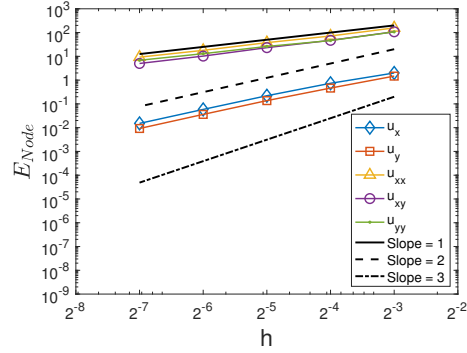
(a) $p = 1$ Unperturbed Mesh



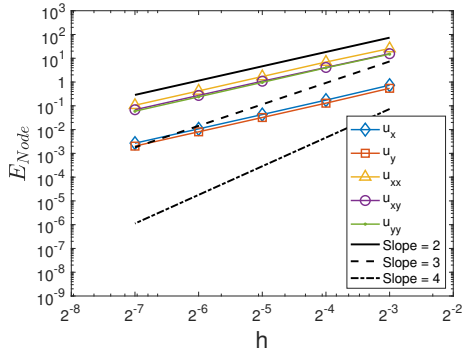
(b) $p = 1$ Perturbed Mesh



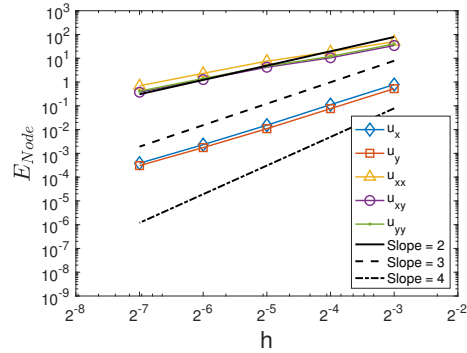
(c) $p = 2$ Unperturbed Mesh



(d) $p = 2$ Perturbed Mesh



(e) $p = 3$ Unperturbed Mesh



(f) $p = 3$ Perturbed Mesh

Figure 3.1: Accuracy of acoustic-formula-based derivative approximation strategies used in $p + 1$ -th-order schemes on the unperturbed mesh and the perturbed mesh.

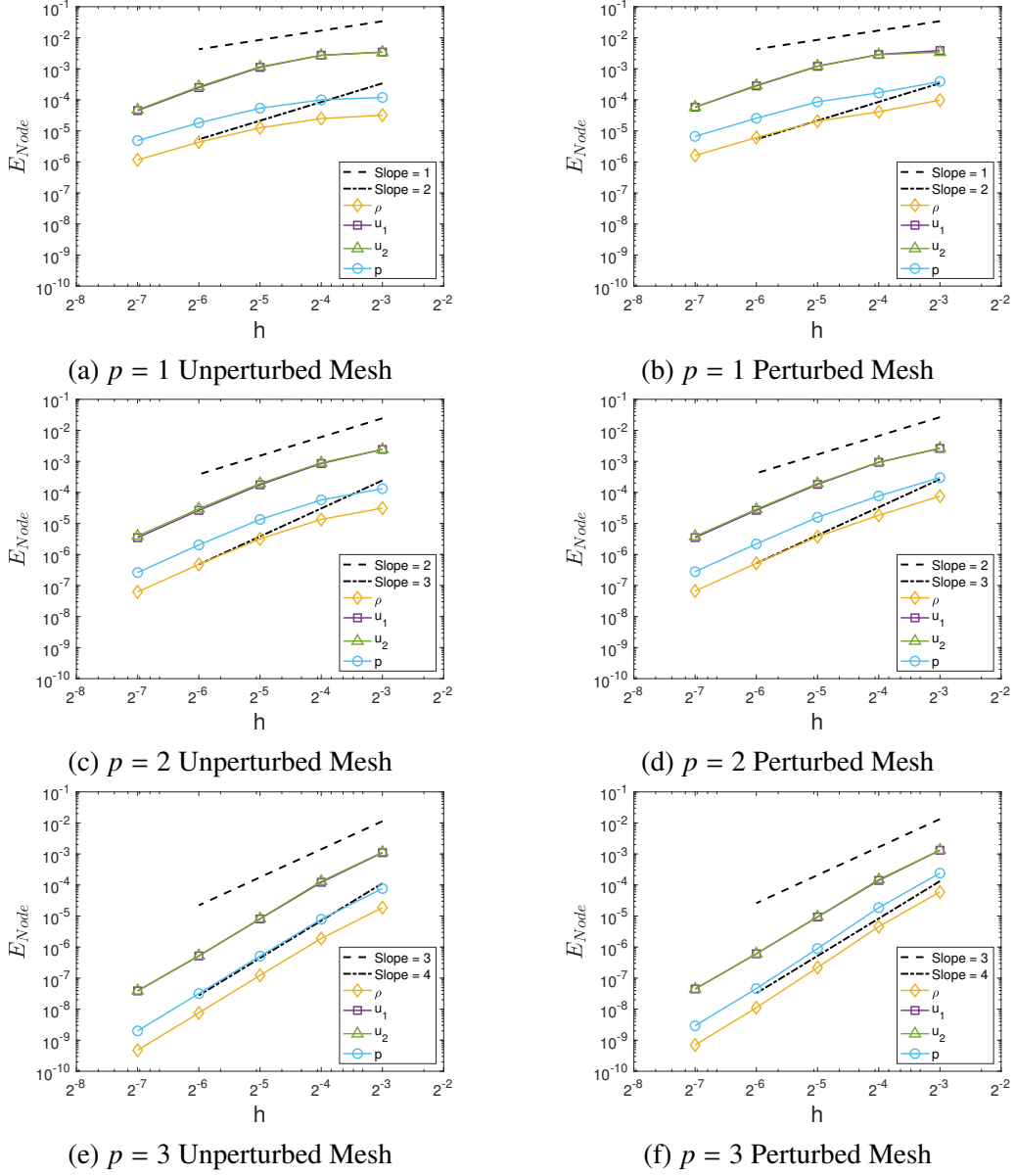


Figure 3.2: Convergence rates of $p + 1$ -th-order schemes for the Euler equations on the unperturbed mesh and the perturbed mesh.

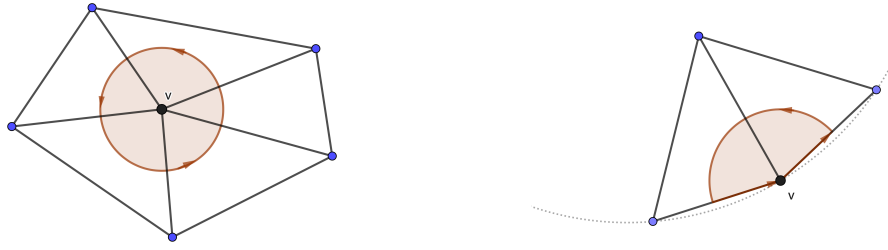
CHAPTER 4

Boundary Conditions, Domain of Complicated Geometry and Stability in the General Active Flux Method

4.1 Boundary Conditions

In the practical use of CFD algorithms, the geometry of the domain of the governing equations is often complicated and nontrivial boundary conditions, e.g., inflow boundary conditions or inviscid boundary conditions, need to be properly imposed. In the early development of the Active Flux method, boundary conditions were enforced via solving a set of equations that involve the gradients of primitive variables [16]. Although the smart use of the curvature of boundaries exempted researchers then from implementing curved elements to achieve third-order accuracy, the extension to general cases e.g., numerical schemes of different accuracy or hyperbolic Navier-Stokes equations, were not clear. To fulfill the complete potential of the Active Flux method, we suggest a simpler but systematical and versatile treatment of the boundary conditions and complex geometry, which does not involve derivatives of variables. From Chapter 4 to Chapter 6, we repeatedly use this treatment in the simulation of the Euler equations and the Navier-Stokes equations where we focus on the second-order scheme from the AF family. The idea is extendable for higher-order schemes.

In this work, we always interpret the Active Flux method as a predictor-corrector method.



(a) An Interior node

(b) A node on the boundary

Figure 4.1: Numerical domain of dependence for evaluating derivatives at Node v

If we compare the Active Flux method with the Finite Volume method or the Discontinuous Galerkin method, we notice that the Lax-Wendroff-type predictor based on the operator splitting (Section 3.1) takes the role of a Riemann solver while the corrector based on the discrepancy distribution (Eq. (2.3) and Eq. (2.8)) has a function similar to the update of exact cell-averages in FV or DG. Note that in the FV or DG methods, the boundary conditions are imposed in the computation of the fluxes through the boundary. Similarly, we let the predictor take the main responsibility of imposing proper boundary conditions. This also turns out to be easy to implement.

4.1.1 Predictor

4.1.1.1 Treatment of Complicated Geometry

Recall that we use certain types of average Eq. (3.16) over a disk centered at a node v to approximate spatial derivatives in Section 3.2.

$$H_\psi(\mathbf{x}, t) = \frac{1}{t} F_\psi(\mathbf{x}, t) = \frac{1}{t} \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} Q_\psi(\mathbf{x}, \rho) d\rho,$$

$$G_\psi(\mathbf{x}, t) = \frac{1}{t} \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} \left(\frac{1}{\pi \rho^2} \iint_{r^2 \leq \rho^2} \psi(x_1 + r_1, x_2 + r_2) dr_1 dr_2 \right) d\rho.$$

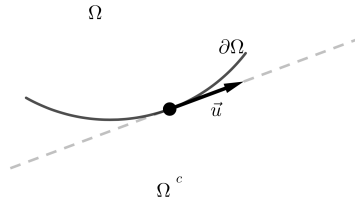


Figure 4.2

The first-order derivatives are approximated by the H -average, which can be regarded as an integral over the disk of a weight function times the first-order derivatives that are obtained by directly differentiating the reconstructed solution u^h . The second-order derivatives are approximated by the G -average, which should be interpreted as the weighted average of some average value of the second-order derivatives. For example,

$$\bar{u}_{xy}^h = \frac{1}{\pi\rho^2} \iint_{r^2 \leq \rho^2} u_{xy}^h dr_1 dr_2,$$

where we use Stokes' theorem to circumvent directly taking the second-order derivatives of u^h . That is, we evaluate the integral of the normal vector times some first-order derivatives of u^h along the boundary of the disk. See Eq. (3.19).

When the node v is inside the domain Ω (Figure 4.1 (a)), both H -average and G -average are well-defined. However, when it is on the boundary, only part of the disk overlaps the domain (Figure 4.1 (b)), thus some treatment is required. For the first-order derivatives, we simply evaluate the integrals over the circular sector enclosed by the two cell boundaries and the circular arc inside the domain Ω . For the second-order derivatives, we need to evaluate the integral of some first-order derivatives of u^h along the boundaries of the circular sector: a circular arc and two radii.

On a boundary where the velocity is almost tangent to the boundary (Figure 4.2), we may fail to trace back the estimated characteristics to an origin \mathbf{x}^* inside the domain. If \mathbf{x}^* is outside the domain, we need to manually add up all terms that appear in the second-order Taylor polynomials in terms of the temporal variable t (Eq. (3.1)).

4.1.1.2 Imposing the Boundary Conditions

Once we address the treatment of a domain with complicated geometry, imposing the boundary condition becomes straightforward.

- Inflow Boundary: Assign the inflow boundary conditions to the nodal values directly.
- Outflow Boundary: Set the pressure as p_∞ . Predict ρ, \mathbf{u} as usual.
- Inviscid Wall Boundary: Use the predictor as usual. Then project the velocity \mathbf{u}^* to the tangent direction. Let \mathbf{n} represent the normal vector pointing outwards.

$$\mathbf{u}^{**} = \mathbf{u}^* - \mathbf{u}^* \cdot \mathbf{n}.$$

We use \mathbf{u}^{**} when computing fluxes.

4.1.2 Corrector

Due to the fact that all the nodal values and fluxes on the boundaries are approximations instead of absolute truth, we can keep the corrector simple by loosening the conservation law in the cells on the boundaries. For interior nodes, the procedure remains the same as the case with periodic boundary conditions. We do not correct the nodal values on the inflow boundary and the pressure on the subsonic outflow boundary; We correct nodal values on the inviscid wall boundary as usual but need to project the velocity to the tangent direction after correction.

4.2 Stability

Generally speaking, enforcing a certain form of discrete conservation laws guarantees correct wave speed thus sometimes may “sharpen” solutions. For $p = 1$, the corrector based on the discrepancy distribution improved the accuracy to second order by fixing the dissipation Eq. (2.10). However, the lack of dissipation frequently leads to instability because there is no mechanism to eliminate spurious fluctuation or singularity in the numerical solutions.

Here we consider another corrector that implicitly brings in some diffusion. We use the example in the one-dimensional space to intuitively illustrate the idea. Suppose U is the numerical approximation of a conserved quantity. Let $U_{i+1/2}^*$ denote the value generated by the predictor at $x_{i+1/2}$, \bar{U}_i denote the cell average over $[x_{i-1/2}, x_{i+1/2}]$ computed from the data at $t = t^n$ and the balance of flux. Instead of correcting the nodal value by discrepancy distribution Eq. (2.3) we set

$$U_{i+1/2}^{n+1(2)} = \frac{\bar{U}_i \Delta x_i + \bar{U}_{i+1} \Delta x_{i+1}}{\Delta x_i + \Delta x_{i+1}}. \quad (4.1)$$

where we could interpret the nodal value $U_{i+1/2}$ as the average value of the solution u over the interval $[x_{i-1/2}, x_{i+3/2}]$. For simplicity, in the following analysis, we assume $\Delta x_i = \Delta x_{i+1} = \Delta x$.

When $p = 1$, $\bar{U}_i = d\bar{U}_i + \frac{1}{2}(U_{i-1/2}^* + U_{i+1/2}^*)$, we can relate the $U_{i+1/2}^{n+1(2)}$ with the corrected solution $U_{i+1/2}^{n+1(1)}$ that is computed with discrepancy distribution ($\beta_1 = \beta_2 = \frac{1}{2}$).

$$\begin{aligned} U_{i+1/2}^{n+1(2)} &= \frac{\bar{U}_i \Delta x_i + \bar{U}_{i+1} \Delta x_{i+1}}{\Delta x_i + \Delta x_{i+1}} \\ &= \frac{1}{2} \left(d\bar{U}_i + \frac{1}{2}(U_{i-1/2}^* + U_{i+1/2}^*) + d\bar{U}_{i+1} + \frac{1}{2}(U_{i+1/2}^* + U_{i+3/2}^*) \right) \\ &= \frac{1}{2} (d\bar{U}_i + d\bar{U}_{i+1}) + U_{i+1/2}^* + \frac{1}{4} (U_{i-1/2}^* - 2U_{i+1/2}^* + U_{i+3/2}^*) \\ &\approx U_{i+1/2}^{n+1(1)} + \frac{\Delta x^2}{4} \Delta U \Big|_{x_{i+1/2}}. \end{aligned} \quad (4.2)$$

When $p = 2$, $\bar{U}_i = d\bar{U}_i + \frac{1}{6}(U_{i-1/2}^* + 4U_i^* + U_{i+1/2}^*)$. Similarly, we can establish the following results.

$$\begin{aligned} U_{i+1/2}^{n+1(2)} &= \frac{1}{2} (d\bar{U}_i + d\bar{U}_{i+1}) + U_{i+1/2}^* + \frac{1}{12} (U_{i-1/2}^* - 2U_{i+1/2}^* + U_{i+3/2}^*) + \frac{4}{12} (U_i^* - 2U_{i+1/2}^* + U_{i+1}^*) \\ &\approx U_{i+1/2}^{n+1(1)} + \frac{\Delta x^2}{6} \Delta U \Big|_{x_{i+1/2}}. \\ U_i^{n+1(2)} &= \bar{U}_i = d\bar{U}_i + U_i^* + \frac{1}{6} (U_{i-1/2}^* - 2U_i^* + U_{i+1/2}^*) \\ &\approx U_{i+1/2}^{n+1(1)} + \frac{\Delta x^2}{24} \Delta U \Big|_{x_i}. \end{aligned} \quad (4.3)$$

Thus $U_{i+1/2}^{n+1(2)}$ can be interpreted as $U_{i+1/2}^{n+1(1)}$ plus artificial diffusion. At the same time, the diffusive

corrector Eq. (4.1) maintains a proper discrete conservation law. For $p = 1$,

$$\int_{\Omega} u^h dx = \sum_i \frac{1}{2} (U_{i-1/2}^{n+1(2)} + U_{i+1/2}^{n+1(2)}) \Delta x_i = \sum_i \bar{U}_i \Delta x. \quad (4.4)$$

Similar results hold for $p = 2$ or multidimensional cases.

To maintain the $p+1$ -th-order accuracy, we control the artificial dissipation by taking a weighted average of $U_{i+1/2}^{n+1(1)}$ and $U_{i+1/2}^{n+1(2)}$:

$$U_{i+1/2}^{n+1} = (1 - w)U_{i+1/2}^{n+1(1)} + wU_{i+1/2}^{n+1(2)}. \quad (4.5)$$

Numerical experiments on the Euler equations indicate that, when $p = 1$, a w as small as the mesh size $O(\Delta x)^1$ is sufficient to guarantee stability. In that case, the total diffusion introduced is of $O(\Delta x^3)$ which has negligible influence on the global accuracy.

For steady state problems, introducing a proper amount of artificial diffusion also speeds up convergence to the steady state solution. For example, we allow large diffusion, e.g., $w = 0.1$, in the first few thousands of iterations in order to reduce entropy error and form smooth solutions. Then the weight of diffusive correction should be reduced to the mesh size level to increase the accuracy.

4.3 Numerical Results for Advanced Test Cases

In the following test cases for steady state problems, we redefine the residual of a conserved variable as

$$R = \frac{1}{3} \sum_v \left| \sum_{T:v \in T} \int_{\partial T} \mathbf{F} \cdot \mathbf{n} dl \right|. \quad (4.6)$$

Note that our numerical schemes are not designed to minimize the residual defined above. As the solutions approach steady state, the above quantity might approach a non-zero number. We consider the solutions converged when the changes of residuals of ρ , $\rho \mathbf{u}$, ρE after 1000 steps of

¹The constant depends on the reference length L . In this thesis, we simply take $L = 1$ for all test cases.

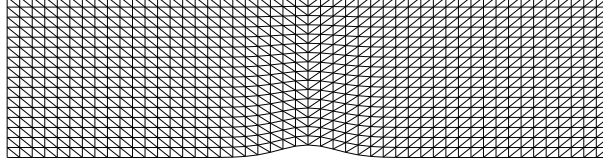


Figure 4.3: The sample mesh for the flow through a channel with smooth bump with $2 \times N \times 3N$ cells ($N = 16$).

iteration are all lower than a certain threshold ϵ_1 . The initial weight of the diffusive corrector is $w = w_0$. When the solutions start to converge, i.e., the changes of all 4 residuals after 1000 iterations are less than ϵ_1 of the previous residual, we adjust w to $k \times \min_{T,e} H_{T,e}$, the minimal height of the triangular cells. If the changes of all residuals every 1000 iterations are less than ϵ_2 after we lower the weight of the diffusive corrector, we end the iterations. To make the description of the parameter setting of the Euler solver concise, we write them as a tuple $(w_0, k, \epsilon_1, \epsilon_2)$.

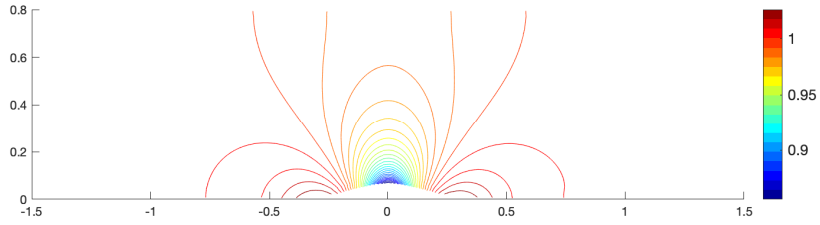
In all the test cases, the ratio of specific heats is $\gamma = 1.4$.

4.3.1 Gaussian Bump

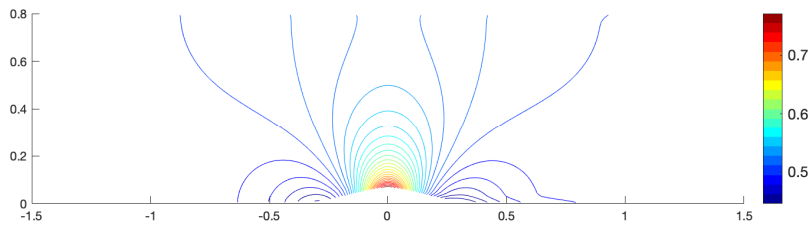
We consider the subsonic internal flow through a channel with a curved boundary. The domain is given by $-1.5 \leq x \leq 1.5$, $0.0625 \exp(-25x^2) \leq y \leq 0.8$. Structured meshes are used to study the performance (See Figure 4.3). We impose inflow boundary conditions on $x = -1.5$, subsonic outflow boundary condition on $x = 1.5$ and inviscid boundary conditions on $y = 0.8$ and $y = 0.0625 \exp(-25x^2)$.

We initialize the flow with the free-stream state, where $M_\infty = 0.5$, $p_\infty = 1/\gamma$, $\rho_\infty = 1$. The solver's parameter setting is $(w_0, k, \epsilon_1, \epsilon_2) = (0.1, 1, 1\%, 0.5\%)$. The steady state solutions described in Figure 4.4 are generated on a mesh with $6 \times 64^2 = 24576$ cells, 12545 nodes. The solutions are almost symmetric about $x = 0$. We plot the distribution of the deviation of entropy $ds = \ln(\gamma p / \rho^\gamma) - \ln(s_\infty)$ in Figure 4.4 (c). There is an entropy layer downstream the bump. But the error is bounded by a small number.

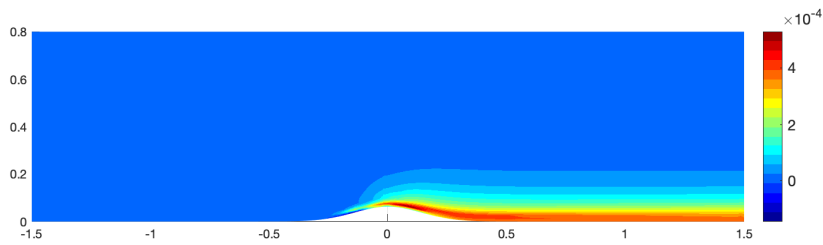
In Figure 4.5, we present the profiles of pressure coefficient $c_p = \frac{p - p_\infty}{\frac{\gamma}{2} \rho_\infty M_\infty^2}$ on the bottom wall generated by meshes of various sizes. In a sufficiently fine mesh, e.g., $N = 32$ or larger, the



(a) The contour of the steady-state density ρ .



(b) The contour of the steady-state Mach number field.



(c) The steady-state entropy deviation distribution: $ds = \ln(\gamma p / \rho^\gamma) - \ln(s_\infty)$

Figure 4.4: The steady state solutions for the flow through a channel with smooth bump on a mesh of $6N^2$ cells ($N = 64$).

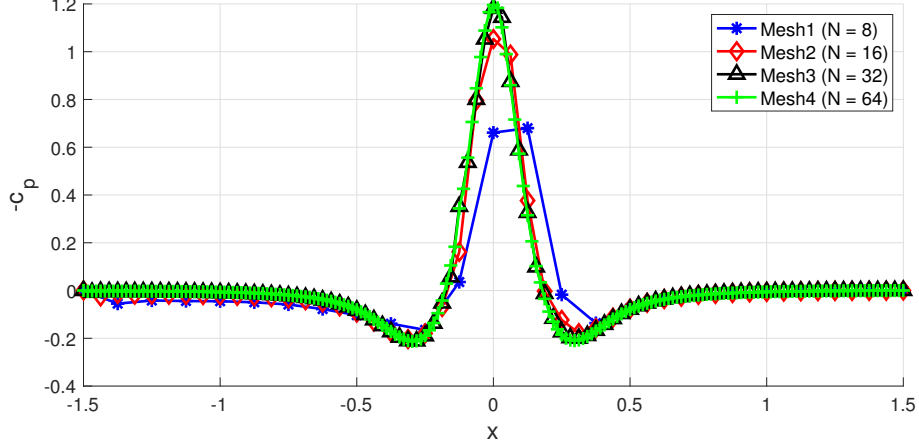


Figure 4.5: Flow through a channel with smooth bump: The steady-state pressure coefficient profiles on the bottom wall generated by different meshes.

graph appears to converge. To study the numerical accuracy of our numerical scheme, we use the following measure measurement.² (n_x, n_y) represents the normal vector at the bottom wall.

Entropy Error:

$$E_s = \sqrt{\frac{\int_{\Omega} \left(\frac{p}{p_{\infty}} \left(\frac{\rho_{\infty}}{\rho} \right)^{\gamma} - 1 \right)^2 d\Omega}{\int_{\Omega} 1 d\Omega}} = \frac{1}{\sqrt{\sum_T S_T}} \sqrt{\sum_v \left(\left(\frac{p}{p_{\infty}} \left(\frac{\rho_{\infty}}{\rho} \right)^{\gamma} - 1 \right)^2 \sum_{T: v \in T} \frac{S_T}{3} \right)};$$

Error in the coefficient of drag:

$$E_{c_d} = \left| \frac{\int_{\text{bottom}} (p - p_{\infty}) n_x dl}{0.0625 \frac{\gamma}{2} p_{\infty} M_{\infty}^2} - c_{d,\text{ref}} \right|;$$

Error in the coefficient of lift:

$$E_{c_l} = \left| \frac{\int_{\text{bottom}} (p - p_{\infty}) n_y dl}{0.0625 \frac{\gamma}{2} p_{\infty} M_{\infty}^2} - c_{l,\text{ref}} \right|.$$

We plot errors versus $1/\sqrt{nDOF}$ in Figure 4.6. (nDOF equals the total number of nodes in our scheme.) Apparently, our scheme with linear reconstruction ($p = 1$) achieves 2nd-order accuracy as expected.

²The exact entropy error should 0; $c_{l,\text{ref}} = 1.537095$, $c_{d,\text{ref}} = 0$.

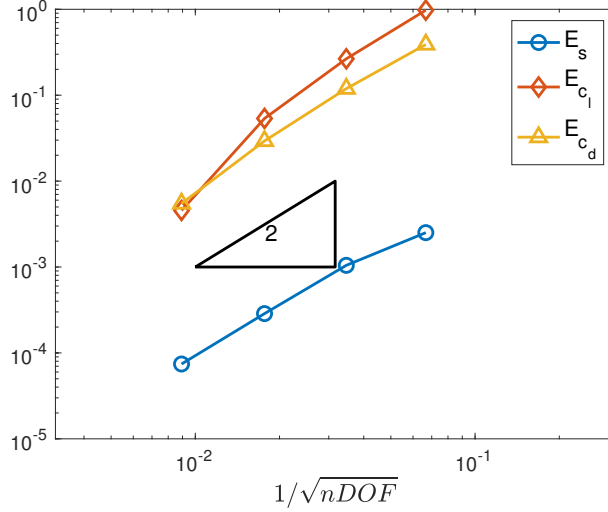


Figure 4.6: Flow through a channel with smooth bump: Convergence plot of errors for $p = 1$.

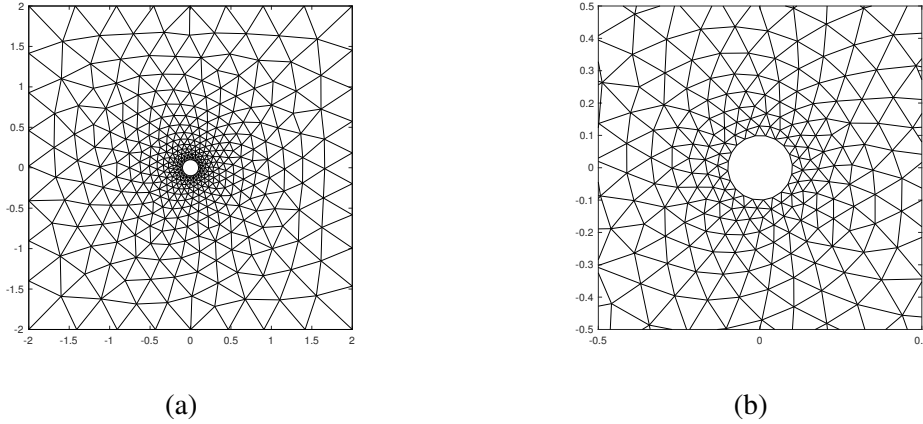


Figure 4.7: Mesh 1 for the cylinder case with 441 nodes and 829 cells

4.3.2 Inviscid Flow around the Cylinder

The flow around a cylinder is another classical test for the Euler equations [1]. We construct a domain $\Omega = [-2, 2] \times [-2, 2]$ with a cylinder of radius 0.1 centered at $(0, 0)$, with inflow boundary conditions on $x = -2$, $y = \pm 2$, outflow boundary conditions on $x = 2$ and inviscid boundary conditions on $x^2 + y^2 = 0.1^2$. A coarse mesh (Mesh 1) is generated by DistMesh, a mesh generator in Matlab by Persson and Strang [34]. See Figure 4.7. We refine Mesh 1 by splitting each triangular cell into 4 sub-cells to generate Mesh 2 (1711 nodes, 3316 cells) and then Mesh 3 (6738 nodes, 13264 cells). See Section D.1 for the details of mesh refinement.

The initial values are the same as the free-stream flow, where $M_\infty = 0.38$, $u = M_\infty$, $v = 0$, $p_\infty = 1/\gamma$, $\rho_\infty = 1$. ($\gamma = 1.4$.) The solver's parameter setting is $(w_0, k, \epsilon_1, \epsilon_2) = (0.1, 2, 1\%, 0.5\%)$. We present the profiles of solutions generated on Mesh 3 in Figure 4.8. There is a thin entropy layer around the inviscid surface due to the numerical viscosity. The current solutions exhibit a little bit of asymmetry which might be caused by the mesh quality. Overall, the solution quality is better than (or at least no worse than) the ones generated by other well-accepted second-order schemes [1]. We further conduct a convergence study of the entropy error. See Figure 4.9. The slope is roughly 1.6, which is satisfying given that the meshes are not fine enough compared to the curvature of the cylinder wall.

4.3.3 Inviscid Flow around the NACA0012 Airfoil

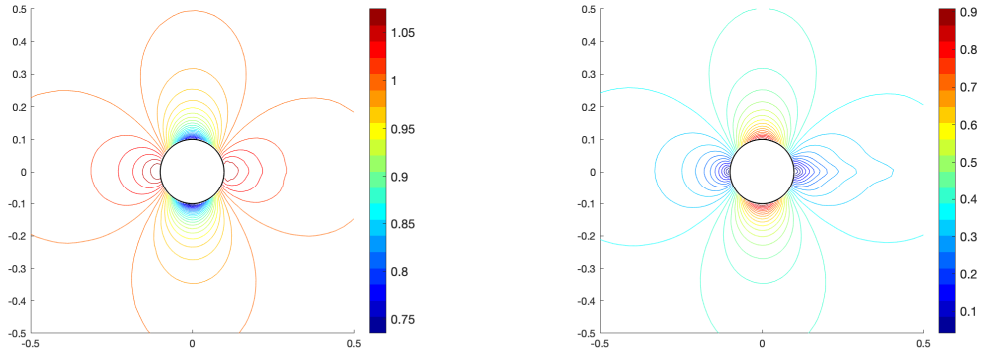
We consider flows around the NACA0012 Airfoil with various attack angle α and freestream Mach number M_∞ . The domain is defined as a large disk of radius 4 centered at $(2, 0)$. The boundary of the airfoil is described by the following formula.

$$y = \pm 0.6(0.2969\sqrt{x} - 0.1260x - 0.3516x^2 + 0.2843x^3 - 0.1036x^4). \quad (4.7)$$

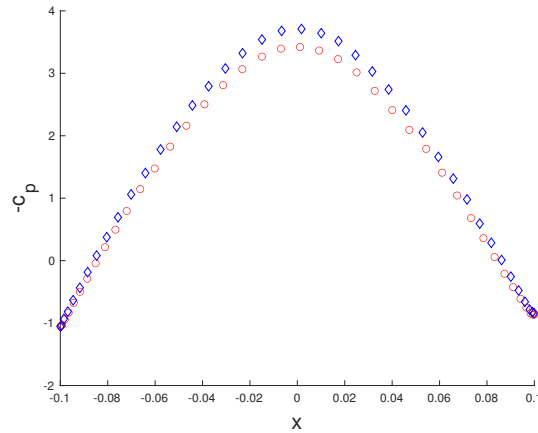
We impose inviscid boundary conditions on the surface of the airfoil. For a point (x, y) on the boundary of the domain, that is, $(x - 2)^2 + y^2 = 16$, if $x \cos \alpha + y \sin \alpha < 0$, we assign inflow boundary conditions, otherwise, outflow boundary conditions. DistMesh [34] is used to generate Mesh 1. See Figure 4.10. We further refine the mesh with the same method as we do in the previous test case. Mesh 2 has 3070 nodes, 5872 cells, Mesh 3 has 6738 nodes, 23488 cells.

The simulation starts with free-stream flow, where $u = M_\infty \cos \alpha$, $v = M_\infty \sin \alpha$, $p_\infty = 1/\gamma$, $\rho_\infty = 1$. For a mild test case where $M_\infty = 0.5$, $\alpha = 1^\circ$, we set $(w_0, k, \epsilon_1, \epsilon_2) = (0.1, 2, 1\%, 0.1\%)$. Figure 4.11 presents the steady state solutions generated on Mesh 3. Figure 4.12 is the convergence plot of the entropy errors on the hierarchical meshes. The slope is around 1.65.

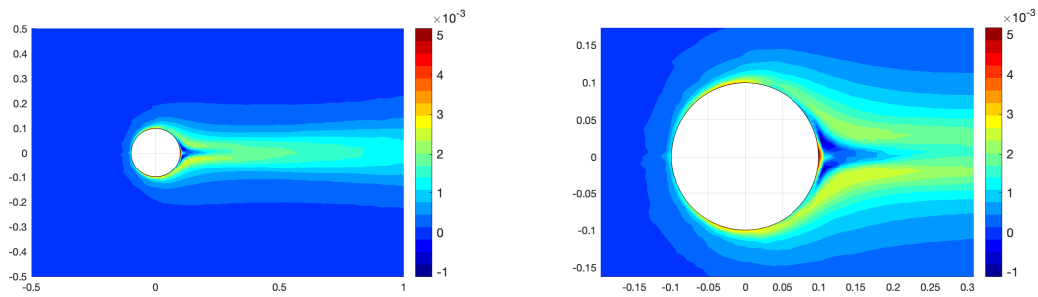
To demonstrate the robustness of our scheme, we further challenge the solver with transonic



(a) The contour of the steady-state density ρ . (b) The contour of the steady-state Mach number field. $M_{\max} = 0.9173$, $M_{\min} = 0.0054$.



(c) Pressure coefficient distribution on the inviscid surface. (Red: c_p on the upper surface; Blue: c_p on the lower surface.)



(d) The steady-state entropy deviation distribution: $ds = \ln(\gamma p / \rho^\gamma) - \ln(s_\infty)$. (e) The steady-state entropy deviation distribution (zoomed in around the inviscid surface).

Figure 4.8: The steady state solutions for the flow around a cylinder on Mesh 3.

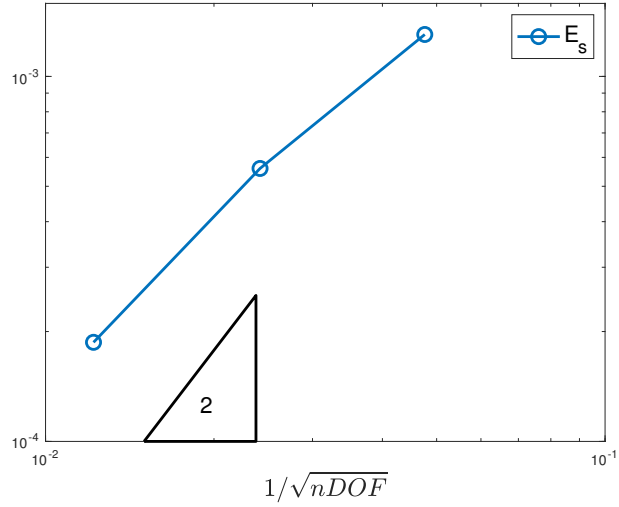


Figure 4.9: Flow around a cylinder: Convergence plot of entropy errors for $p = 1$.

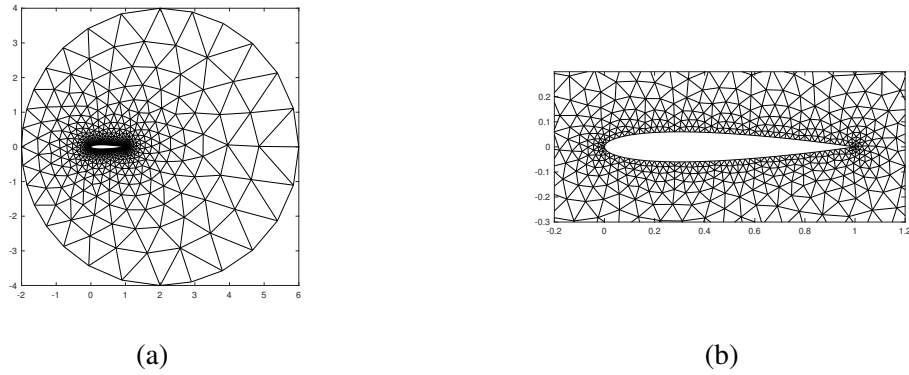


Figure 4.10: A sample mesh for the NACA0012 Airfoil case with 801 nodes and 1468 cells

test cases. Shock may form on both upper and lower surfaces of the airfoil. Note that no limiter is used here. We increase k to 5 to ensure stability. The numerical results generated by Mesh 3 are presented in Figure 4.13 ($M_\infty = 0.8$, $\alpha = 1.25^\circ$) and Figure 4.14 ($M_\infty = 0.85$, $\alpha = 1^\circ$). The solutions are qualitatively correct. Due to the fact that our domain boundary is not sufficiently far away from the airfoil surface and that we have no special treatment of discontinuous solutions, they are not as accurate as the exact ones.

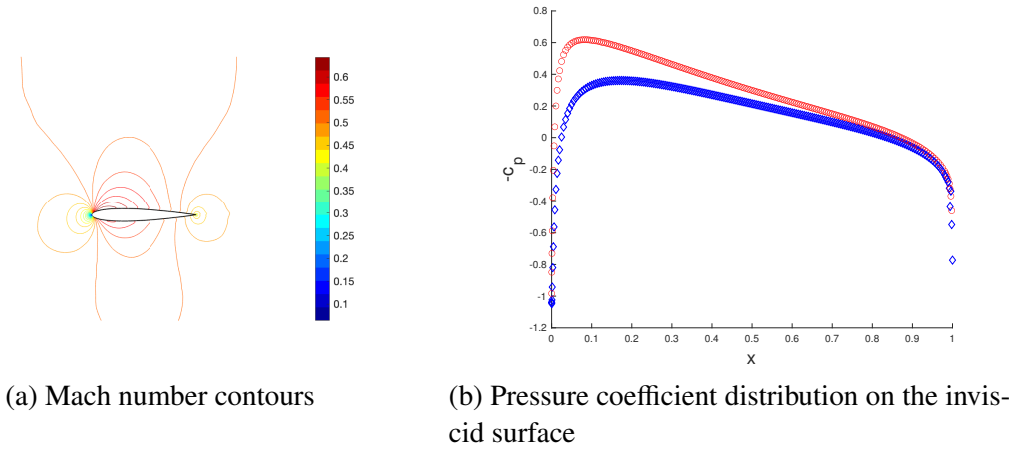


Figure 4.11: Subsonic flow around the NACA0012 Airfoil with $M_\infty = 0.5$, $\alpha = 1^\circ$.

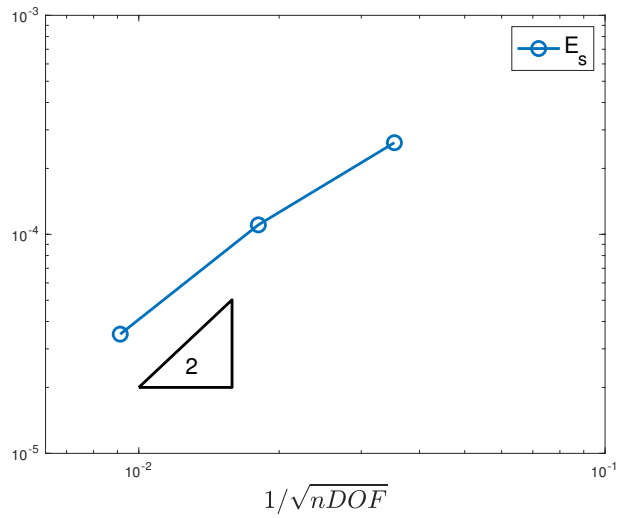


Figure 4.12: Subsonic flow around the NACA0012 ($M_\infty = 0.5$, $\alpha = 1^\circ$): Convergence plot of entropy errors for $p = 1$.

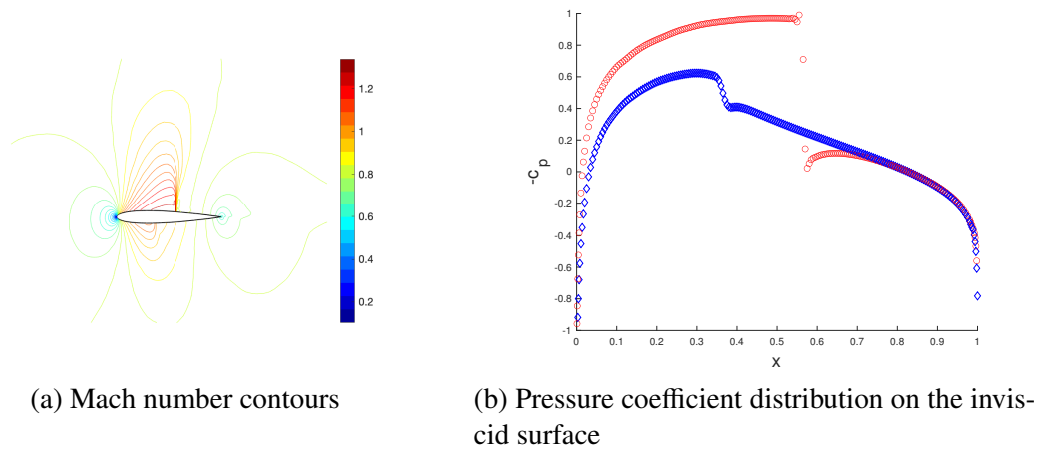
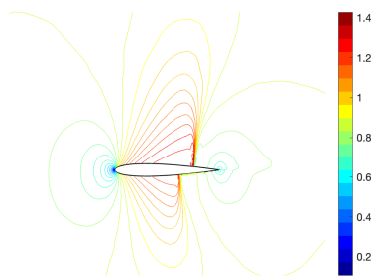
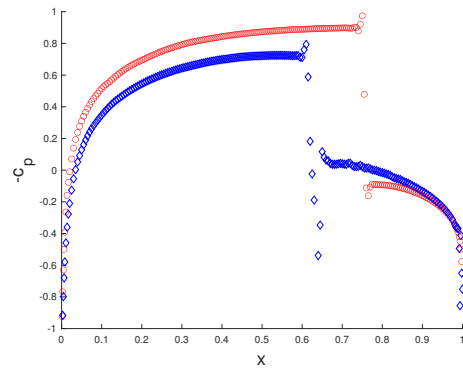


Figure 4.13: Transonic flow around the NACA0012 Airfoil with $M_\infty = 0.8$, $\alpha = 1.25^\circ$.



(a) Mach number contours



(b) Pressure coefficient distribution on the inviscid surface

Figure 4.14: Transonic flow around the NACA0012 Airfoil with $M_\infty = 0.85$, $\alpha = 1^\circ$.

CHAPTER 5

Including Viscosity I: A Traditional Strategy

5.1 A Benchmark Scheme for Viscosity and Diffusion

In this chapter, we discuss a simple benchmark scheme for the Navier-Stokes equations in the family of Active Flux schemes that we introduced in Chapter 2. This benchmark scheme uses piecewise linear functions for reconstruction ($p = 1$).

$$\begin{cases} \frac{\partial \rho}{\partial t} + \mathbf{v} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{v} = 0 \\ \rho \frac{\partial \mathbf{v}}{\partial t} + \rho \mathbf{v} \cdot \nabla \mathbf{v} + \nabla p = \nabla \cdot \mathbf{T} \\ \frac{\partial p}{\partial t} + \mathbf{v} \cdot \nabla p + \gamma p \nabla \cdot \mathbf{v} = \frac{\gamma \mu}{Pr} \nabla \cdot \left(\nabla \frac{p}{q} \right) + (\gamma - 1) \mathbf{T} : \nabla \mathbf{v}, \end{cases} \quad (5.1)$$

where

$$\mathbf{T} = \lambda \nabla \cdot \mathbf{v} + \mu (\nabla \mathbf{v} + (\nabla \mathbf{v})^T).$$

The presence of heat conduction and viscosity, which are essentially diffusive, drastically changes the nature of the system and poses new challenges in numerical simulation. Unlike a hyperbolic system where information travels at finite speed, a system containing diffusion allows small perturbation to impact the whole domain immediately. In other words, the diffusive terms make the system stiff. Thus to ensure stability, we have to take a small time step size, $\Delta t \sim h^2$, if we use explicit methods for evolution¹. (h represents the size of the smallest cell in the mesh.) Besides,

¹ Although time step size of the same order of the mesh size h is allowed in implicit methods, solving globally coupled equations in every time step is also computationally costly. In this work, we focus on explicit methods.

the terms that model heat conduction and viscosity in the Navier-Stokes equations are second-order derivatives of the primitive variables. Given that the reconstructed numerical solutions in the Active Flux method are of class C^0 , finding an efficient and accurate way to approximate second-order derivatives on unstructured meshes is another challenge.

5.1.1 Approximation of Derivatives

We notice in Figure 3.1 that the approximation of derivatives based on the exact solutions to linear acoustics is not ideal when $p = 1$. The approximation of second-order derivatives shows no convergence when we refine the mesh.

Consider a continuous function $u(x, y)$. Let $\mathbf{n} = (n_1, n_2)$ be outward-pointing normal vector on the boundary of cell T . The cell average of the gradient (u_x, u_y) of u on cell T is given by

$$\bar{u}_{xT} = \frac{1}{S_T} \oint_{\partial T} un_1 dl, \quad \bar{u}_{yT} = \frac{1}{S_T} \oint_{\partial T} un_2 dl. \quad (5.2)$$

We store the nodal values of u at the vertices on the triangular mesh in U . Let u^h denote the reconstructed continuous piecewise linear function from U . We approximate the gradient of u^h at node v by taking the weighted average of \bar{u}_{xT}^h and \bar{u}_{yT}^h from the adjacent cells. This can be regarded as the counterpart of the one-dimensional central difference approximation of derivatives on the two-dimensional unstructured triangular mesh. Let $U_{z,v}$ represent the approximated derivative of u^h with respect to variable z on node v .

$$U_{x,v} = \frac{\sum_{\{T: v \in T\}} \bar{u}_{xT}^h S_T}{\sum_{\{T: v \in T\}} S_T}, \quad U_{y,v} = \frac{\sum_{\{T: v \in T\}} \bar{u}_{yT}^h S_T}{\sum_{\{T: v \in T\}} S_T}. \quad (5.3)$$

In essence, the gradient of u at node v is approximated by the average gradient over the union of cells adjacent to v in Eq. (5.3). The accuracy should be between first order and second order. We expect that the more ‘‘symmetric’’ the neighboring cells around v the more accurate the approximation. This strategy is more efficient than the one based on the formulas for the linear acoustic system. Only nodal values, instead of the coefficients of reconstructed polynomials, are used to

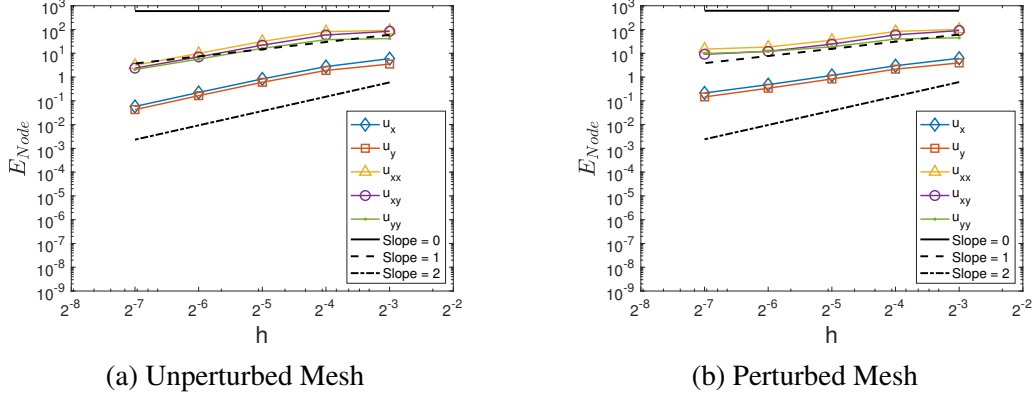


Figure 5.1: Accuracy of the alternative derivative approximation strategy for $p = 1$ on the unperturbed mesh and the perturbed mesh.

evaluate Eq. (5.2).

We repeat this process on U_x and U_y to approximate second-order derivatives. The approximated cross partial derivative takes the average of $U_{xy,v}$ and $U_{yx,v}$. Note that this step actually implicitly uses reconstructed continuous piecewise linear functions from $U_{x,v}$ and $U_{y,v}$. Thus the accuracy of second-order derivatives is better than the approximation based on the formula of linear acoustics.

We conduct numerical experiments to examine the quality of approximation. Set $u(x, y) = \sin(6\pi x) \sin(4\pi y)$. Figure 5.1 indicates that this strategy gives roughly second-order-accurate approximation of 1st- and 2nd-order derivatives on unperturbed mesh, and 1st-order-accurate approximation on perturbed meshes. The meshes are described in Figure 2.12.

Note that in terms of approximating derivatives, this strategy Eq. (5.3) is neither a perfect idea nor the only choice. For example, one may consider fitting the values on adjacent nodes into the Taylor expansion at node v

$$u(x, y) = U_v + U_{x,v}\Delta x + U_{y,v}\Delta y + \frac{1}{2}U_{xx,v}\Delta x^2 + U_{xy,v}\Delta x\Delta y + \frac{1}{2}U_{yy,v}\Delta y^2$$

with the method of least squares to determine the coefficients and thus get approximation of derivatives, or consider the strategy used in the finite element method. But the cost of our current strategy is relatively low while the accuracy is good enough for constructing a second-order predictor. Thus

we stick to it for benchmark results.

5.1.2 The Benchmark Numerical Scheme for the Heat Equation

We consider the heat equation as a model problem.

$$\begin{cases} u_t = \nu \nabla \cdot \mathbf{q}, \\ \mathbf{q} = \nabla u. \end{cases} \quad (5.4)$$

For simplicity, we set $\nu = 1$. To evolve the numerical solution from $t = t^n$ to t^{n+1} , we approximate \mathbf{q}^n and $\nabla \mathbf{q}^n$ as discussed above. The predictor simply uses the Euler method: $u^* = u^n + \Delta t \nabla \cdot \mathbf{q}^n$. Then we estimate \mathbf{q}^* from u^* , which will be used in evaluating the temporal integral of heat flux through the cell boundaries. In the heat equation, u is the conserved quantity. We compute the discrepancy following the definition given by Eq. (2.7).

$$d\bar{U}_T = \frac{-1}{S_T} \left(\int_{\partial T} \int_{t^n}^{t^{n+1}} -\mathbf{q}^h \cdot \mathbf{n} dt dl + \iint_T u^h(\mathbf{x}, t^{n+1}) - u^h(\mathbf{x}, t^n) d\sigma \right). \quad (5.5)$$

The corrector (discrepancy distribution) remains the same as described in Eq. (2.8). We test this numerical scheme for both transient and steady-steady solutions to the heat equation. See Section 5.2.

5.1.3 The Benchmark Numerical Scheme for the Navier-Stokes Equations

We construct a simple low-order scheme for the Navier-Stokes equations on top of the Euler solver (See Chapter 3). The viscous stress tensor \mathbf{T} , the heat flux \mathbf{Q} , and their gradients are estimated by the strategy discussed in Section 5.1.1. We simply add Δt times terms that model diffusion and viscosity to the Euler predictor to build the predictor for the Navier-Stokes Equations. For example,

$$\mathbf{v}^* = \mathbf{v}_{\text{Euler}}^* + \frac{\nabla \cdot \mathbf{T}}{\rho} \Delta t, \quad (5.6)$$

where $\mathbf{v}_{\text{Euler}}^*$ is computed by the predictor from the Euler solver. The accuracy of this simple predictor is not better than second order. Like the benchmark scheme for the heat equation, we also evaluate \mathbf{T}^* and \mathbf{Q}^* again after the prediction to compute the discrepancy in the cell averages of the conserved variables. No change is made to discrepancy distribution process. For the test cases that involve complicated geometry, the numerical strategy described in Section 4.2 is adopted to enhance stability. We use this scheme to compute the steady viscous flow around a cylinder. See Section 5.2.

5.2 Numerical Experiments

5.2.1 Transient Problems of the Heat Equation

We consider a domain $\Omega = [0, 1] \times [0, 1]$ with periodic boundary conditions. It is not hard to verify that

$$u(x, y, t) = e^{-4\pi^2(k_1^2+k_2^2)t} \sin(2k_1\pi x) \sin(2k_2\pi y), \quad \forall k_1, k_2 \in \mathbb{Z}$$

solves equation (5.4).

We set $k_1 = 1$, $k_2 = 2$, initial value $u_0(x, y) = \sin(2k_1\pi x) \sin(2k_2\pi y)$ in our tests. Since our scheme is explicit, we set $\Delta t = \frac{1}{2} \left(\min_{T,e} H_{T,e} \right)^2$, where H_e is the height of cell T rising from edge e . Errors are measured at $T = 0.01$ in L_1 -norm (Equation (2.19)). We run the tests on both unperturbed mesh and perturbed mesh (See Figure 2.12). Figure 5.2 indicates the benchmark scheme achieves at least first-order accuracy. Interestingly, we notice that in the simulation of the heat equation, the corrector does not bring second-order accuracy as it does in Chapter 2. However, if we remove the corrector, the numerical scheme becomes unstable.

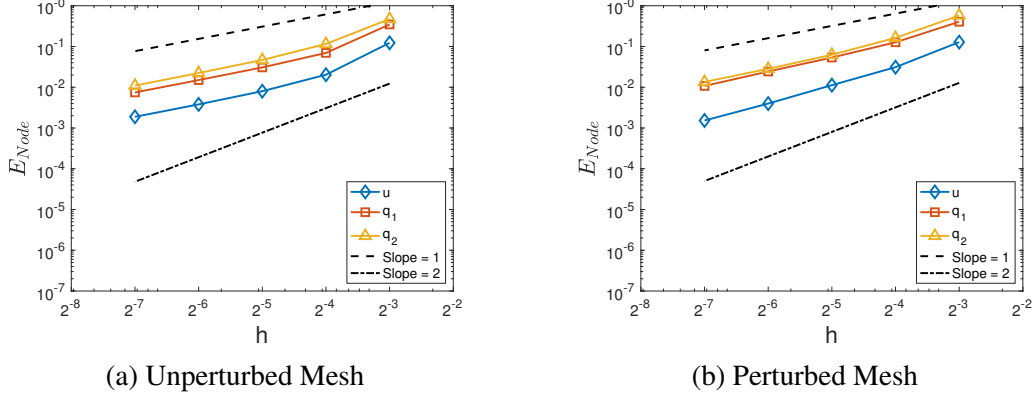


Figure 5.2: Convergence rates of the benchmark scheme for the transient solutions to the heat equation on the unperturbed mesh and the perturbed mesh.

5.2.2 Steady-State Problems of the Heat Equation

We consider the heat equation on a domain $\Omega = [0, 1] \times [0, 1]$ with Dirichlet boundary conditions $u(x, 0) = u(0, y) = 0$, $u(x, 1) = \sin(\pi x)$, $u(1, y) = \sin(\pi y)$. The exact solution is given by

$$u(x, y) = \frac{\sinh(\pi x) \sin(\pi y) + \sinh(\pi y) \sin(\pi x)}{\sinh(\pi)}.$$

We set the initial value

$$u_0(x, y) = \frac{\sinh(\pi x) \sin(\pi y) + \sinh(\pi y) \sin(\pi x)}{\sinh(\pi)} + \sin(\pi x) \sin(\pi y),$$

and time step size $\Delta t = \frac{1}{2} \left(\min_{T,e} H_{T,e} \right)^2$. To enforce the boundary conditions, we do not modify the nodal values of u on the boundary in both prediction stage and correction stage. The simulation is run on both unperturbed and perturbed meshes. (Recall the construction of the meshes in Section 2.5.2.) On a mesh with $2N^2$ cells, we stop the iteration when the change of U measured in L_1 -norm is less than 10^{-8} after every N^2 steps of iteration. Figure 5.3 shows that both the primary variable u and its gradient \mathbf{q} have second-order accuracy on unperturbed mesh; on the perturbed mesh, error in the primary variable u converges with rate 2 while the errors in the gradient \mathbf{q} which is estimated from the nodal values of u only converge with a rate slightly above 1.

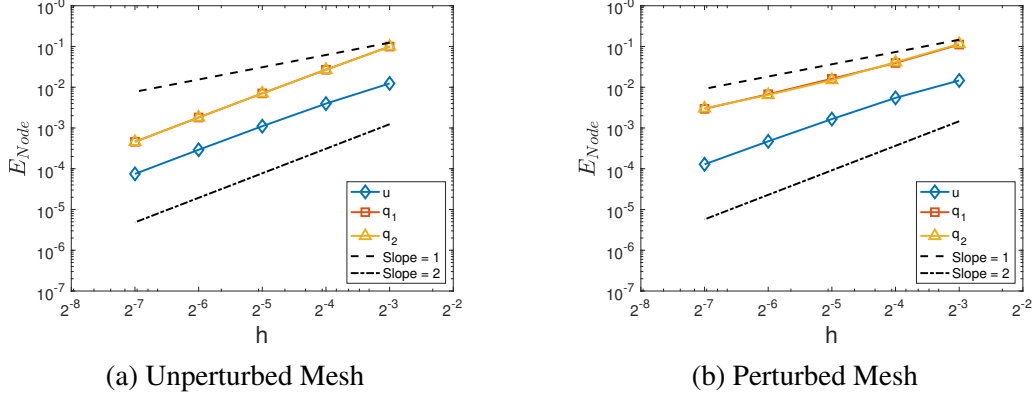


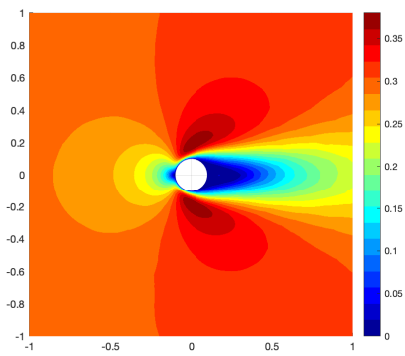
Figure 5.3: Convergence rates of the benchmark scheme for the steady-state solutions to the heat equation on the unperturbed mesh and the perturbed mesh.

5.2.3 The Steady Viscous Flow around A Cylinder

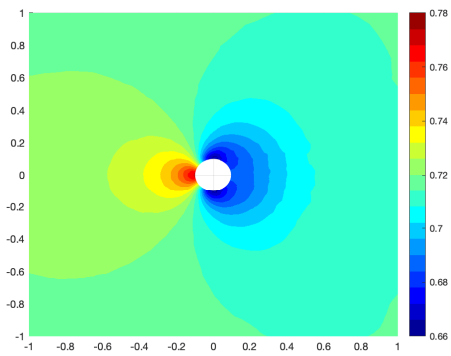
We adopt the same geometric setting and meshes that are used for the Euler equations (Section 4.3.2). No-slip and adiabatic boundary conditions are imposed on the boundary of the cylinder. The velocity at the no-slip boundary is always 0 and we do not modify them in predictor or corrector. When computing the cell averages via the balance of fluxes, we set the heat flux \mathbf{Q} through the edge on the adiabatic wall to be zero. Note that \mathbf{Q} , or the gradient of temperature, is not necessarily to be 0 on the adiabatic boundary.

The simulation is started with free-stream flow, where $M_\infty = 0.3$, $u = M_\infty$, $v = 0$, $p_\infty = 1/\gamma$, $\rho_\infty = 1$, $\gamma = 1.4$. The diameter of the cylinder is $D = 0.2$. We set $\mu = 10^{-3}$, $\lambda = -2\mu/3$, which gives a Reynolds number $Re = \frac{\rho_\infty M_\infty D}{\mu} = 60$. $Pr = 0.72$. We take a small time step $\Delta t = \left(\min_{T,e} H_{T,e}\right)^2$. In the following numerical experiments, we fix the total time $T = \Delta t \times M$, where M is the total number of iterations. The initial weight of the diffusive corrector is $w = w_0 = 0.01$. When the change of all residuals after $M/10$ iterations is less than 10%, we reduce the weight w to $0.2 \times \min_{T,e} H_{T,e}$.

We present the steady-state Mach number distribution and the pressure field generated on Mesh 3 (6738 nodes, 13264 cells) after $M = 512000$ iterations in Figure 5.4.



(a) Mach Number Field



(b) Pressure Field

Figure 5.4: The steady state solutions for the viscous flow around a cylinder on Mesh 3.

CHAPTER 6

Including Viscosity II: The Hyperbolic Reformulations

As expected, the benchmark scheme for the Navier-Stokes equations introduced in the previous chapter is inefficient due to the stiffness caused by viscosity. In the steady-state problem, it takes a long time for the numerical solution to converge. When we halve the mesh size in two-dimensional space, i.e., equally splitting every triangular cell into 4 sub-triangles, the total simulation time will be 16-fold if we do not boost the scheme with techniques like the multigrid method. See Table 6.1.

In this chapter, we consider the hyperbolic reformulation inspired by Cattaneos hyperbolic heat-conduction equation [8] as a numerical strategy to improve accuracy and efficiency. Some application of such Cattaneo-type hyperbolic models in advection-diffusion equations and the Navier-Stokes equations¹ has been explored in [31, 32, 44, 25], where various combinations of equations and numerical schemes are adopted. Note that researchers also have pursued other hyperbolic models for the Navier-Stokes equations out of motivations in physics. For example, Peshkov and Romenski et al. suggested an alternative hyperbolic model, where viscosity is modeled by the distortion field of particles, in pursuit of a unified framework for viscous fluids and elastoplastic solids [35, 12].

¹The Cattaneo model is also used in hyperbolic reformulations of dispersive systems [7].

6.1 Hyperbolic Reformulations of Equations

6.1.1 The Hyperbolic Heat Equations

We consider Cattaneo's hyperbolic heat-conduction equations [8] as a model problem.

$$\begin{cases} \frac{\partial u}{\partial t} = \nu \nabla \cdot \mathbf{q} \\ \frac{\partial \mathbf{q}}{\partial t} = \frac{1}{\tau} (\nabla u - \mathbf{q}), \end{cases} \quad (6.1)$$

where τ is the relaxation time. Note that the key step of constructing a hyperbolic model is to introduce independent relaxed variables for derivatives of the primary variables, which eliminates the high-order derivatives in the original equations. In the case of the heat equation, \mathbf{q} is introduced as the relaxed gradient of u ,

$$\frac{\partial \mathbf{q}}{\partial t} = \frac{1}{\tau} (\nabla u - \mathbf{q}).$$

When the system reaches the steady state, $\mathbf{q} = \nabla u$, $\nabla \cdot \mathbf{q} = 0$, which is identical to the steady solution to the heat equation Eq. (5.4). Thus the hyperbolic reformulation Eq.(6.1) can be used to compute the steady state of the original equation Eq. (5.4). The same idea is applied to solve the steady-state Navier-Stokes equations.

The advantage of this strategy lies in the hyperbolicity of the new system Eq. (6.1). Both analysis of characteristics and the exact solutions Eq. (B.15) indicate that the solutions at $(\mathbf{x}_0, t_0 + \Delta t)$ only depend on the data at $t = t_0$ within a disk/ball centered at $\mathbf{x} = \mathbf{x}_0$ with radius $\Delta t \sqrt{\nu/\tau}$. The finite speed of information propagation allows a large time step size of the same order of the mesh h . Although it is true that the hyperbolic system may require longer physical time in the equations to be steady and more computation is required in one step of time marching, a carefully chosen relaxation time τ will significantly reduce the number of iterations. When we reach steady state, we not only have the primary variable u but also get its gradient \mathbf{q} of the same accuracy. We would refer readers to the documentation of our elementary but detailed and (hopefully insightful) analysis on

the one-dimensional hyperbolic heat equations and hyperbolic advection-diffusion equations in Appendix C, where we discuss how to pick the optimal relaxation time and why the stiffness is reduced. Note that the hyperbolic heat equations have been studied by many researchers from various aspects via different paths. For example, the dispersion relation of waves in the hyperbolic system, which depends on τ , has been investigated in [38, 44]. Luckily, all the analyses, including ours, are consistent.

We demonstrate the effectiveness of the hyperbolic reformulation in reducing the computational cost by comparing it with the benchmark scheme in Section 5.1.2. To make the comparison fair, we introduce an equally simple second-order scheme based on the Active Flux method ². The predictor is simply the Euler method:

$$\begin{aligned} u^* &= u^n + \Delta t \nabla \cdot \mathbf{q}, \\ \mathbf{q}^* &= \mathbf{q}^n + \frac{\Delta t}{\tau} (\nabla u - \mathbf{q}). \end{aligned} \quad (6.2)$$

The first-order derivatives are computed by the acoustic-formula-based derivative approximation.

Note that the second equation in Eq. (6.1) can be rewritten in the conservation form

$$\frac{\partial}{\partial t} (e^{t/\tau} \mathbf{q}) = \frac{1}{\tau} \nabla (e^{t/\tau} u). \quad (6.3)$$

We have the following definition of the discrepancy in cell averages of u and \mathbf{q} .

$$\begin{aligned} d\bar{U}_T &= \frac{-1}{S_T} \left(\oint_{\partial T} \int_{t^n}^{t^{n+1}} -\mathbf{q}^h \cdot \mathbf{n} \, dt \, dl + \iint_T u^h(\mathbf{x}, t^{n+1}) - u^h(\mathbf{x}, t^n) d\sigma \right), \\ d\bar{\mathbf{Q}}_T &= \frac{-e^{-t^{n+1}/\tau}}{S_T} \left(\oint_{\partial T} \frac{1}{\tau} \mathbf{n} \int_{t^n}^{t^{n+1}} -e^{t/\tau} u^h \, dt \, dl + \iint_T e^{t^{n+1}/\tau} \mathbf{q}^h(\mathbf{x}, t^{n+1}) - e^{t^n/\tau} \mathbf{q}^h(\mathbf{x}, t^n) d\sigma \right). \end{aligned} \quad (6.4)$$

Then we use Eq. (2.8) to distribute the discrepancy. The numerical results can be found in Section 6.2.1.

Notes:

²Higher-order schemes that use the exact solutions Eq. (B.15) in the predictor have been developed. See our AIAA conference paper [18].

1. We know that as τ approaches 0, the solution to Eq. (6.1) is getting closer to the actual diffusion. Interestingly, when τ is close to zero, Eq. (6.4) indicates that the discrepancy in the cell average of \mathbf{q} is almost determined by u^* and \mathbf{q}^* . At the same time, the small τ also forces us to take a small time step. This makes the scheme resemble the benchmark scheme in Section 5.1.2.
2. As we mention in Appendix C, the stiffness caused by diffusion in the original heat equation is transformed into the source term in the hyperbolic heat equations. If we use the exact solutions as predictor, the only restriction on time step size is from the wave speed $\sqrt{\nu/\tau}$. However, when we use the predictor based on the Euler method, there may be further restrictions related to τ .

6.1.2 The Hyperbolic Viscosity and the Decomposition of Waves

In this subsection, we present a novel hyperbolic reformulation of the viscosity. Consider the sub-system of viscosity in the Navier-Stokes equations Eq.(5.1),

$$\begin{cases} \rho \frac{\partial \mathbf{v}}{\partial t} = \nabla \cdot \mathbf{T}, \\ \mathbf{T} = \lambda(\nabla \cdot \mathbf{v})\mathbf{I} + \mu(\nabla \cdot \mathbf{v} + \nabla \cdot \mathbf{v}^T). \end{cases} \quad (6.5)$$

A natural choice of hyperbolic reformulation, which is also adopted by [31, 30], is given by

$$\frac{\partial \mathbf{T}}{\partial t} = \frac{\lambda \nabla \cdot \mathbf{v} + \mu(\nabla \mathbf{v} + (\nabla \mathbf{v})^T) - \mathbf{T}}{\tau}. \quad (6.6)$$

The analysis based on the eigenvalues of the hyperbolic system in [31, 30] indicates that the hyperbolic reformulation introduces two different waves. Here we take a different path to explicitly analyze the wave structures of the hyperbolic viscosity.

With some proper transformation (which can be found in any textbooks about solid mechanics or fluid mechanics, e.g., [17]), we have

$$\nabla \cdot (\lambda(\nabla \cdot \mathbf{v})\mathbf{I} + \mu(\nabla \cdot \mathbf{v} + \nabla \cdot \mathbf{v}^T)) = -\mu \nabla \times (\nabla \times \mathbf{v}) + (2\mu + \lambda) \nabla(\nabla \cdot \mathbf{v}). \quad (6.7)$$

If we decompose the velocity field \mathbf{v} into divergence-free part and curl-free part by Helmholtz theorem, i.e., $\mathbf{v}_1 + \mathbf{v}_2 = \mathbf{v}$, $\nabla \cdot \mathbf{v}_1 = 0$, $\nabla \times \mathbf{v}_2 = 0$, it is not hard to derive the governing equations of \mathbf{v}_1 and \mathbf{v}_2 .

$$\begin{cases} \rho \frac{\partial \mathbf{v}_1}{\partial t} = -\mu \nabla \times \omega, \\ \frac{\partial \omega}{\partial t} = \frac{1}{\tau} (\nabla \times \mathbf{v}_1 - \omega). \end{cases} \quad (6.8)$$

$$\begin{cases} \rho \frac{\partial \mathbf{v}_2}{\partial t} = (2\mu + \lambda) \nabla \psi, \\ \frac{\partial \psi}{\partial t} = \frac{1}{\tau} (\nabla \cdot \mathbf{v}_2 - \psi). \end{cases} \quad (6.9)$$

These two waves can be regarded as analogs of the transverse/shear-type wave and compressional/sound-type wave in elastic materials. The corresponding wave speeds are $\sqrt{\mu/\tau}$ and $\sqrt{(2\mu + \lambda)/\tau}$. The behavior of each wave is similar to the wave in the hyperbolic heat equations.

The decomposition also inspires us to write the hyperbolic reformulation of the viscosity in a more compact way: we only introduce relaxed variables for $\nabla \times \mathbf{v}$ and $\nabla \cdot \mathbf{v}$.

$$\begin{cases} \rho \frac{\partial \mathbf{v}}{\partial t} = -\mu \nabla \times \omega + (2\mu + \lambda) \nabla \psi; \\ \frac{\partial \omega}{\partial t} = \frac{1}{\tau} (\nabla \times \mathbf{v} - \omega); \\ \frac{\partial \psi}{\partial t} = \frac{1}{\tau} (\nabla \cdot \mathbf{v} - \psi); \end{cases} \quad (6.10)$$

Applying the similar techniques that we use for deriving the exact solutions to the hyperbolic heat equations in Appendix B.2, we can also derive the exact solutions to Eq. (6.10). See Appendix B.3.

6.1.3 The Hyperbolic Navier-Stokes Equations

We further extend the hyperbolic models for heat conduction Eq. (6.1) and viscosity Eq. (6.10) to the hyperbolic reformulation of the full Navier-Stokes equations in two-dimensional space.

$$\begin{cases} \frac{\partial \rho}{\partial t} + \mathbf{v} \cdot \nabla \rho + \rho \nabla \cdot \mathbf{v} = 0, \\ \rho \frac{\partial \mathbf{v}}{\partial t} + \rho \mathbf{v} \cdot \nabla \mathbf{v} + \nabla p = \nabla \cdot \mathbf{T}, \\ \frac{\partial p}{\partial t} + \mathbf{v} \cdot \nabla p + \gamma p \nabla \cdot \mathbf{v} = \frac{\gamma \mu}{Pr} \nabla \cdot \mathbf{q} + (\gamma - 1) \mathbf{T} : \nabla \mathbf{v}, \end{cases} \quad (6.11)$$

where we introduce relaxed variables

$$\frac{\partial \mathbf{q}}{\partial t} = \frac{1}{\tau} \left(\nabla \left(\frac{p}{\rho} \right) - \nabla \mathbf{q} \right), \quad (6.12)$$

$$\begin{aligned} \frac{\partial \psi}{\partial t} &= \frac{1}{\tau} (\nabla \cdot \mathbf{v} - \psi), \\ \frac{\partial \omega}{\partial t} &= \frac{1}{\tau} (\nabla \times \mathbf{v} - \omega), \\ \frac{\partial r_1}{\partial t} &= \frac{1}{\tau} \left(\frac{\partial v_1}{\partial x} - \frac{\partial v_2}{\partial y} - r_1 \right), \\ \frac{\partial r_2}{\partial t} &= \frac{1}{\tau} \left(\frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x} - r_2 \right). \end{aligned} \quad (6.13)$$

Then we set

$$\nabla \cdot \mathbf{T} = -\mu \nabla \times \omega + (2\mu + \lambda) \nabla \psi, \quad (6.14)$$

$$\mathbf{T} = \begin{pmatrix} \lambda \psi + \mu(\psi + r_1) & \mu r_2 \\ \mu r_2 & \lambda \psi + \mu(\psi - r_1) \end{pmatrix}. \quad (6.15)$$

Note: In two-dimensional space, $\omega = (0, 0, \omega_k)^T$, we simply write

$$\omega = \omega_k = -\frac{\partial v_1}{\partial y} + \frac{\partial v_2}{\partial x}.$$

Then

$$\nabla \times \omega = \left(\frac{\partial}{\partial y} \omega, -\frac{\partial}{\partial x} \omega \right)^T.$$

To compare the benchmark scheme based on the traditional Navier-Stokes equations and the ones that combine our hyperbolic model and the Active Flux method, we consider a simple 2nd-order scheme. We use the Euler method, $\mathbf{u}^* = \mathbf{u}^n + \Delta t \mathbf{u}_t$, as the predictor. In the corrector, we define the discrepancy of conserved variables $\rho, \rho \mathbf{v}, \frac{1}{2} \rho \mathbf{v} \cdot \mathbf{v} + \frac{p}{\gamma-1}$ in the regular way, and handle the discrepancy of relaxed variables in the same way as Eq.(6.4). We distribute the discrepancy as usual (Eq. (2.8)). To ensure stability, the numerical strategy described in Section 4.2 is used for test cases with complex geometry.

6.2 Numerical Results

6.2.1 Steady State of the Heat Equation

The test case is generally the same as the one in Section 5.2.2. The relaxation time $\tau = 0.025$. In the simulation of the hyperbolic heat equation, we set initial values

$$u_0(x, y) = \frac{\sinh(\pi x) \sin(\pi y) + \sinh(\pi y) \sin(\pi x)}{\sinh(\pi)} + \sin(\pi x) \sin(\pi y), \quad \mathbf{q}_0 = (0, 0).$$

$u(x, y, t)$ is supposed to satisfy the Dirichlet boundary conditions and the numerical treatment remains the same. No restriction is imposed on the boundary values of \mathbf{q} . We take the time step $\Delta t = \min\left(0.9 \frac{1}{\sqrt{v/\tau}} \min_{T,e} H_{T,e}, \tau\right)$. The tests are run on both unperturbed and perturbed mesh as usual. On a mesh with $2N^2$ cells, we stop the iteration when the change of U measured in L_1 -norm is less than 10^{-10} after every N steps of iteration. Figure 6.1 indicates that the scheme that combines the hyperbolic reformulation and Active Flux method achieves 2nd-order accuracy for both primary variable u and the gradient of u . It is worth noting that, the magnitude of errors in Figure 6.1 is 10 times smaller than the errors in Figure 5.3. We record the total running time taken by each test case to reach the steady-state solutions in Table 6.1. All the tests were run on the

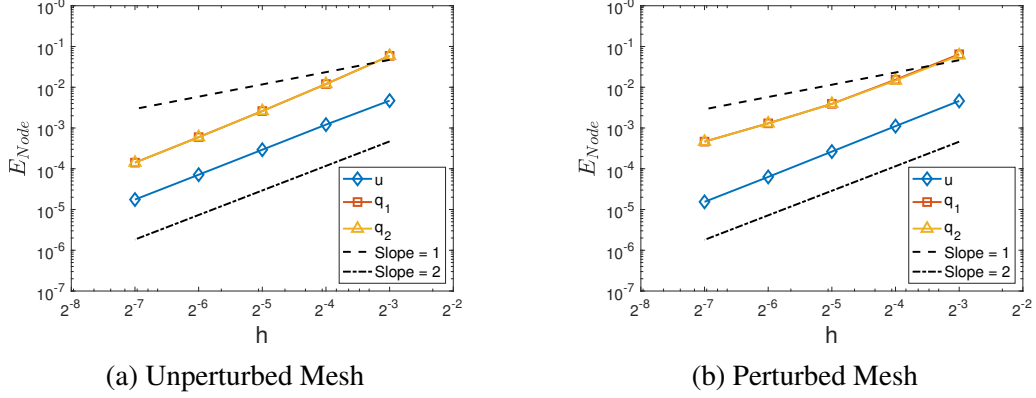


Figure 6.1: Convergence rates of the steady-state solutions to the hyperbolic heat equation on the unperturbed mesh and the perturbed mesh.

		$N = 8$	$N = 16$	$N = 32$	$N = 64$	$N = 128$
Unperturbed Mesh	Benchmark	0.173 s	2.593 s	39.58 s	636.1 s	10381.9 s
	Hyperbolic	0.512 s	2.999 s	18.23 s	124.7 s	902.5 s
Perturbed Mesh	Benchmark	0.169 s	2.665 s	65.54 s	1348 s	33774.4 s
	Hyperbolic	0.95 s	5.217 s	40.41 s	310.9 s	2534.4 s

Table 6.1: Comparison of the running time of the benchmark scheme and the one based on hyperbolic reformulation in solving the steady-state problem of the heat equation.

same machine with 1 CPU processor. We notice that the scheme based on hyperbolic equations gets much more accurate solutions within a much shorter time on fine meshes than the benchmark scheme that we suggest in Chapter 5.

6.2.2 Steady Viscous Flow around a Cylinder

We repeat the numerical experiments on the steady viscous flow around a cylinder which we simulate with the benchmark scheme in Section 5.2.3. We set $\tau = 0.1$. The initial values of relaxed variables are 0. Time step $\Delta t = \min\left(0.95 \frac{1}{10\text{MAX}_V} \min_{T,e} H_{T,e}, \tau\right)$. The sound speed is 1. Thus we estimate the upper bound of information propagation speed by $\text{MAX}_V = M_\infty + 1 + \sqrt{\frac{2\mu + \lambda}{\tau}} + \sqrt{\frac{\mu}{\tau}}$. M represents the total number of iterations. The initial weight of the diffusive corrector is $w = w_0 = 0.1$. When the change of all residuals after $M/10$ iterations is less than 5%, we reduce the weight w to $2 \times \min_{T,e} H_{T,e}$.

Figure 6.2 is generated on Mesh 3 (6738 nodes, 13264 cells) after $M = 32000$ iterations, which

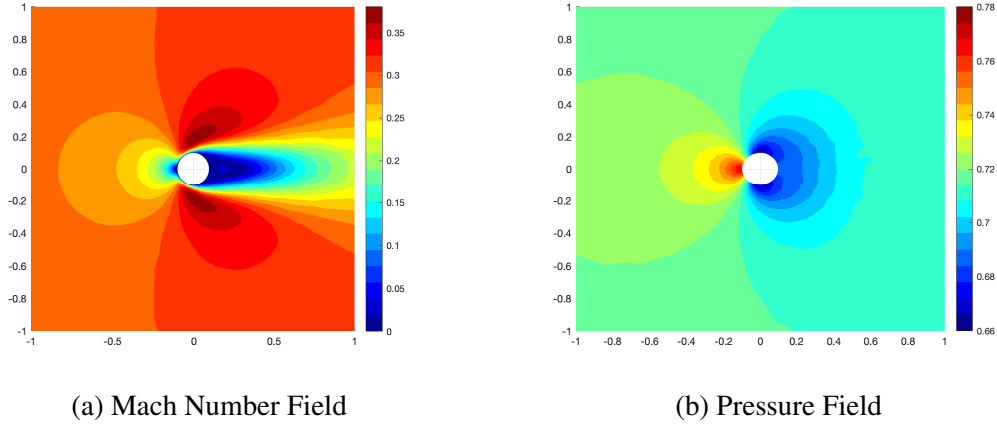
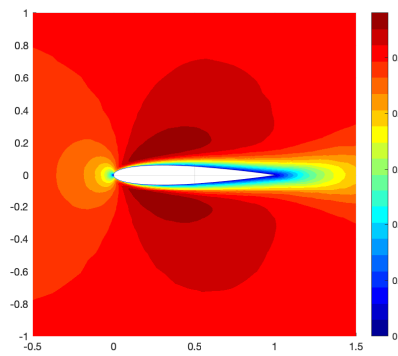


Figure 6.2: The steady-state solutions for the viscous flow around a cylinder on Mesh 3 generated by the Hyperbolic Navier-Stokes equations.

is qualitatively identical to Figure 5.4. The code was run on the same machine with 1 processor. The simulation of the hyperbolic model of the Navier-Stokes equations method only takes 4820.9 seconds while the traditional one simulated by the benchmark scheme in Chapter 5 takes 80623.2 seconds.

6.2.3 Steady Viscous Flow around the NACA0012 Airfoil

We further challenge the scheme based on the hyperbolic reformulation of the Navier-Stokes equations by the viscous flow around the NACA0012 Airfoil. The geometric setting remains the same as Section 4.3.3. We set $\tau = 0.1$. The simulation is started with uniform flow, where $M_\infty = 0.5$, $v_1 = M_\infty$, $v_2 = 0$, $p_\infty = 1/\gamma$, $\rho_\infty = 1$, $\gamma = 1.4$. The viscosity coefficients $\mu = 2.5 \times 10^{-4}$, $\lambda = -2\mu/3$. $Pr = 0.72$. We use the chord length $L = 1$ as the characteristic length. The corresponding Reynolds number is $Re = \frac{\rho_\infty M_\infty L}{\mu} = 2000$. The initial weight of the diffusive corrector is $w = w_0 = 0.1$. When the change of all residuals after $M/10$ iterations is less than 5%, we reduce the weight w to $4 \times \min_{T,e} H_{T,e}$. The numerical results on Mesh 3 (6738 nodes, 23488 cells) after $M = 12800$ iterations are presented in Figure 6.3. The Mach number field shows reasonable boundary layers. We cannot conduct a comparison with the benchmark scheme because it fails to generate any meaningful solutions in a reasonable time.



(a) Mach Number Field

Figure 6.3: The steady-state solutions for the viscous flow around the NACA0012 Airfoil on Mesh 3 generated by the Hyperbolic Navier-Stokes equations.

CHAPTER 7

Conclusions and Future Work

In this work, we extend the original 3rd-order AF scheme into a family of systematic and flexible schemes by introducing the idea of discrepancy distribution in the corrector. The numerical experiments indicate the schemes achieve $p + 1$ -th-order accuracy when polynomials of degree p are used in reconstruction. We also have successful experiments with schemes based on linear reconstruction for the inviscid and viscous flows on domains with complex geometry. For simulations of viscous flows, we suggest hyperbolic reformulation to accelerate convergence to steady-state solutions.

Various directions can be pursued from this work:

1. Currently we do not have high-order results for inviscid flows on domains with complex geometry due to the restriction from the accuracy of the boundary treatment. High-order treatment for curved boundaries could be pursued.
2. The solvers for the hyperbolic Navier-Stokes equations should be extended for higher-order accuracy. The current restrictions are mainly from the accuracy of boundary treatment and the predictor we used in the NS solver.
3. Our predictor for the hyperbolic Navier-Stokes equations is not as elaborated as the one for the Euler equations. Better use of the wave structure in the hyperbolic model could be utilized.
4. The code is yet to be developed for three-dimensional space.

APPENDIX A

The Governing Equations of Fluid Dynamics

We consider the equation for fluid dynamics in the most general form [24][41]. Let \mathbf{f} be the body force, \mathbf{T} be the stress tensor, and \mathbf{q} be the heat flux density vector. And e represents the local internal energy per unit mass. The general model of fluid is given by the following equations:

- Conservation of Mass:

$$\frac{D\rho}{Dt} + \rho\nabla \cdot \mathbf{v} = 0;$$

- Conservation of Linear Momentum:

$$\rho \frac{D\mathbf{v}}{Dt} = \rho\mathbf{f} + \nabla \cdot \mathbf{T};$$

- Conservation of Energy:

$$\rho \frac{D}{Dt} \left(e + \frac{1}{2} \mathbf{v} \cdot \mathbf{v} \right) = \rho(\mathbf{f} \cdot \mathbf{v}) - \nabla \cdot (\mathbf{q} - \mathbf{T} \cdot \mathbf{v}).$$

Note: $\frac{D}{Dt}$ is the material derivative, which is defined as $\frac{D}{Dt} = \frac{\partial}{\partial t} + (\mathbf{v} \cdot \nabla)$.

To make the equations above complete, we need constitutive equations as well. For viscosity, with the assumption of newtonian fluid,

$$\mathbf{T} = [-p + \lambda(\nabla \cdot \mathbf{v})]\mathbf{I} + 2\mu\mathbf{D},$$

where the deformation tensor/rate-of-strain tensor \mathbf{D} is defined as

$$\mathbf{D}_{ij} = \frac{1}{2} \left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} \right).$$

The equations for conservation of momentum can be simplified as:

$$\rho \frac{D\mathbf{v}}{Dt} = \rho \mathbf{f} + \nabla[-p + (\lambda + \mu)\nabla \cdot \mathbf{v}] + \mu \nabla^2 \mathbf{v}.$$

Note:

- Specially, for inviscid flow, $\mu = \lambda = 0$.
- Stokes' hypothesis: $\lambda + \frac{2}{3}\mu = 0$. (In three dimensional space.)

For ideal gas (or perfect gas), we have $p = \rho R\theta$, $e = c_v\theta + \text{constant}$, where θ is the local temperature, R is a constant for any particular gas, c_v is the local specific heat. It's not hard to derive that $p = A\rho^\gamma$, $e = \frac{p}{(\gamma-1)\rho}$ [10]. Here A depends on the entropy S , and $\gamma = (c_v + R)/c_v$. By Newton-Fourier law of cooling $\mathbf{q} = -\kappa\nabla\theta$.

In active flux method, in order to decompose Euler equation into advective part and acoustic part, we need the equations in primitive form with primitive variables: ρ , \mathbf{v} , p , as well as the conservative form. It's not hard to derive from the equation for energy balance that

$$\rho \frac{De}{Dt} = -\nabla \cdot \mathbf{q} + \mathbf{T} : \mathbf{D}$$

where

$$\mathbf{T} : \mathbf{D} = \sum_i \sum_j \mathbf{T}_{ij} \mathbf{D}_{ij}.$$

Replacing \mathbf{q} by $-\kappa\nabla\theta$

$$\rho \frac{De}{Dt} = \frac{\kappa}{c_v} \nabla \cdot (\nabla e) + \mathbf{T} : \mathbf{D}.$$

Since $\gamma = c_p/c_v$, Prandtl number $Pr = \frac{\mu c_p}{\kappa}$, we write $\frac{\kappa}{c_v}$ as $\frac{\mu\gamma}{Pr}$. Thus,

$$\frac{Dp}{Dt} = -\gamma p \nabla \cdot \mathbf{v} + \frac{\mu\gamma}{Pr} \nabla \cdot \left(\nabla \frac{p}{\rho} \right) + (\gamma - 1)\Phi.$$

The dissipation function Φ is defined as $\Phi = (\mathbf{T} + p\mathbf{I}) : \mathbf{D} = \lambda(\nabla \cdot \mathbf{v})^2 + 2\mu\mathbf{D} : \mathbf{D}$.

In summary, the Navier-Stokes equations for ideal gas are given by

$$\begin{cases} \frac{D\rho}{Dt} + \rho \nabla \cdot \mathbf{v} = 0 \\ \rho \frac{D\mathbf{v}}{Dt} + \nabla p = \nabla[(\lambda + \mu)\nabla \cdot \mathbf{v}] + \mu \nabla^2 \mathbf{v} \\ \frac{Dp}{Dt} + \gamma p \nabla \cdot \mathbf{v} = \frac{\mu\gamma}{Pr} \nabla \cdot \left(\nabla \frac{p}{\rho} \right) + (\gamma - 1)\Phi. \end{cases} \quad (\text{A.1})$$

APPENDIX B

Exact Solutions

B.1 The Pressureless Euler Equations

A concise derivation of the exact solutions to the Pressureless Euler equations by the author is as follows.

$$\begin{cases} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0, \\ \frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} = 0. \end{cases}$$

Note that the equations about velocity \mathbf{v} is actually independent of the equation for ρ . Along the characteristic curve $\mathbf{x}(s)$ governed by $\frac{d\mathbf{x}}{ds} = \mathbf{v}(\mathbf{x}(s), s)$,

$$\begin{cases} \frac{d\mathbf{v}}{ds} = \frac{d\mathbf{x}}{ds} \cdot \nabla \mathbf{v} + \frac{\partial \mathbf{v}}{\partial t} = 0, \\ \frac{d\rho}{ds} = \frac{d\mathbf{x}}{ds} \cdot \nabla \rho + \frac{\partial \rho}{\partial t} = -\rho \nabla \cdot \mathbf{v}. \end{cases} \quad (\text{B.1})$$

Like the Burgers' equation in the one-dimensional space, the velocity remains constant along the characteristics. The density ρ is solved by

$$\rho(s) = \frac{\rho(s_0)}{\exp\left(\int_{s_0}^s \nabla \cdot \mathbf{v} ds\right)}.$$

The denominator can be simplified further.

$$\exp\left(\int_{s_0}^s \nabla \cdot \mathbf{v} ds\right) = \exp\left(\int_{s_0}^s \mathbf{tr}(\nabla \mathbf{v}) ds\right) = \det \left| \exp\left(\int_{s_0}^s \nabla \mathbf{v} dx\right) \right|.^{1}$$

It is easy to verify that

$$\frac{d\nabla \mathbf{v}}{ds} = -(\nabla \mathbf{v})(\nabla \mathbf{v}).$$

Thus

$$\exp\left(\int_{s_0}^s \nabla \mathbf{v} dx\right) = \mathbf{I} + (s - s_0)\nabla \mathbf{v} \Big|_{s_0}.$$

B.2 The Hyperbolic Heat Equations

Recall the hyperbolic heat equations.

$$\begin{cases} \frac{\partial u}{\partial t} = \nu \nabla \cdot \mathbf{q}, \\ \frac{\partial \mathbf{q}}{\partial t} = \frac{1}{\tau}(\nabla u - \mathbf{q}). \end{cases} \quad (\text{B.2})$$

Let $v = e^{\frac{t}{2\tau}} u$, $\mathbf{p} = \sqrt{\tau \nu} e^{\frac{t}{2\tau}} \mathbf{q}$:

$$\begin{cases} \frac{\partial v}{\partial t} = \sqrt{\frac{\nu}{\tau}} \nabla \cdot \mathbf{p} + \frac{1}{2\tau} v \\ \frac{\partial \mathbf{p}}{\partial t} = \sqrt{\frac{\nu}{\tau}} \nabla v - \frac{1}{2\tau} \mathbf{p}. \end{cases} \quad (\text{B.3})$$

So without loss of generality, we consider the following hyperbolic equations.

$$\begin{cases} \frac{\partial v}{\partial t} = \nabla \cdot \mathbf{p} + cv \\ \frac{\partial \mathbf{p}}{\partial t} = \nabla v - c\mathbf{p}. \end{cases} \quad (\text{B.4})$$

Eliminating \mathbf{p} gives

$$\frac{\partial^2 v}{\partial t^2} = c^2 v + \Delta v. \quad (\text{B.5})$$

¹When \mathbf{A} is a square matrix, $\exp(\mathbf{A})$ is defined as $\sum_{k=0}^{\infty} \frac{\mathbf{A}^k}{k!}$.

Eliminating v gives

$$\frac{\partial^2 \mathbf{p}}{\partial t^2} = c^2 \mathbf{p} + \nabla(\nabla \cdot \mathbf{p}). \quad (\text{B.6})$$

Applying techniques introduced in 12.6 and 12.7 of [11], we know that for initial conditions $v(\mathbf{x}, 0) = 0$, $v_t(\mathbf{x}, 0) = \psi(\mathbf{x})$, in two-dimensional space v can be solved by

$$v(\mathbf{x}, t) = F_\psi(\mathbf{x}, t; c) = \int_0^t \frac{\rho}{\sqrt{t^2 - \rho^2}} \cosh c \sqrt{t^2 - \rho^2} Q_\psi(\mathbf{x}, \rho) d\rho, \quad (\text{B.7})$$

where,

$$Q(\mathbf{x}, \rho) = \frac{1}{2\pi} \int_0^{2\pi} \psi(x_1 + \rho \cos \theta, x_2 + \rho \sin \theta) d\theta.$$

It's straightforward to verify that $v^* = v_t$ solves the Eq. (B.5) with initial conditions $v^*(\mathbf{x}, 0) = \psi(\mathbf{x})$, $v_t^*(\mathbf{x}, 0) = 0$.

Thus for a general initial value problem $v(\mathbf{x}, 0) = \phi(\mathbf{x})$, $v_t(\mathbf{x}, 0) = \psi(\mathbf{x})$, the solution can be written as

$$v(\mathbf{x}, t) = F_\psi(\mathbf{x}, t; c) + \frac{\partial}{\partial t} F_\phi(\mathbf{x}, t; c). \quad (\text{B.8})$$

We notice that $\frac{\cosh c \sqrt{t^2 - \rho^2}}{\sqrt{t^2 - \rho^2}}$ approaches infinity as ρ approaches t . Integration by part twice gives

$$F_\psi(\mathbf{x}, t; c) = -\frac{1}{c} \sinh c \sqrt{t^2 - \rho^2} Q_\psi(\mathbf{x}, \rho) \Big|_0^t + \int_0^t \frac{1}{c} \sinh c \sqrt{t^2 - \rho^2} \frac{\partial Q_\psi(\mathbf{x}, \rho)}{\partial \rho} d\rho. \quad (\text{B.9})$$

Then we can derive the expression of the temporal derivative of F ,

$$\frac{\partial}{\partial t} F_\psi(\mathbf{x}, t; c) = \cosh ct Q_\psi(\mathbf{x}, 0) + \int_0^t \cosh c \sqrt{t^2 - \rho^2} \frac{t}{\sqrt{t^2 - \rho^2}} \frac{\partial Q_\psi(\mathbf{x}, \rho)}{\partial \rho} d\rho. \quad (\text{B.10})$$

We interpret $Q_\psi(\mathbf{x}, 0)$ as the limit of $Q_\psi(\mathbf{x}, \rho)$ as ρ approaches 0, that is,

$$Q_\psi(\mathbf{x}, 0) = \lim_{\rho \rightarrow 0} Q_\psi(\mathbf{x}, \rho) = \psi(\mathbf{x}).$$

However, it's difficult to figure out the analytical solution of \mathbf{p} directly from Eq. (B.6). We take

an alternative strategy here.

By Helmholtz's theorem, we can easily decompose \mathbf{p} into divergence-free part $\mathbf{p}^{(1)}$ and curl-free part $\mathbf{p}^{(2)}$, that is,

$$\mathbf{p} = \mathbf{p}^{(1)} + \mathbf{p}^{(2)},$$

where

$$\nabla \cdot \mathbf{p}^{(1)} = 0, \quad \nabla \times \mathbf{p}^{(2)} = 0.$$

It's not hard to see that v , $\mathbf{p}^{(1)}$, $\mathbf{p}^{(2)}$ satisfy the following equations

$$\begin{cases} \frac{\partial v}{\partial t} = \nabla \cdot \mathbf{p}^{(2)} + cv \\ \frac{\partial \mathbf{p}^{(2)}}{\partial t} = \nabla v - c\mathbf{p}^{(2)} \\ \frac{\partial \mathbf{p}^{(1)}}{\partial t} = -c\mathbf{p}^{(1)}. \end{cases} \quad (\text{B.11})$$

It's worth noting that only the curl-free part $\mathbf{p}^{(2)}$ interact with v . The divergence-free part $\mathbf{p}^{(1)}$ just decays exponentially on its own. Inspired by an identity in three-dimensional space $\nabla(\nabla \cdot \mathbf{q}) = \Delta \mathbf{q} + \nabla \times (\nabla \times \mathbf{q})$, we notice that

$$\nabla(\nabla \cdot \mathbf{p}^{(2)}) = \Delta \mathbf{p}^{(2)}.$$

Thus,

$$\frac{\partial^2 \mathbf{p}^{(2)}}{\partial t^2} = c^2 \mathbf{p}^{(2)} + \Delta \mathbf{p}^{(2)}. \quad (\text{B.12})$$

Then we have a solution:

$$\mathbf{p}(\mathbf{x}, t) = F_{\frac{\partial}{\partial t} \mathbf{p}^{(2)}(\cdot, 0)}(\mathbf{x}, t; c) + \frac{\partial}{\partial t} F_{\mathbf{p}^{(2)}(\cdot, 0)}(\mathbf{x}, t; c) + e^{-ct} \mathbf{p}^{(1)}(\mathbf{x}, 0).$$

With the fact $\frac{\partial}{\partial t}\mathbf{p}^{(1)}(\mathbf{x}, 0) = -c\mathbf{p}^{(1)}(\mathbf{x}, 0)$, we have

$$F_{\frac{\partial}{\partial t}\mathbf{p}^{(1)}(\cdot, 0)} + \frac{\partial}{\partial t}F_{\mathbf{p}^{(1)}(\cdot, 0)} = e^{-ct}\mathbf{p}^{(1)}(\mathbf{x}, 0) + \int_0^t \left(\frac{t}{\sqrt{t^2 - \rho^2}} \cosh c\sqrt{t^2 - \rho^2} - \sinh c\sqrt{t^2 - \rho^2} \right) \frac{Q_{\mathbf{p}^{(1)}(\cdot, 0)}(\mathbf{x}, \rho)}{\partial \rho} d\rho.$$

Since

$$\begin{aligned} \frac{\partial Q_{\mathbf{p}^{(1)}(\cdot, 0)}(\mathbf{x}, \rho)}{\partial \rho} &= \frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \Delta \mathbf{p}^{(1)}(\mathbf{x} + (r_1, r_2), 0) dr_1 dr_2 \\ &= -\frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{p}^{(1)}(\mathbf{x} + (r_1, r_2), 0)) - \Delta \mathbf{p}^{(1)}(\mathbf{x} + (r_1, r_2), 0) dr_1 dr_2 \\ &= -\frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{p}(\mathbf{x} + (r_1, r_2), 0)) - \Delta \mathbf{p}(\mathbf{x} + (r_1, r_2), 0) dr_1 dr_2 \\ &= -\frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{p}(\mathbf{x} + (r_1, r_2), 0)) dr_1 dr_2 + \frac{\partial Q_{\mathbf{p}(\cdot, 0)}(\mathbf{x}, \rho)}{\partial \rho}, \end{aligned}$$

we conclude that

$$\begin{aligned} \mathbf{p}(\mathbf{x}, t) &= F_{\frac{\partial}{\partial t}\mathbf{p}(\cdot, 0)} + \frac{\partial}{\partial t}F_{\mathbf{p}(\cdot, 0)} \\ &+ \int_0^t \left(\left(\frac{t}{\sqrt{t^2 - \rho^2}} \cosh c\sqrt{t^2 - \rho^2} - \sinh c\sqrt{t^2 - \rho^2} \right) \right. \\ &\left. \left(\frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{p}(\mathbf{x} + (r_1, r_2), 0)) dr_1 dr_2 - \frac{\partial Q_{\mathbf{p}(\cdot, 0)}(\mathbf{x}, \rho)}{\partial \rho} \right) \right) d\rho. \end{aligned} \quad (\text{B.13})$$

To avoid taking high order derivatives in numerical schemes, we find that

$$\frac{1}{2\pi\rho} \iint_{r^2 \leq \rho^2} \nabla(\nabla \cdot \mathbf{p}(\mathbf{x} + (r_1, r_2), 0)) dr_1 dr_2 = \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\nabla \cdot \mathbf{p}) \Big|_{\mathbf{x} + \rho(\cos \theta, \sin \theta)} d\theta,$$

where $\mathbf{n} = (\cos \theta, \sin \theta)$.

In summary, combining Eq. (B.8) and Eq. (B.13), we get the following results after proper simplification. Given data of v and \mathbf{p} at time $t = t_0$, we have the exact expressions of v and \mathbf{p} at time $t = t_0 + \Delta t$:

$$v(\mathbf{x}, t_0 + \Delta t) = \frac{\partial}{\partial t} F_{v(\cdot, t_0)}(\mathbf{x}, \Delta t; c) + F_{cv(\cdot, t_0)}(\mathbf{x}, \Delta t; c) + F_{\nabla \cdot \mathbf{p}(\cdot, t_0)}(\mathbf{x}, \Delta t; c),$$

$$\begin{aligned} \mathbf{p}(\mathbf{x}, t_0 + \Delta t) &= e^{-c\Delta t} \mathbf{Q}_{\mathbf{p}(\cdot, t_0)}(\mathbf{x}, 0) + F_{\nabla v(\cdot, t_0)}(\mathbf{x}, \Delta t; c) \\ &+ \int_0^{\Delta t} \left(\frac{\Delta t}{\sqrt{(\Delta t)^2 - \rho^2}} \cosh c \sqrt{(\Delta t)^2 - \rho^2} - \sinh c \sqrt{(\Delta t)^2 - \rho^2} \right) \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\nabla \cdot \mathbf{p}) d\theta d\rho. \end{aligned} \quad (\text{B.14})$$

Thus with initial conditions of u and \mathbf{q} at time $t = t_0$, the exact solutions to the hyperbolic heat equations Eq. (C.1) at time $t = t_0 + \Delta t$ are given by the following formulas. Let $c = \frac{1}{2\tau}$, $a = \sqrt{\frac{v}{\tau}}$.

$$\begin{aligned} u(\mathbf{x}, t_0 + \Delta t) &= e^{-c\Delta t} \left(\frac{1}{a} \frac{\partial}{\partial t} F_{u(\cdot, t_0)}(\mathbf{x}, a\Delta t; \frac{c}{a}) + F_{\frac{\varepsilon}{a} u(\cdot, t_0)}(\mathbf{x}, a\Delta t; \frac{c}{a}) + F_{\sqrt{v\tau} \nabla \cdot \mathbf{q}(\cdot, t_0)}(\mathbf{x}, a\Delta t; \frac{c}{a}) \right), \\ \mathbf{q}(\mathbf{x}, t_0 + \Delta t) &= \frac{e^{-c\Delta t}}{\sqrt{v\tau}} \left[e^{-c\Delta t} \mathbf{Q}_{\sqrt{v\tau} \mathbf{q}(\cdot, t_0)}(\mathbf{x}, 0) + F_{\nabla u(\cdot, t_0)}(\mathbf{x}, a\Delta t; \frac{c}{a}) \right. \\ &+ \int_0^{a\Delta t} \left(\frac{a\Delta t}{\sqrt{(a\Delta t)^2 - \rho^2}} \cosh \frac{c}{a} \sqrt{(a\Delta t)^2 - \rho^2} - \sinh \frac{c}{a} \sqrt{(a\Delta t)^2 - \rho^2} \right) \\ &\left. \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\sqrt{v\tau} \nabla \cdot \mathbf{q}) d\theta \right] d\rho. \end{aligned} \quad (\text{B.15})$$

This also indicates that the physical domain of dependence of u and \mathbf{q} at $(\mathbf{x}, t_0 + \Delta t)$ is a disk centered at \mathbf{x} with radius $a\Delta t$ at $t = t_0$.

Note: The derivation of the exact solutions to the hyperbolic heat equations was originally published by the author in [18].

B.3 The Hyperbolic Formulation of Viscosity

Recall the equations for hyperbolic viscosity Eq. (6.10) and the systems that govern \mathbf{v}_1 and \mathbf{v}_2 ,

where $\mathbf{v}_1 + \mathbf{v}_2 = \mathbf{v}$, $\nabla \cdot \mathbf{v}_1 = 0$, $\nabla \times \mathbf{v}_2 = 0$,

$$\begin{cases} \rho \frac{\partial \mathbf{v}}{\partial t} = -\mu \nabla \times \omega + (2\mu + \lambda) \nabla \psi, \\ \frac{\partial \omega}{\partial t} = \frac{1}{\tau_1} (\nabla \times \mathbf{v} - \omega), \\ \frac{\partial \psi}{\partial t} = \frac{1}{\tau_2} (\nabla \cdot \mathbf{v} - \psi). \end{cases} \quad (\text{B.16})$$

$$\begin{cases} \rho \frac{\partial \mathbf{v}_1}{\partial t} = -\mu \nabla \times \omega, \\ \frac{\partial \omega}{\partial t} = \frac{1}{\tau_1} (\nabla \times \mathbf{v}_1 - \omega). \end{cases} \quad (\text{B.17})$$

$$\begin{cases} \rho \frac{\partial \mathbf{v}_2}{\partial t} = (2\mu + \lambda) \nabla \psi, \\ \frac{\partial \psi}{\partial t} = \frac{1}{\tau_2} (\nabla \cdot \mathbf{v}_2 - \psi). \end{cases} \quad (\text{B.18})$$

It's not hard to derive the following independent second-order governing equations for \mathbf{v}_1 , ω , \mathbf{v}_2 , ψ .

$$\rho \frac{\partial^2 \mathbf{v}_1}{\partial t^2} + \frac{\rho}{\tau_1} \frac{\partial \mathbf{v}_1}{\partial t} = \frac{\mu}{\tau_1} \Delta \mathbf{v}_1; \quad (\text{B.19})$$

$$\rho \frac{\partial^2 \omega}{\partial t^2} + \frac{\rho}{\tau_1} \frac{\partial \omega}{\partial t} = \frac{\mu}{\tau_1} \Delta \omega; \quad (\text{B.20})$$

$$\rho \frac{\partial^2 \mathbf{v}_2}{\partial t^2} + \frac{\rho}{\tau_2} \frac{\partial \mathbf{v}_2}{\partial t} = \frac{2\mu + \lambda}{\tau_2} \Delta \mathbf{v}_2; \quad (\text{B.21})$$

$$\rho \frac{\partial^2 \psi}{\partial t^2} + \frac{\rho}{\tau_2} \frac{\partial \psi}{\partial t} = \frac{2\mu + \lambda}{\tau_2} \Delta \psi. \quad (\text{B.22})$$

Recall the solution to $v_{tt} = a^2 \Delta v + c^2 u$ is given by

$$v(\mathbf{x}, t) = \frac{1}{a} F_{v_t(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{a} \frac{\partial}{\partial t} F_{v(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}). \quad (\text{B.23})$$

(Note: The chain rule should be used in the second term. The definition of F is the same as the one in Section B.2.)

Let $c = \frac{1}{2\tau_1}$, $a = \sqrt{\frac{\mu}{\rho\tau_1}}$. With proper transformation, it's not hard to see that

$$\mathbf{v}_1(\mathbf{x}, t) = e^{-ct} \left(\frac{-\mu}{a\rho} F_{\nabla \times \omega(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{2a\tau_1} F_{\mathbf{v}_1(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{a} \frac{\partial}{\partial t} F_{\mathbf{v}_1(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) \right). \quad (\text{B.24})$$

$$\omega(\mathbf{x}, t) = e^{-ct} \left(\frac{1}{a\tau_1} F_{\nabla \times \mathbf{v}_1(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) - \frac{1}{2a\tau_1} F_{\omega(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{a} \frac{\partial}{\partial t} F_{\omega(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) \right). \quad (\text{B.25})$$

Similarly we can derive the formulas for \mathbf{v}_2 and ψ .

Let $c = \frac{1}{2\tau_2}$, $a = \sqrt{\frac{\lambda+2\mu}{\rho\tau_2}}$. With proper transformation, it's not hard to see that

$$\mathbf{v}_2(\mathbf{x}, t) = e^{-ct} \left(\frac{2\mu + \lambda}{a\rho} F_{\nabla \psi(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{2a\tau_2} F_{\mathbf{v}_2(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{a} \frac{\partial}{\partial t} F_{\mathbf{v}_2(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) \right). \quad (\text{B.26})$$

$$\psi(\mathbf{x}, t) = e^{-ct} \left(\frac{1}{a\tau_2} F_{\nabla \cdot \mathbf{v}_2(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) - \frac{1}{2a\tau_2} F_{\psi(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) + \frac{1}{a} \frac{\partial}{\partial t} F_{\psi(\cdot, 0)}(\mathbf{x}, at; \frac{c}{a}) \right). \quad (\text{B.27})$$

By properly combining Eq. (B.26) and Eq. (B.24), we have

$$\begin{aligned} \mathbf{v}(\mathbf{x}, t) &= e^{-c_1 t} \left(\frac{-\mu}{a_1 \rho} F_{\nabla \times \omega(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{2a_1 \tau_1} F_{\mathbf{v}(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{a_1} \frac{\partial}{\partial t} F_{\mathbf{v}(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) \right) \\ &\quad - e^{-c_1 t} \left(\frac{1}{2a_1 \tau_1} F_{\mathbf{v}_2(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{a_1} \frac{\partial}{\partial t} F_{\mathbf{v}_2(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) \right) + \mathbf{v}_2(\mathbf{x}, 0) \\ &\quad + e^{-c_2 t} \left(\frac{2\mu + \lambda}{a_2 \rho} F_{\nabla \psi(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) + \frac{1}{2a_2 \tau_2} F_{\mathbf{v}(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) + \frac{1}{a_2} \frac{\partial}{\partial t} F_{\mathbf{v}(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) \right) \\ &\quad - e^{-c_2 t} \left(\frac{1}{2a_2 \tau_2} F_{\mathbf{v}_1(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) + \frac{1}{a_2} \frac{\partial}{\partial t} F_{\mathbf{v}_1(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) \right) + \mathbf{v}_1(\mathbf{x}, 0) \\ &\quad - \mathbf{v}_1(\mathbf{x}, 0) - \mathbf{v}_2(\mathbf{x}, 0) \end{aligned}$$

Luckily, like the case in the hyperbolic heat equations we can simplify some terms further.

Since $\nabla \times \mathbf{v}_2 = 0$, we have $\Delta \mathbf{v}_2 = \nabla(\nabla \cdot \mathbf{v}_2) = \nabla(\nabla \cdot \mathbf{v})$. Then,

$$\begin{aligned}
& -e^{-c_1 t} \left(\frac{1}{2a_1 \tau_1} F_{\mathbf{v}_2(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{a_1} \frac{\partial}{\partial t} F_{\mathbf{v}_2(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) \right) + \mathbf{v}_2(\mathbf{x}, 0) \\
& - e^{-c_1 t} \int_0^{a_1 t} \left(\frac{a_1 t}{\sqrt{(a_1 t)^2 - \rho^2}} \cosh \frac{c_1}{a_1} \sqrt{(a_1 t)^2 - \rho^2} + \sinh \frac{c_1}{a_1} \sqrt{(a_1 t)^2 - \rho^2} \right) \frac{\partial Q_{\mathbf{v}_2}}{\partial \rho} d\rho \\
& = -e^{-c_1 t} \int_0^{a_1 t} \left(\frac{a_1 t}{\sqrt{(a_1 t)^2 - \rho^2}} \cosh \frac{c_1}{a_1} \sqrt{(a_1 t)^2 - \rho^2} + \sinh \frac{c_1}{a_1} \sqrt{(a_1 t)^2 - \rho^2} \right) \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\sqrt{v\tau} \nabla \cdot \mathbf{v}) d\theta d\rho
\end{aligned}$$

Similarly, with $\nabla \cdot \mathbf{v}_1 = 0$, we have $\Delta \mathbf{v}_1 = \Delta \mathbf{v}_1 - \nabla(\nabla \cdot \mathbf{v}_1) = \Delta \mathbf{v} - \nabla(\nabla \cdot \mathbf{v})$. Then,

$$\begin{aligned}
& -e^{-c_2 t} \left(\frac{1}{2a_2 \tau_2} F_{\mathbf{v}_1(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) + \frac{1}{a_2} \frac{\partial}{\partial t} F_{\mathbf{v}_1(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) \right) + \mathbf{v}_1(\mathbf{x}, 0) \\
& - e^{-c_2 t} \int_0^{a_2 t} \left(\frac{a_2 t}{\sqrt{(a_2 t)^2 - \rho^2}} \cosh \frac{c_2}{a_2} \sqrt{(a_2 t)^2 - \rho^2} + \sinh \frac{c_2}{a_2} \sqrt{(a_2 t)^2 - \rho^2} \right) \frac{\partial Q_{\mathbf{v}_1}}{\partial \rho} d\rho \\
& = -e^{-c_2 t} \int_0^{a_2 t} \left(\frac{a_2 t}{\sqrt{(a_2 t)^2 - \rho^2}} \cosh \frac{c_2}{a_2} \sqrt{(a_2 t)^2 - \rho^2} + \sinh \frac{c_2}{a_2} \sqrt{(a_2 t)^2 - \rho^2} \right) \\
& \quad \left(\frac{\partial Q_{\mathbf{v}}}{\partial \rho} - \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\sqrt{v\tau} \nabla \cdot \mathbf{v}) d\theta \right) d\rho
\end{aligned}$$

In summary, we have

$$\begin{aligned}
\mathbf{v}(\mathbf{x}, t) &= e^{-c_1 t} \left(\frac{-\mu}{a_1 \rho} F_{\nabla \times \omega(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{2a_1 \tau_1} F_{\mathbf{v}(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{a_1} \frac{\partial}{\partial t} F_{\mathbf{v}(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) \right) \\
& - e^{-c_1 t} \int_0^{a_1 t} \left(\frac{a_1 t}{\sqrt{(a_1 t)^2 - \rho^2}} \cosh \frac{c_1}{a_1} \sqrt{(a_1 t)^2 - \rho^2} + \sinh \frac{c_1}{a_1} \sqrt{(a_1 t)^2 - \rho^2} \right) \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\sqrt{v\tau} \nabla \cdot \mathbf{v}) d\theta d\rho \\
& + e^{-c_2 t} \left(\frac{2\mu + \lambda}{a_2 \rho} F_{\nabla \psi(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) \right) \\
& + \int_0^{a_2 t} \left(\frac{a_2 t}{\sqrt{(a_2 t)^2 - \rho^2}} \cosh \frac{c_2}{a_2} \sqrt{(a_2 t)^2 - \rho^2} + \sinh \frac{c_2}{a_2} \sqrt{(a_2 t)^2 - \rho^2} \right) \frac{1}{2\pi} \int_0^{2\pi} \mathbf{n}(\sqrt{v\tau} \nabla \cdot \mathbf{v}) d\theta d\rho
\end{aligned} \tag{B.28}$$

$$\omega(\mathbf{x}, t) = e^{-c_1 t} \left(\frac{1}{a_1 \tau_1} F_{\nabla \times \mathbf{v}(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) - \frac{1}{2a_1 \tau_1} F_{\omega(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) + \frac{1}{a_1} \frac{\partial}{\partial t} F_{\omega(\cdot, 0)}(\mathbf{x}, a_1 t; \frac{c_1}{a_1}) \right). \tag{B.29}$$

$$\psi(\mathbf{x}, t) = e^{-c_2 t} \left(\frac{1}{a_2 \tau_2} F_{\nabla \cdot \mathbf{v}(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) - \frac{1}{2a_2 \tau_2} F_{\psi(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) + \frac{1}{a_2} \frac{\partial}{\partial t} F_{\psi(\cdot, 0)}(\mathbf{x}, a_2 t; \frac{c_2}{a_2}) \right). \tag{B.30}$$

APPENDIX C

Elementary Analysis of the Hyperbolic Heat Equations in the One-dimensional Space

To get some hints about designing the best hyperbolic formulation of the Navier-Stokes equation, we take the simplest hyperbolic reformulation of the heat equation in one-dimensional space as a model problem.

The hyperbolic heat-conduction equation, which was first proposed by C. Cattaneo[8].

$$\begin{cases} \frac{\partial u}{\partial t} = \nu \frac{\partial q}{\partial x} \\ \frac{\partial q}{\partial t} = \frac{1}{\tau} \left(\frac{\partial u}{\partial x} - q \right), \end{cases} \quad (\text{C.1})$$

where ν is a positive diffusion coefficient, and τ is the relaxation time, which is introduced to eliminate the paradox that information travels at infinite speed. Some analysis about the characteristics and dispersive waves of Eq. (C.1) has been done in [38]. There are also numerical schemes for it based on residual-distribution [29] and Active Flux [33] for Eq. (C.1).

It's not hard to see that the steady state of Eq. (C.1) is equivalent to that of the traditional heat equation:

$$\frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2}. \quad (\text{C.2})$$

The hyperbolic nature of Eq. (C.1) allows time step of $O(h)$ in numerical simulation, while explicit schemes for the heat equation Eq. (C.2) require time step no greater than $O(h^2)$ for stability concerns. If the steady solution is sought through solving the unsteady problem until it converges,

this property may enable a fast algorithm. Besides, introducing more variables in the hyperbolic formulation of the heat equation enables the gradient u_x to have the same order of accuracy as the main variable u in steady states.

Here we take the one-dimensional equation as an example to illustrate how the hyperbolic reformulation reduces stiffness of the heat equation.

Eliminating q in Eq. (C.1) gives

$$\tau \frac{\partial^2 u}{\partial t^2} + \frac{\partial u}{\partial t} - \nu \frac{\partial^2 u}{\partial x^2} = 0, \quad (\text{C.3})$$

which resembles the telegrapher's equation. When τ is small, we interpret Eq. (C.3) as a perturbed heat equation.

Consider an initial value problem of the heat equation on $[0, 1]$.

$$\begin{cases} \frac{\partial u}{\partial t} = \nu \frac{\partial^2 u}{\partial x^2}, & x \in [0, 1]; \\ u(0, t) = 0; & u(1, t) = 0; \\ u(x, 0) = f(x). \end{cases} \quad (\text{C.4})$$

The solution is written in the form:

$$u(x, t) = \sum c_k e^{-\nu k^2 \pi^2 t} \sin(k\pi x).$$

And we see that the components with higher frequency decay faster.

Then the corresponding hyperbolic reformulation of this problem is given by

$$\begin{cases} \tau \frac{\partial^2 u}{\partial t^2} + \frac{\partial u}{\partial t} - \nu \frac{\partial^2 u}{\partial x^2} = 0, & x \in [0, 1]; \\ u(0, t) = 0, & u(1, t) = 0; \\ u(x, 0) = f(x), & u_t(x, 0) = 0. \end{cases} \quad (\text{C.5})$$

The solution is in the form:

$$u(x, t) = \sum (c_k^{(1)} e^{a_k^{(1)} t} + c_k^{(2)} e^{a_k^{(2)} t}) \sin(k\pi x). \quad (\text{C.6})$$

$a_k^{(1)}, a_k^{(2)}$ satisfy the following equation:

$$\tau a_k^2 + a_k + \nu k^2 \pi^2 = 0.$$

So, for k that satisfies $4\tau\nu k^2 \pi^2 \leq 1$,

$$a_k^{(1)} = \frac{-1 + \sqrt{1 - 4\tau\nu k^2 \pi^2}}{2\tau}, \quad a_k^{(2)} = \frac{-1 - \sqrt{1 - 4\tau\nu k^2 \pi^2}}{2\tau};$$

for k that satisfies $4\tau\nu k^2 \pi^2 > 1$,

$$a_k^{(1)} = \frac{-1 + i\sqrt{4\tau\nu k^2 \pi^2 - 1}}{2\tau}, \quad a_k^{(2)} = \frac{-1 - i\sqrt{4\tau\nu k^2 \pi^2 - 1}}{2\tau}.$$

We notice that, for fixed k , as τ approaches 0,

$$a_k^{(1)} \approx -\nu k^2 \pi^2, \quad a_k^{(2)} \approx -\frac{1}{\tau};$$

and $c_k^{(1)} \rightarrow c_k, c_k^{(2)} \rightarrow 0$, which means that for given t , the solution approaches the solution of Eq. (C.5). Besides, the eigenvalues of Eq. (C.5) vary from $-\nu\pi^2$ to $-\frac{1}{2\tau}$ and thus the system is very stiff.

When τ is large, or more specifically, $\tau \geq \frac{1}{4\nu\pi^2}$, the amplitudes of all wave components in the initial condition oscillate with different temporal frequencies but damp at the same rate $-\frac{1}{2\tau}$. Thus it is much less stiff.

In other words, the hyperbolic model of heat conduction transforms the damping of the solution caused by diffusion into decay caused by source terms. Although source terms might be stiff in general, it can be fixed by point-implicit schemes with a large time step size of $O(h)$. For example,

a scheme based on characteristics and Active Flux method was introduced by Nishikawa in [33].

We study the wave structure of the system by omitting the source terms.

$$\begin{cases} \frac{\partial u}{\partial t} = v \frac{\partial q}{\partial x} \\ \frac{\partial q}{\partial t} = \frac{1}{\tau} \left(\frac{\partial u}{\partial x} \right) \end{cases} \quad (\text{C.7})$$

The characteristic speeds are $\pm \sqrt{\frac{v}{\tau}}$. We can set time step $\Delta t = \frac{1}{2} h \sqrt{\frac{\tau}{v}}$. To reach a solution where $\|u\|_{\infty} < \varepsilon$, it takes around $2\tau \ln(\varepsilon)/\Delta t = 4 \ln(\varepsilon) \sqrt{v\tau}/h$ steps. To minimize the total number of iterations, we can pick the smallest τ that make Eq. (C.1) not stiff, e.g., $\tau = \frac{1}{4v\pi^2}$ for Eq. (C.5).

The simple analysis above matches the conclusions drawn from experiments in [29], [38] well.

APPENDIX D

Notes about My Code

D.1 Mesh Refinement

To construct hierarchical meshes for the mesh convergence study of our schemes, we refine the mesh by splitting cells into 4 sub-cells of similar size. For interior cells, this is straightforward: we introduce the midpoints of edges as additional nodes and connect them to form 4 congruent sub-triangles in each cell (Figure D.1). If the cell is on the boundary, instead of directly using the midpoint of the edge that approximates the boundary, we push that node to the boundary with an iterative method (Algorithm D.1) and construct 4 new cells (Figure D.2).

Algorithm D.1 The algorithm for pushing a node to the boudnary.

\mathbf{x}^0 = Coordinates of the midpoint of an edge e that approximates the boundary;

$n = 0$;

$\varepsilon = 10^{-14}$;

$dx = 10^{-8}$;

Step = $\frac{1}{50}$ length of edge e ;

while $f(\mathbf{x}^n) > \varepsilon$ **do** ▶ $f(\mathbf{x})$ is a differentiable function that reflects the distance from \mathbf{x} to the boundary. f and $\frac{\partial f}{\partial \mathbf{n}}$ vanish on the boundary.

$f_x = (f(\mathbf{x}^n + (dx, 0)) - f(\mathbf{x}^n))/dx$;

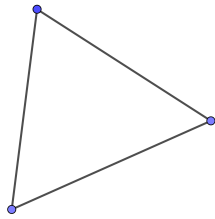
$f_y = (f(\mathbf{x}^n + (0, dx)) - f(\mathbf{x}^n))/dx$;

$\mathbf{x}^{n+1} = \mathbf{x}^n + \text{Step} \times (f_x, f_y)$;

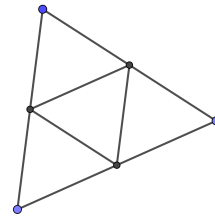
$n = n + 1$;

end while

return \mathbf{x}^n ;

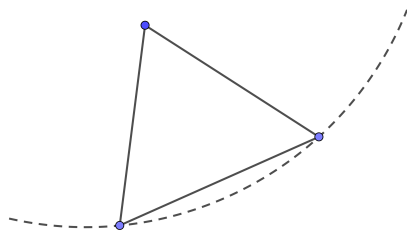


(a) The original cell

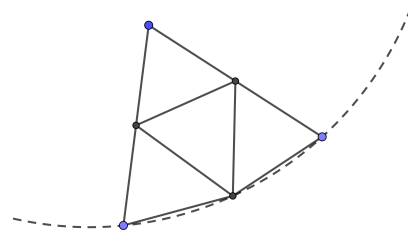


(b) The refined cell

Figure D.1: Refinement of an interior cell.



(a) The original cell



(b) The refined cell

Figure D.2: Refinement of a cell on the boundary.

APPENDIX E

Archive of Miscellaneous Investigation

E.1 A Primitive Limiter for One-dimensional Space

Generally, almost all high-order numerical schemes rely on the smoothness of the exact solution. Our analysis above, which assumes the accuracy of predictor, is not an exception, either. Unfortunately, almost all polynomial approximation for a function with derivatives of a large magnitude not only is inaccurate but also contain spurious oscillation, which may result in instability of numerical schemes. Thus limiters are designed to restrain the spurious oscillation [23].

We notice that the original third-order AF schemes experienced instability when shock wave forms, no matter in the simulation of the Burgers equation or Euler equations. Similar failure of the original AF Euler solver are also reported by other researcher [20]. For cells near the discontinuity, even the reconstruction from the exact nodal values contains large absolute error, which make the predictor lose accuracy. Consequently, the discrepancy evaluated in the corrector step is no longer one-order smaller, because the error in fluxes is not canceled near the discontinuity. The bubble function method deteriorates the issues further. It does not allow correction on the predicted nodal values at the interfaces. Absorbing all information from the conservation law by a bubble function frequently overshoots and creates more spurious oscillation for the next step.

On the other hand, the large discrepancy also gives the corrector more leverage on the numerical solutions if we distribute it properly. Here we propose a simple but effective limiter to increase the robustness of the AF schemes. The basic idea is not to significantly increase the total variation

after discrepancy distribution. Note that absorbing the discrepancy into the nodal values at the interfaces will change the slope at the center of the cell by

$$\frac{U_{k+1/2}^{n+1} - U_{k-1/2}^{n+1}}{\Delta x} - \frac{U_{k+1/2}^* - U_{k-1/2}^*}{\Delta x} = \frac{\beta_2 d\bar{U}_k + \beta_1 d\bar{U}_{k+1} - \beta_2 d\bar{U}_{k-1} - \beta_1 d\bar{U}_k}{2\Delta x}.$$

For $p = 1$, where continuous piecewise linear functions are reconstructed, a simple strategy is to distribute more to the left when the discrepancy and the slope at the center are of the same signs, to the right when different. In other words, if $d\bar{U}_k(U_{k+1/2} - U_{k-1/2}) > 0$, we set $\tilde{\beta}_1 = \beta_1 - q\beta_1$, $\tilde{\beta}_2 = \beta_2 + q\beta_1$. Otherwise, $\tilde{\beta}_1 = \beta_1 + q\beta_1$, $\tilde{\beta}_2 = \beta_2 - q\beta_1$.

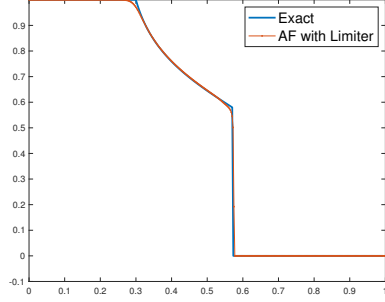
For $p = 2$, we treat it as linear problems on sub-intervals $[x_{k-1/2}, x_k]$, $[x_k, x_{k+1/2}]$. First we decide the weights on left and right sub-cells w_1 and w_2 in the similar way, i.e., when $d\bar{U}_k(U_{k+1/2} - U_{k-1/2}) > 0$, $\tilde{w}_1 = w_1 - q_0 w_1$, $\tilde{w}_2 = w_2 + q_0 w_1$. Then on each sub-cell, distribute the discrepancy to the nodes at the endpoints by the same rule with parameter q_1 .

The above process actually introduces some artificial dissipation proportional to the discrepancy. It can be further combined with conditions on the flow directions. The parameter q can also be chosen flexibly. It is worth noting that when the solution is smooth, the simple limiter does not reduce the order of accuracy. Thus the typical issue that limiters go overboard does not exist here. The setting of the simple limiter is extendable to higher-order accuracy or multidimensional space. We will further refine the idea in future work. To illustrate the merits of the simpler limiter, we challenge the AF schemes with Riemann problems on the Buckley-Leverett equation and the Euler equations.

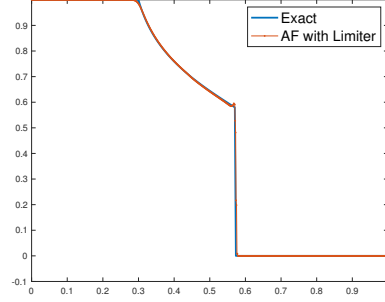
E.1.1 The Buckley-Leverett Equation

The Buckley-Leverett equation is used to model two-phase flow in a porous medium.

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \quad \text{where } f(u) = \frac{u^2}{u^2 + 0.5(1-u)^2}. \quad (\text{E.1})$$



(a) $p = 1, \beta_1 = \beta_2 = 1, \Delta x = 1/400$.



(b) $p = 2, \alpha = \beta_1 = \beta_2 = 1, \Delta x = 1/400$.

Figure E.1: Numerical solutions to the Buckley-Leverett equations generated by the Active Flux schemes with simple limiters.

Note that the flux function $f(u)$ is neither convex or concave over $[0, 1]$. Thus for an initial condition with discontinuity,

$$\begin{cases} u(x, 0) = 1, & 0 \leq x < 0.3 \\ u(x, 0) = 0, & 0.3 \leq x \leq 1, \end{cases}$$

the solution contains both rarefied wave and shock wave. We need to introduce sufficient numerical dissipation to catch the entropy solution but not to smear the solution too much. We set all the initial discrepancy distribution weights equal 1. For $p = 1$, we adjust the weights with parameter $q = 1$; For $p = 2$, we set the parameter $q_0 = 0.5, q_1 = 0.25$. To avoid excessive dissipation, we only adjust the weights when the flow direction is consistent with the adjustment. In the following test cases, we divide the domain into 400 cells and set $\Delta t = \Delta x/p/2.5$. In Figure E.1, the numerical solutions are compared with the exact solution found by the equal area rule [23] at $T = 0.2$.

E.1.2 The Euler Equations

We test the AF schemes by two Riemann problems of the Euler equations. The modified Sod's test aims to assess the entropy property of the schemes; Toro's Test 3 serves as an advanced challenge of robustness and accuracy. Details about the test cases can be found in Toro's book [43]. We set

Test	$[\rho_L, v_L, p_L]$	$[\rho_R, v_R, p_R]$	x_0	T
Modified Sod's Test (Toro's Test 1)	[1.0, 0.75, 1.0]	[0.125, 0.0, 0.1]	0.3	0.2
Toro's Test 3	[1, 0, 1000]	[1, 0, 0.01]	0.5	0.012

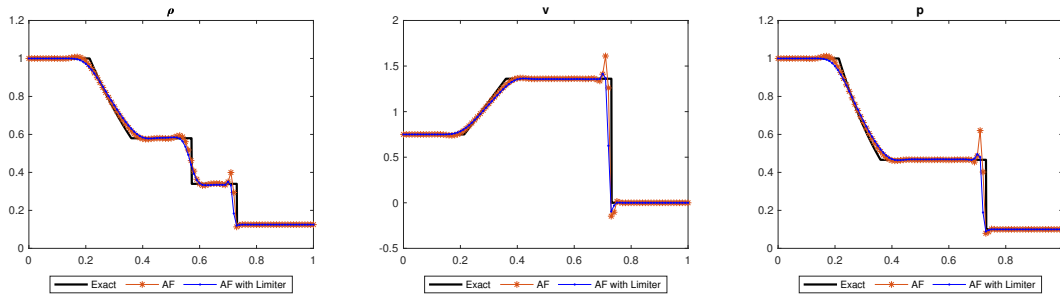
Table E.1: The initial values of test cases for the Euler equations

the initial conditions as follows.

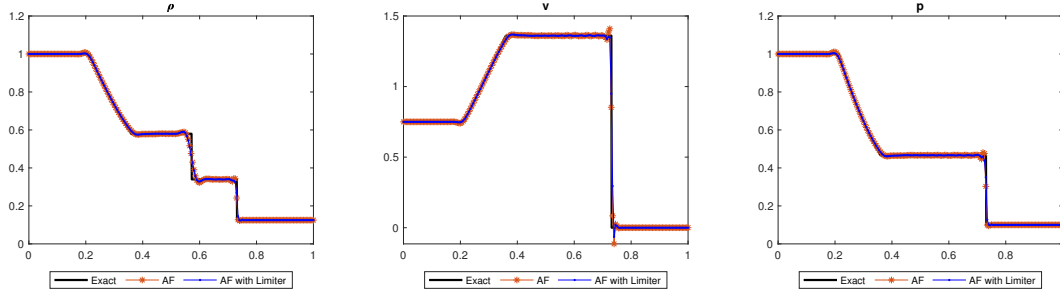
$$\begin{cases} [\rho(x, 0), v(x, 0), p(x, 0)] = [\rho_L, v_L, p_L] & 0 \leq x < x_0 \\ [\rho(x, 0), v(x, 0), p(x, 0)] = [\rho_R, v_R, p_R] & x_0 \leq x \leq 1, \end{cases}$$

Transmissive boundary conditions are imposed on the domain $\Omega = [0, 1]$. If the velocity v at the boundary points inwards, we will not distribute the discrepancy to the nodal values there. The domain is divided into 100 cells. We roughly estimate the maximum wave speed: $v_{\max} = 3$ for the Modified Sod's test, $v_{\max} = 50$ for Toro's test 3. For $p = 1$, $\Delta t = 0.6\Delta x/p/v_{\max}$. The parameter for the adjustment of distribution weights is set to be $q = 0.4$. For $p = 2$, $\Delta t = 0.3\Delta x/p/v_{\max}$, $q_0 = 0.5$, $q_2 = 0.25$.

We plot the numerical solutions at $t = T$ along with the exact solutions generated by the code `E1RPEX.F` in the library `NUMERICA` developed by Toro [43]. See Figure E.2 and Figure E.3. We notice that the new treatment of conservation law increased the stability and the limiter further improve the solution quality.

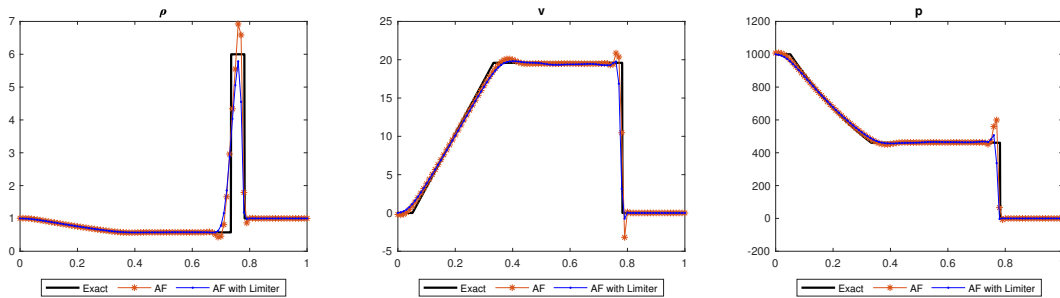


(a) $p = 1, \beta_1 = \beta_2 = 1, \Delta x = 1/100$

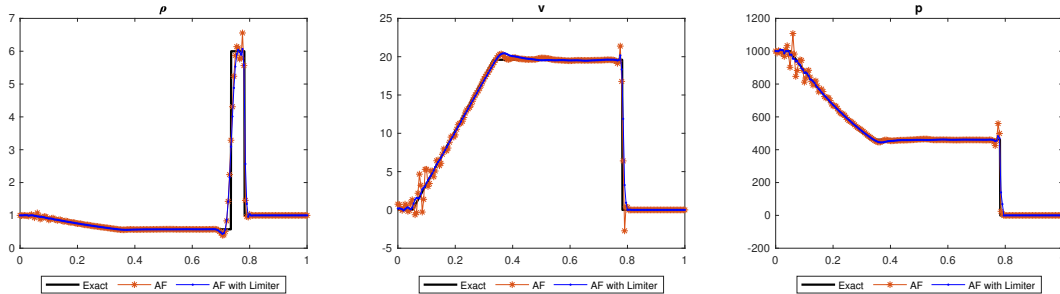


(b) $p = 2, \alpha = \beta_1 = \beta_2 = 1, \Delta x = 1/100$

Figure E.2: Numerical solutions to the modified Sod's test generated by the Active Flux schemes with simple limiters.



(a) $p = 1, \beta_1 = \beta_2 = 1, \Delta x = 1/100$



(b) $p = 2, \alpha = \beta_1 = \beta_2 = 1, \Delta x = 1/100$

Figure E.3: Numerical solutions to Toro's test 3 generated by the Active Flux schemes with simple limiters.

BIBLIOGRAPHY

- [1] Rémi Abgrall. Toward the ultimate conservative scheme: following the quest. *Journal of Computational Physics*, 167(2):277–315, 2001.
- [2] Rémi Abgrall. Residual Distribution schemes: current status and future trends. *Computers & Fluids*, 35(7):641–669, 2006.
- [3] Rémi Abgrall. Some remarks about conservation for Residual Distribution schemes. *Computational Methods in Applied Mathematics*, 18(3):327–351, 2018.
- [4] Rémi Abgrall. A combination of Residual Distribution and the Active Flux formulations or a new class of schemes that can combine several writings of the same hyperbolic problem: application to the 1D Euler equations. *arXiv preprint arXiv:2011.12572*, 2020.
- [5] Wasilij Barsukow. The Active Flux scheme for nonlinear problems. *Journal of Scientific Computing*, 86(1):1–34, 2021.
- [6] Wasilij Barsukow, Jonathan Hohm, Christian Klingenberg, and Philip L Roe. The Active Flux scheme on Cartesian grids and its low Mach number limit. *Journal of Scientific Computing*, 81(1):594–622, 2019.
- [7] Saray Busto, Michael Dumbser, Cipriano Escalante, Nicolas Favrie, and Sergey Gavrilyuk. On high order ADER discontinuous Galerkin schemes for first order hyperbolic reformulations of nonlinear dispersive systems. *Journal of Scientific Computing*, 87(2):1–47, 2021.
- [8] Carlo Cattaneo. A form of heat-conduction equations which eliminates the paradox of instantaneous propagation. *Comptes Rendus*, 247:431, 1958.
- [9] Erik Chudzik, Christiane Helzel, and David Kerkmann. The Cartesian Grid Active Flux Method: Linear stability and bound preserving limiting. *Applied Mathematics and Computation*, 393:125501, 2021.
- [10] Richard Courant and Kurt Otto Friedrichs. *Supersonic flow and shock waves*, volume 21. Springer Science & Business Media, 1999.
- [11] Richard Courant and David Hilbert. *Methods of Mathematical Physics. Volume II, Partial Differential Equations*, 1962.
- [12] Michael Dumbser, Ilya Peshkov, and Evgeniy Romenski. A unified hyperbolic formulation for viscous fluids and elastoplastic solids. In *XVI International Conference on Hyperbolic Problems: Theory, Numerics, Applications*, pages 451–463. Springer, 2016.

- [13] Timothy Andrew Eymann. *Active Flux Schemes*. PhD thesis, University of Michigan, 2013.
- [14] Doreen Fan, Jungyeoul Maeng, and Philip L Roe. A new approach to numerical conservation laws, ii acoustics. *Journal of Computational Physics*, 2019. submitted.
- [15] Doreen Fan and Philip L Roe. Investigations of a new scheme for wave propagation. In *22nd AIAA Computational Fluid Dynamics Conference*, page 2449, 2015.
- [16] Duoming Fan. *On the acoustic component of Active Flux schemes for nonlinear hyperbolic conservation laws*. PhD thesis, University of Michigan, 2017.
- [17] Richard P Feynman, Robert B Leighton, and Matthew Sands. *The Feynman lectures on physics*, volume 2. Basic books, 2011.
- [18] Fanchen He and Philip L Roe. A novel numerical scheme based on Active Flux method for Hyperbolic heat equations in multidimensional space. In *AIAA Aviation 2019 Forum*, page 3638, 2019.
- [19] Fanchen He and Philip L Roe. The Treatment of Conservation in the Active Flux Method. In *AIAA AVIATION 2020 FORUM*, page 3032, 2020.
- [20] Christiane Helzel, David Kerkmann, and Leonardo Scandurra. A new ADER method inspired by the Active Flux method. *Journal of Scientific Computing*, 80(3):1463–1497, 2019.
- [21] Satoshi Ii and Feng Xiao. CIP/multi-moment finite volume method for Euler equations: A semi-Lagrangian characteristic formulation. *Journal of Computational Physics*, 222(2):849–871, 2007.
- [22] Antony Jameson, Timothy Baker, and Nigel Weatherill. Calculation of inviscid transonic flow over a complete aircraft. In *24th Aerospace Sciences Meeting*, 1986.
- [23] Randall J LeVeque. *Numerical methods for conservation laws*, volume 132. Springer, 1992.
- [24] Chia-Chiao Lin and Lee A Segel. *Mathematics applied to deterministic problems in the natural sciences*, volume 1. SIAM, 1988.
- [25] Jialin Lou, Lingquan Li, Hong Luo, and Hiroaki Nishikawa. Reconstructed Discontinuous Galerkin Methods for Linear Advection-Diffusion Equations Based on First-Order Hyperbolic System. *Journal of Computational Physics*, 2018.
- [26] Jungyeoul Maeng. *On the advective component of Active Flux schemes for nonlinear hyperbolic conservation laws*. PhD thesis, University of Michigan, 2017.
- [27] Jungyeoul Maeng, Doreen Fan, and Philip L Roe. A new approach to numerical conservation laws, i motivation and advection. *Journal of Computational Physics*, 2019. submitted.
- [28] Parviz Moin and Krishnan Mahesh. Direct numerical simulation: a tool in turbulence research. *Annual review of fluid mechanics*, 30(1):539–578, 1998.

- [29] Hiroaki Nishikawa. A first-order system approach for diffusion equation. i: Second-order residual-distribution schemes. *Journal of Computational Physics*, 227(1):315–352, 2007.
- [30] Hiroaki Nishikawa. New-generation hyperbolic Navier-Stokes schemes: $O(1/h)$ speed-up and accurate viscous/heat fluxes. In *20th AIAA Computational Fluid Dynamics Conference*, page 3043, 2011.
- [31] Hiroaki Nishikawa. First, second, and third order finite-volume schemes for Navier-Stokes equations. In *7th AIAA Theoretical Fluid Mechanics Conference*, page 2091, 2014.
- [32] Hiroaki Nishikawa. Alternative formulations for first-, second-, and third-order hyperbolic Navier-Stokes schemes. In *22nd AIAA Computational Fluid Dynamics Conference*, page 2451, 2015.
- [33] Hiroaki Nishikawa and Philip L Roe. Third-order Active-Flux scheme for advection diffusion: hyperbolic diffusion, boundary condition, and Newton solver. *Computers & Fluids*, 125:71–81, 2016.
- [34] Per-Olof Persson and Gilbert Strang. A simple mesh generator in MATLAB. *SIAM review*, 46(2):329–345, 2004.
- [35] Ilya Peshkov and Evgeniy Romenski. A hyperbolic model for viscous Newtonian flows. *Continuum Mechanics and Thermodynamics*, 28(1-2):85–104, 2016.
- [36] Philip Roe. Is discontinuous reconstruction really a good idea? *Journal of Scientific Computing*, 73(2-3):1094–1114, 2017.
- [37] Philip Roe. Designing CFD methods for bandwidth—A physical approach. *Computers & Fluids*, 214:104774, 2021.
- [38] Philip L Roe and Mohit Arora. Characteristic-based schemes for dispersive waves i. the method of characteristics for smooth solutions. *Numerical Methods for Partial Differential Equations*, 9(5):459–505, 1993.
- [39] Philip L Roe, Jungyeoul Maeng, and Doreen Fan. Comparing Active Flux and Discontinuous Galerkin Methods for Compressible Flow. In *2018 AIAA Aerospace Sciences Meeting*, page 0836, 2018.
- [40] Philip L Roe, Jungyeoul Maeng, and Doreen Fan. A new approach to numerical conservation laws, III. Euler equations and related systems. *Journal of Computational Physics*, 2019. submitted.
- [41] Lee A Segel. *Mathematics applied to continuum mechanics*, volume 52. SIAM, 2007.
- [42] Jeffrey Slotnick, Abdollah Khodadoust, Juan Alonso, David Darmofal, William Gropp, Elizabeth Lurie, and Dimitri Mavriplis. CFD vision 2030 study: a path to revolutionary computational aerosciences. 2014.
- [43] Eleuterio F Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.

- [44] Eleuterio F Toro and Gino I Montecinos. Advection-diffusion-reaction equations: hyperbolization and high-order ADER discretizations. *SIAM Journal on Scientific Computing*, 36(5):A2423–A2457, 2014.
- [45] Bram Van Leer. Towards the ultimate conservative difference scheme. iv. a new approach to numerical convection. *Journal of Computational Physics*, 23(3):276–299, 1977.
- [46] Zhijian J Wang, Krzysztof Fidkowski, Rémi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, Herman Deconinck, Ralf Hartmann, Koen Hillewaert, Hung T Huynh, et al. High-order CFD methods: current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, 2013.