APPLICATION OF THE ENTROPY VISCOSITY METHOD AND THE FLUX-CORRECTED TRANSPORT ALGORITHM TO SCALAR TRANSPORT EQUATIONS AND THE SHALLOW WATER EQUATIONS

A Dissertation

by

JOSHUA EDMUND HANSEL

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Jean Ragusa
Committee Members,	Marvin Adams
	Jim Morel
	Jean-Luc Guermond
Head of Department,	Yassin Hassan

August 2016

Major Subject: Nuclear Engineering

ABSTRACT

The Flux-Corrected Transport (FCT) algorithm, in conjunction with the entropy viscosity method, was applied with the continuous finite element method to (a) a scalar transport equation that includes reaction and source terms; and (b) the shallow water equations. The resulting scheme shows convergence to the entropy solution, is positivity-preserving, and reduces or eliminates the onset of spurious oscillations. For smooth problems, second-order spatial accuracy is achieved if adequate solution bounds are used in the FCT algorithm. For the scalar transport equation, the method of characteristics is used to derive local solution bounds to impose on the numerical solution. For the shallow water equations, local transformations are made to characteristic variables for the limitation process of FCT. Explicit SSPRK time discretizations are considered for both scalar transport and the shallow water equations, and additionally for scalar transport, Theta time discretizations and steady-state are considered. Explicit FCT schemes are shown to be relatively robust, but implicit/steady-state FCT schemes, however, are shown to have significant nonlinear convergence issues in many cases.

To Siobhán

ACKNOWLEDGMENTS

First and foremost, I would like to thank my advisor, Dr. Jean Ragusa, for his years of guidance, wisdom, patience, and expertise. I would also like to thank Dr. Marco Delchini for his time, effort, and knowledge in the development of methods for the shallow water equations.

I would like to thank Dr. Jean-Luc Guermond and Dr. Bojan Popov for their time and expertise in entropy-based artificial dissipation and conservation law methodology. Their recent work on discrete maximum principle- preserving and invariant domain methods was the foundation of this dissertation.

I would like to thank all of those who have contributed to the development of the `deal.II` finite element library, which has been a tremendous help in the implementation of the methods given in this dissertation. I would especially like to thank Dr. Wolfgang Bangerth and Dr. Bruno Turcksin, who have quickly answered many questions I had regarding the use of `deal.II`.

I would like to thank those who have contributed to the development of FCT methods over the years, whose works were essential in the development of the methodology used in this research. I would like to thank Dr. Dmitri Kuzmin for his responses in my questions about FCT.

I would like to thank the the U.S. Department of Energy for providing funding for this research through the Nuclear Energy University Programs (NEUP) fellowship. I would also like to thank Dr. Richard Martineau of Idaho National Laboratory for providing funding of the last two semesters of research.

Lastly, I would like to thank my friends and family for their love and support.

NOMENCLATURE

BE	backward (implicit) Euler
CFEM	continuous finite element method
CN	Crank-Nicolson
DI	domain-invariant
DMP	discrete maximum principle
DoF	degree of freedom
EV	entropy viscosity
FCT	flux-corrected transport
FE	forward (explicit) Euler
FEM	finite element method
FV	finite volume
IVP	initial value problem
LED	local extremum diminishing
ODE	ordinary differential equation
PDE	partial differential equation
RK	Runge-Kutta
SS	steady-state
SSPRK	strong stability-preserving Runge-Kutta
SSPRK33	3-stage, 3-order-accurate SSPRK scheme

TABLE OF CONTENTS

Page

ABSTRACT ii

DEDICATION iii

ACKNOWLEDGMENTS iv

NOMENCLATURE v

TABLE OF CONTENTS vi

LIST OF FIGURES ix

LIST OF TABLES xiii

1. INTRODUCTION 1

    1.1 Model Equations 1
    1.2 Previous Work on Obtaining Non-Negative Transport Solutions 4
    1.3 Previous Work on Mitigating Spurious Oscillations in Hyperbolic Conservation Law Problems 5

2. METHODOLOGY 8

    2.1 Model Equations 8
        2.1.1 Scalar Conservation Laws 8
        2.1.2 The Shallow Water Equations 9
    2.2 Discretization 14
        2.2.1 Spatial Discretization for Scalar Conservation Laws 14
        2.2.2 Spatial Discretization for Conservation Law Systems 21
        2.2.3 Temporal Discretization 25
    2.3 Low-Order Scheme for Scalar Conservation Laws 28
        2.3.1 Low-Order Viscosity 29
        2.3.2 Low-Order System 31
        2.3.3 M-Matrix Property 32
        2.3.4 Positivity Preservation 37
        2.3.5 Local Discrete Maximum Principles 40
    2.4 Low-Order Scheme for Conservation Law Systems 44
        2.4.1 Invariant Domain 44
        2.4.2 Low-Order System 48
    2.5 High-Order Scheme for Scalar Conservation Laws 48

2.5.1 Entropy Viscosity 48
2.5.2 High-Order System 50
2.6 High-Order Scheme for Conservation Law Systems 51
2.6.1 Entropy Viscosity 51
2.6.2 High-Order System 54
2.7 FCT Scheme for Scalar Conservation Laws 54
2.7.1 FCT System 55
2.7.2 Solution Bounds 59
2.7.3 Antidiffusion Bounds 59
2.7.4 Limiting Coefficients 69
2.8 FCT Scheme for Systems of Conservation Laws 74
2.8.1 Introduction to FCT for Systems 74
2.8.2 General FCT Scheme for Systems 75
2.9 Implementation 78
2.9.1 Nonlinear Iteration 78
2.9.2 FCT Nonlinear Iteration 83

3. RESULTS 85

3.1 Overview 85
3.2 Scalar Transport 85
3.2.1 Overview 85
3.2.2 Convergence Studies 89
3.2.3 Multi-Region Test Problem 105
3.2.4 Void-to-Absorber Test Problem 106
3.2.5 Skew Void-to-Absorber Test Problem 114
3.2.6 Glance-in-Void Test Problem 119
3.2.7 Obstruction Test Problem 122
3.2.8 Source Void-to-Absorber Test Problem 129
3.2.9 Source-in-Absorber Test Problem 137
3.2.10 Interface Test Problem 143
3.2.11 Three-Region Test Problem 149
3.2.12 Summary 155
3.3 The Shallow Water Equations 159
3.3.1 Overview 159
3.3.2 1-D Dam Break 159
3.3.3 Bathtub Test Problem 167
3.3.4 2-D Dam Break 171

4. CONCLUSIONS 176

REFERENCES 178

APPENDIX A. DERIVATION OF LOCAL SOLUTION BOUNDS FOR LINEAR TRANSPORT USING THE METHOD OF CHARACTERISTICS 182

A.1 Introduction 182
A.2 Integral Form of the Linear Transport Equation 182
A.3 Local Maximum Principles 184

APPENDIX B. DERIVATION OF THE ENTROPY FLUX FOR THE SHALLOW WATER EQUATIONS 191

APPENDIX C. BOUNDARY CONDITIONS FOR THE SHALLOW WATER EQUATIONS 197

C.1 Characteristic Boundary Conditions for the Shallow Water Equations .. 197
C.2 Wall Boundary Conditions for the Shallow Water Equations 199

APPENDIX D. DERIVATION OF THE MAX WAVE SPEED FOR THE SHALLOW WATER EQUATIONS 201

D.1 Shock Wave 203
D.2 Rarefaction Wave 205
D.3 Obtaining the Solution in the Star Region 205
D.4 Fast Estimate of Maximum Wave Speed 205

LIST OF FIGURES

FIGURE Page

2.1 Illustration of Shared Support between Test Functions 16

2.2 Illustration of Cell Degree of Freedom Indices $\mathcal{I}_K$ 29

2.3 Limiter Input and Output..... 72

2.4 Multi-Pass Limiting Diagram 73

3.1 Comparison of Solutions for Convergence Test Problem 1 with 32 Cells 94

3.2 Viscosity Profiles for Convergence Test Problem 1 with 32 Cells..... 95

3.3 Comparison of Errors for Convergence Test Problem 1 96

3.4 Comparison of FCT Solutions with Different Solution Bounds with 32 Cells 97

3.5 Comparison of Error of FCT Solutions with Different Solution Bounds 98

3.6 Comparison of Solutions for Convergence Test Problem 2 with 32 Cells 101

3.7 Viscosity Profiles for Convergence Test Problem 2 with 32 Cells..... 102

3.8 Comparison of Errors for Convergence Test Problem 2 103

3.9 Convergence of Entropy Residual and Entropy Jumps for Convergence Test Problem 2 104

3.10 Comparison of Solutions for 2-D Normal Void-to-Absorber Test Problem Using Explicit Euler Time Discretization and Low-Order DMP Solution Bounds..... 109

3.11 Comparison of Solutions for 2-D Normal Void-to-Absorber Test Problem Using SSPRK33 Time Discretization and Low-Order DMP Solution Bounds..... 110

3.12 Viscosity Profiles for the 2-D Void-to-Absorber Test Problem Using SSPRK33 Time Discretization..... 111

3.13 Comparison of Solutions for the 3-D Normal Void-to-Absorber Test Problem Using SSPRK33 Time Discretization and Low-Order DMP Solution Bounds 112

3.14 Comparison of Steady-State Solutions for 2-D Normal Void-to-Absorber Test Problem with 4096 Cells with Exact Solution Bounds 113

3.15 Comparison of Solutions for the Skew Void-to-Absorber Test Problem Using Explicit Euler Time Discretization and DMP Solution Bounds with 16384 Cells 117

3.16 Comparison of Solutions for the Skew Void-to-Absorber Test Problem Using SSPRK33 Time Discretization and DMP Solution Bounds with 16384 Cells 118

3.17 Viscosity Profiles for the the Skew Void-to-Absorber Test Problem Using SSPRK33 Time Discretization 118

3.18 Comparison of Solutions for the Glance-in-Void Test Problem Using Explicit Euler Time Discretization and DMP Solution Bounds with 4096 Cells..... 121

3.19 Comparison of Solutions for the Obstruction Test Problem Using Explicit Euler Time Discretization with DMP Solution Bounds..... 124

3.20 Comparison of Solutions for the Obstruction Test Problem Using Explicit Euler Time Discretization with DMP Solution Bounds..... 125

3.21 EV-FCT Solution for the Obstruction Test Problem Using Implicit Euler Time Discretization with DMP Solution Bounds 127

3.22 EV-FCT Solution for the Obstruction Test Problem Using Steady-State Time Discretization with DMP Solution Bounds 128

3.23 Comparison of Solutions for the Source-Void-to-Absorber Problem Using SSPRK33 with DMP Solution Bounds with 32 Cells..... 131

3.24 Comparison of Solutions for the Source-Void-to-Absorber Problem Using SSPRK33 with DMP Solution Bounds with 256 Cells 132

3.25 Steady-State Solutions for the Source-Void-to-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 1$ and DMP Solution Bounds 133

3.26 Steady-State Solutions for the Source-Void-to-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 0$ and DMP Solution Bounds 134

3.27 Steady-State Solutions for the Source-Void-to-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions and DMP Solution Bounds 135

3.28 Steady-State Solutions for the Source-Void-to-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions and Boundary Penalty and DMP Solution Bounds 136

3.29 Steady-State Solutions for the Source-in-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 1$ 139

3.30 Steady-State Solutions for the Source-in-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 0$ 140

3.31 Steady-State Solutions for the Source-in-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions 141

3.32 Steady-State Solutions for the Source-in-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions and Boundary Penalty 142

3.33 Comparison of Steady-State FCT Solutions for the Interface Test Problem Obtained Using Original and Modified Analytic Solution Bounds with 32 Cells..... 145

3.34 Comparison of Steady-State FCT Solutions for the Interface Test Problem Obtained with Non-Upwind Analytic Solution Bounds with Single-Pass and Multi-Pass Limiting 147

3.35 Comparison of Steady-State FCT Solutions for the Interface Test Problem Obtained with Upwind Analytic Solution Bounds with Single-Pass and Multi-Pass Limiting 148

3.36 Comparison of Solutions for the 3-Region Test Problem Using SSPRK-33 and DMP Solution Bounds 151

3.37 Comparison of Solutions for the 3-Region Test Problem Using SSPRK-33 and Analytic Solution Bounds..... 152

3.38 Comparison of Solutions for the 3-Region Test Problem Using SSPRK-33 and Modified Analytic Solution Bounds 153

3.39 Comparison of Solutions for the 3-Region Test Problem Using SSPRK-33 and Upwind Analytic Solution Bounds 154

3.40 Comparison of Height Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 32 Cells..... 161

3.41 Comparison of Momentum Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 32 Cells..... 162

3.42 Comparison of Height Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 256 Cells 163

3.43 Comparison of Momentum Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 256 Cells 164

3.44 Comparison of Height Solutions for the 1-D Dam Break Test Problem Using SSPRK33 Time Discretization with 256 Cells 165

3.45 Comparison of Momentum Solutions for the 1-D Dam Break Test Problem Using SSPRK33 Time Discretization with 256 Cells 166

3.46 Initial Height Profile for the Bathtub Test Problem..... 167

3.47 Comparison of Low-Order and High-Order Solutions for the Bathtub Test Problem Using Explicit Euler Time Discretization 170

3.48 Initial Height Profile for the 2-D Dam Break Test Problem..... 173

3.49 Comparison of Height Solutions for the 2-D Dam Break Test Problem 174

3.50 Comparison of Solution Surfaces for the 2-D Dam Break Test Problem 174

3.51 Comparison of Viscosity Profiles for the 2-D Dam Break Test Problem 175

A.1 Illustration of Neighborhoods $L(\mathbf{x}, \tau)$ and $N(\mathbf{x}, \tau)$..... 185

LIST OF TABLES

TABLE		Page
2.1	Discrete Maximum Principles	60
2.2	Nonlinear System Matrix and Right-hand-side Vector for Different Schemes	82
3.1	Convergence Test Problem 1 Summary	93
3.2	Convergence Test Problem 1 Run Parameters	93
3.3	Convergence Test Problem 2 Summary	99
3.4	Convergence Test Problem 2 Run Parameters	100
3.5	Normal Void-to-Absorber Test Problem Summary	107
3.6	Normal Void-to-Absorber Test Problem Run Parameters	108
3.7	Skew Void-to-Absorber Test Problem Summary	114
3.8	Skew Void-to-Absorber Test Problem Run Parameters	115
3.9	Glance-in-Void Test Problem Summary	119
3.10	Normal Void-to-Absorber Test Problem Run Parameters	120
3.11	Obstruction Test Problem Summary	122
3.12	Obstruction Test Problem Run Parameters	123
3.13	Nonlinear Iterations vs. CFL Number for the Obstruction Test Problem Using Implicit Euler Time Discretization with 256 Cells	126
3.14	Nonlinear Iterations vs. Number of Cells for the Obstruction Test Problem Using Implicit Euler Time Discretization with CFL = 1	126
3.15	Nonlinear Iterations vs. Number of Cells for the Steady-State Obstruction Test Problem	127
3.16	Source-Void-to-Absorber Test Problem Summary	129
3.17	Source-Void-to-Absorber Test Problem Run Parameters	130

3.18 Nonlinear Iterations vs. Number of Cells for the Source-Void-to-Absorber Test Problem Using Implicit Euler Time Discretization with CFL = 1 135

3.19 Nonlinear Iterations vs. CFL Number for the Source-Void-to-Absorber Test Problem Using Implicit Euler Time Discretization with 128 Cells 136

3.20 Source-in-Absorber Test Problem Summary..... 137

3.21 Source-in-Absorber Test Problem Run Parameters 138

3.22 Interface Test Problem Summary..... 143

3.23 Interface Test Problem Run Parameters 144

3.24 FCT Iterations Required for Different Solution Bounds for the Interface Test Problem 148

3.25 Three-Region Test Problem Summary..... 149

3.26 Three-Region Test Problem Run Parameters 150

3.27 Bathtub Test Problem Summary 160

3.28 1-D Dam Break Test Problem Run Parameters..... 160

3.29 Bathtub Test Problem Summary 168

3.30 Bathtub Test Problem Run Parameters..... 168

3.31 2-D Dam Break Test Problem Summary 171

3.32 2-D Dam Break Test Problem Run Parameters..... 172

C.1 Signs of Eigenvalues for Different Cases..... 198

C.2 Summary of Open Boundary Conditions for the 1-D Shallow Water Equations..... 199

C.3 Summary of Wall Boundary Conditions for the 1-D Shallow Water Equations..... 200

1. INTRODUCTION

This section is organized as follows: Section 1.1 introduces the model equations considered in this dissertation, Section 1.2 reviews previous work on obtaining non-negative transport solutions, and Section 1.3 reviews previous work on the solution of hyperbolic conservation laws, which includes an introduction to entropy viscosity and FCT.

1.1 Model Equations

This research considers two physical models, both of which are of great importance in the field of nuclear engineering: linear transport, and the shallow water equations (SWE). The linear transport model considered in this research is the following:

$$\frac{1}{v}\frac{\partial\psi}{\partial t} + \mathbf{\Omega}\cdot\nabla\psi\left(\mathbf{x},t\right) + \Sigma_{\mathrm{t}}(\mathbf{x})\psi\left(\mathbf{x},t\right) = Q\left(\mathbf{x},t\right)\,, \tag{1.1}$$

where $\psi\left(\mathbf{x},t\right)$ is the angular flux in direction $\mathbf{\Omega}$, v is the transport speed, $\Sigma_{\mathrm{t}}(\mathbf{x})$ is the macroscopic total cross-section, and $Q\left(\mathbf{x},t\right)$ is an extraneous source. This model is a special case of the energy-dependent neutron transport equation,

$$\begin{aligned}\frac{1}{v(E)}\frac{\partial\psi}{\partial t} + \mathbf{\Omega}\cdot\nabla\psi(\mathbf{x},\mathbf{\Omega},E,t) + \Sigma_{\mathrm{t}}(\mathbf{x},E,t)\psi(\mathbf{x},\mathbf{\Omega},E,t) = Q_{\mathrm{ext}}(\mathbf{x},\mathbf{\Omega},E,t)\\ + \frac{\chi_{\mathrm{p}}(E)}{4\pi}\int_0^\infty dE'\nu_{\mathrm{p}}(\mathbf{x},E',t)\Sigma_{\mathrm{f}}(\mathbf{x},E',t)\phi(\mathbf{x},\mathbf{\Omega},E',t) + \sum_{i=1}^{n_{\mathrm{d}}}\frac{\chi_{\mathrm{d},i}(E)}{4\pi}\lambda_i C_i\left(\mathbf{x},t\right)\\ + \int_0^\infty dE'\int_{4\pi} d\mathbf{\Omega}'\Sigma_{\mathrm{s}}(\mathbf{x},E'\to E,\mathbf{\Omega}'\to\mathbf{\Omega},t)\psi(\mathbf{x},\mathbf{\Omega}',E',t)\,,\end{aligned} \tag{1.2}$$

where E is energy, Q_{ext} is the extraneous source, Σ_{f} is the fission cross section, Σ_{s} is the double-differential scattering cross section, χ_{p} is the prompt neutron energy

spectrum, ν_{p} is the prompt neutron yield, n_{d} is the number of delayed neutron precursors, and $\chi_{\mathrm{d},i}$, λ_i, and C_i are the delayed neutron energy spectrum, decay constant, and concentration of precursor i, respectively. Equation (1.1) is obtained by lumping the extraneous source, fission source, and scattering source into a single source term Q:

$$\begin{aligned} Q(\mathbf{x},\mathbf{\Omega},E,t) &\equiv Q_{\mathrm{ext}}(\mathbf{x},\mathbf{\Omega},E,t) \\ &+ \frac{\chi_{\mathrm{p}}(E)}{4\pi}\int_0^\infty dE'\nu_{\mathrm{p}}(\mathbf{x},E',t)\Sigma_{\mathrm{f}}(\mathbf{x},E',t)\phi(\mathbf{x},\mathbf{\Omega},E',t) + \sum_{i=1}^{n_{\mathrm{d}}}\frac{\chi_{\mathrm{d},i}(E)}{4\pi}\lambda_i C_i\left(\mathbf{x},t\right) \\ &+ \int_0^\infty dE'\int_{4\pi} d\mathbf{\Omega}'\Sigma_{\mathrm{s}}(\mathbf{x},E'\rightarrow E,\mathbf{\Omega}'\rightarrow\mathbf{\Omega},t)\psi(\mathbf{x},\mathbf{\Omega}',E',t)\,. \end{aligned} \tag{1.3}$$

This approach is representative of a typical approach in the iterative solution of the transport equation, known as source iteration:

$$\frac{1}{v}\frac{\partial\psi^{(\ell)}}{\partial t} + \mathbf{\Omega}\cdot\nabla\psi^{(\ell)} + \Sigma_{\mathrm{t}}\psi^{(\ell)} = Q^{(\ell-1)}\,, \tag{1.4}$$

where $Q^{(\ell-1)}$ is evaluated with the previous solution iterate $\psi^{(\ell-1)}$. The multigroup method may still be used for discretization in energy, and the Discrete Ordinates, S_N, method may be used for discretization in angle, without invalidating the methodology developed in this dissertation. Note that the arguments E and $\mathbf{\Omega}$ are dropped for the remainder of the dissertation because this dependence is not important since the omission of scattering and fission terms decouples the equation in energy and direction. The units are thus intentionally left ambiguous.

The transport equation is often referred to as the Boltzmann equation because it is a linearized form of the equation developed by Boltzmann in the 1800s to study kinetic theory of gases [8][5]. In general, transport theory is used in describing the

transport of particles or waves through some background media. Such particles might include neutrons, electrons, ions, gas molecules, or photons. Since its origination, transport theory has evolved independently in many different disciplines and applications such as nuclear reactors, atmospheric science, radiation therapy, radiation shielding, and stars. In fact, much of the early transport theory was developed to meet the needs of astrophysical research in the study of stellar and planetary atmospheres [8]. In the field of nuclear engineering, the transport equation is especially important, as the transport equation is used to guide nuclear reactor design and operation as well as design and analysis of radiation shielding.

The transport equation is classified as a hyperbolic partial differential equation (PDE). Hyperbolic PDEs are characterized by finite wave speeds in the solution and share a common body of methodology; the methodology presented in this dissertation is largely applicable to other systems of hyperbolic equations. In addition to the transport equation, this dissertation considers the shallow water equations (SWE), which are also hyperbolic:

$$\frac{\partial h}{\partial t} + \frac{\partial (hu)}{\partial x} + \frac{\partial (hv)}{\partial y} = 0\,, \tag{1.5a}$$

$$\frac{\partial (hu)}{\partial t} + \frac{\partial}{\partial x}\left(hu^2 + \frac{1}{2}gh^2\right) + \frac{\partial}{\partial y}\left(huv\right) = 0\,, \tag{1.5b}$$

$$\frac{\partial (hv)}{\partial t} + \frac{\partial}{\partial x}\left(huv\right) + \frac{\partial}{\partial y}\left(hv^2 + \frac{1}{2}gh^2\right) = 0\,, \tag{1.5c}$$

where h is fluid the height, u is the x-velocity, v is the y-velocity, and g is acceleration due to gravity. The SWE are derived from the Navier-Stokes equations by making the assumption that the horizontal length scale is much larger than the vertical length scale. Under this assumption, the vertical velocity is small (not necessarily zero), and upon depth-integrating the equations, the vertical velocity is removed.

The SWE have many applications, including coastal flows, lakes, rivers, dam breaks, and tsunamis [18].

The importance of the tsunami modeling in nuclear engineering has been highlighted in recent years. The magnitude 9.0 earthquake that occurred off the coast of Tohoku in 2011, often referred to as the Great East Japan earthquake [2], produced a tsunami that not only resulted in an enormous loss of life and infrastructure, but also resulted in the most severe nuclear accident since the Chernobyl disaster in 1986 [3], becoming only the second accident in history to receive the highest rating of 7 on the International Nuclear and Radiological Event Scale (INES) [1]. Kanayama [17] performed a tsunami simulation of Hakata Bay using the viscous shallow water equations, which was used to evaluate damage to coastal areas of the Tohoku district.

1.2 Previous Work on Obtaining Non-Negative Transport Solutions

Negativities have long been an outstanding issue in the numerical solution of the neutron transport equation [20]. While negativities may not significantly degrade the accuracy of a method, they may cause drastic, unintended consequences where assumptions of non-negativity are employed.

There have been a number of attempts to address this issue for various spatial discretizations. Many rely on ad-hoc fix-ups, such as the classic set-to-zero fix-up for the diamond difference scheme [22]. Hamilton [15] introduced a similar fix-up method for the linear discontinuous (LD) finite element method (FEM) that conserves local particle balance and keeps the third-order accuracy of the standard LD FEM. Walters and Wareing [29] developed a nonlinear spatial differencing scheme for one-dimensional slab geometry. So-called characteristic methods were developed by Walters and Wareing [28] and Minor [26]. Wareing notes in [30] that these char-

acteristic methods are difficult to derive and implement and offers instead a nonlinear positive spatial differencing scheme called the exponential discontinuous scheme, which was applicable in 1-D, 2-D, and 3-D Cartesian meshes. More recently, Maginot has developed a consistent set-to-zero method for LD FEM [24], as well as a non-negative, Bilinear Discontinuous (BLD) method [25].

1.3 Previous Work on Mitigating Spurious Oscillations in Hyperbolic Conservation Law Problems

The solution of hyperbolic conservation law equations presents a number of unique challenges; in the vicinity of strong gradients and discontinuities, numerical solutions are prone to spurious oscillations that may generate unphysical values. For example, numerical schemes may generate negative solution values for physically non-negative quantities such as scalar flux or angular flux if adequate precautions are not taken. These negativities are not only undesirable because they are physically incorrect, but also because often numerical solution algorithms completely break down, causing simulations to terminate prematurely. Even more consequential is the possibility that these negative solution values go undiscovered and cause significant inaccuracies in quantities of interest. This is a particularly serious possibility, as these erroneous results may lead to poor design choices, thus presenting significant safety concerns.

The formation of spurious oscillations and negativities is a well-known issue in numerical discretizations of hyperbolic partial differential equations (PDEs), which include, for example, linear advection, Burger's equation, the inviscid Euler equations of gas dynamics, and the shallow water equations. These PDEs result from manipulating the corresponding integral conservation law equations; however, these manipulations are only valid when the solution is smooth - in the presence of shocks, the PDE form breaks down [21]. Thus it becomes necessary to work with these

equations in a weak form, which holds in the presence of shocks. However, the mathematical formulations for these problems do not necessarily yield unique weak solutions; this is a manifestation of the omission of some physics in the approximate hyperbolic PDE model [21].

To produce a unique, physically meaningful solution, it is necessary to enforce additional conditions, often called *admissibility conditions* or *entropy conditions*, which filter out spurious weak solutions, leaving only the physical, *entropy-satisfying* weak solution [21]. There are a number of entropy conditions that may be applied: some examples are the Lax entropy condition and the Oleinik entropy condition [21]; however, it is typically impractical to apply these conditions in a numerical simulation. The research in this dissertation employs the notion of an entropy-based artificial viscosity, based on the recent work of Guermond and others [13].

While entropy-based methods mitigate the issues of spurious oscillations and negativities, they still do not resolve the issues entirely, even though such methods help in convergence to the entropy solution; spurious oscillations and negativities are still present, although smaller in magnitude. To further address these issues, one can employ the Flux-Corrected Transport (FCT) algorithm, which has shown some success in the solution of hyperbolic conservation laws for several decades. The FCT algorithm was introduced in 1973 by Boris and Book [7] for finite difference discretizations of transport problems, and it has been applied to the finite element method more recently. The idea of FCT is to blend a low-order scheme that is monotone with a high-order scheme using a nonlinear limiting procedure. FCT takes the difference of the high-order and low-order schemes to define antidiffusive fluxes (or *correction* fluxes), which when added to the low-order scheme as a source, becomes equivalent to the high-order scheme. However, the FCT algorithm limits these antidiffusive fluxes to satisfy some physically-motivated criteria, such as discrete local-extremum-

diminishing (LED) bounds.

For the monotone, low-order method required by the FCT algorithm, this research uses the discrete maximum principle (DMP) preserving method introduced by Guermond [11] for the scalar transport model, and the invariant domain method for the case of conservation law systems, also introduced by Guermond [14]. The invariant domain property is a key property in ensuring monotonicity; it essentially ensures that the numerical solution will not leave a domain determined by the initial data [16]. For the case of a scalar conservation law, this property reduces to a discrete maximum principle. For the high-order method required by the FCT algorithm, FEM-FCT traditionally has used the Galerkin method without any artificial dissipation [19], which in some cases is adequate; however, in this work, an entropy-based dissipation is added to the scheme to enforce an entropy inequality, as performed by Guermond [13][12].

2. METHODOLOGY

This chapter is organized as follows. Section 2.1 gives the model equations considered in this research, Section 2.2 describes the spatial and temporal discretizations used in this research, Section 2.3 describes a first-order scheme for scalar conservation laws that is positivity-preserving and discrete maximum principle- preserving, Section 2.4 describes a first-order scheme for conservation law systems that is positivity-preserving and domain-invariant, Section 2.5 describes a high-order entropy-based scheme for scalar conservation laws, Section 2.6 describes a high-order entropy-based scheme for conservation law systems, Sections 2.7 and 2.8 describe flux-corrected transport (FCT) schemes for scalar conservation laws and conservation law systems, respectively, and Section 2.9 gives some implementation details.

2.1 Model Equations

2.1.1 Scalar Conservation Laws

The primary application of this research is radiation transport, as given by Equation (1.1); however, most of the analysis performed is valid for any scalar conservation law of the following form:

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{f}(u) + \sigma\left(\mathbf{x}, t\right) u\left(\mathbf{x}, t\right) = q\left(\mathbf{x}, t\right) \ , \tag{2.1}$$

where $u\left(\mathbf{x}, t\right)$ is a general scalar conserved quantity at position $\mathbf{x}$ and time t, $\mathbf{f}(u)$ is a general flux function, $\sigma\left(\mathbf{x}, t\right) \geq 0$ is a reaction coefficient, and $q\left(\mathbf{x}, t\right) \geq 0$ is a source term. Note that traditional FCT methodology does not consider the the presence of the reaction term $\sigma\left(\mathbf{x}, t\right) u\left(\mathbf{x}, t\right)$ or source term $q\left(\mathbf{x}, t\right)$ [19]; extension to include these terms is a significant driver for this research since this allows the application

of FCT to radiation transport, for example. Since the radiation transport equation has a linear flux function $\mathbf{f}(u)$, hereafter it is assumed that $\mathbf{f}(u) \equiv \mathbf{v}u$, where $\mathbf{v}$ is a constant velocity.

The radiation transport equation given by Equation (1.1) fits the model of Equation (2.1) by making the following substitutions:

$$u \to \psi\,, \quad \mathbf{v} \to v\mathbf{\Omega}\,, \quad \sigma \to v\Sigma_{\mathrm{t}}\,, \quad q \to vQ\,,$$

Initial conditions are included if the problem is transient:

$$u(\mathbf{x}, 0) = u^0(\mathbf{x}) \quad \forall \mathbf{x} \in \mathcal{D}\,. \tag{2.2}$$

Boundary conditions will depend on the chosen conservation law and the particular problem. This research assumes an incoming flux boundary condition:

$$u\left(\mathbf{x}, t\right) = u^{\mathrm{inc}}\left(\mathbf{x}, t\right) \quad \forall \mathbf{x} \in \partial\mathcal{D}^-, \quad \partial\mathcal{D}^- = \left\{\mathbf{x} \in \partial\mathcal{D} : \mathbf{v} \cdot \mathbf{n}(\mathbf{x}) < 0\right\}. \tag{2.3}$$

These conditions together make the problem well-posed, but for general nonlinear conservation laws, care must be taken to ensure that the boundary conditions used result in a well-posed problem.

Some options for the implementation of the incoming flux boundary condition are presented in Section 2.2.1.1.

2.1.2 The Shallow Water Equations

The shallow water equations, also known as the Saint-Venant equations, are an approximation of conservation of mass and momentum equations applied to free surface flows, which assume the fluid to be incompressible, non-viscous, and non-heat-

conducting [27]. The shallow water equations are derived by making the additional approximation that the vertical component of acceleration can be neglected due to horizontal length scales being much greater than the depth length scale and then depth-integrating the conservation equations [27][21][9]:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{u}) = \mathbf{s}(\mathbf{u}),$$
$$\mathbf{u} = \begin{bmatrix} h \\ q_x \\ q_y \end{bmatrix}, \quad \mathbf{F}(\mathbf{u}) = \begin{bmatrix} q_x & q_y \\ \frac{q_x^2}{h} + \frac{1}{2}gh^2 & \frac{q_x q_y}{h} \\ \frac{q_x q_y}{h} & \frac{q_y^2}{h} + \frac{1}{2}gh^2 \end{bmatrix}, \quad \mathbf{s}(\mathbf{u}) = \begin{bmatrix} 0 \\ -gh\frac{\partial b}{\partial x} \\ -gh\frac{\partial b}{\partial y} \end{bmatrix}, \tag{2.4}$$

written more concisely as

$$\mathbf{u} = \begin{bmatrix} h \\ \mathbf{q} \end{bmatrix}, \quad \mathbf{F}(\mathbf{u}) = \begin{bmatrix} \mathbf{q} \\ \frac{\mathbf{q} \otimes \mathbf{q}}{h} + \frac{1}{2}gh^2\mathbf{I} \end{bmatrix}, \quad \mathbf{s}(\mathbf{u}) = \begin{bmatrix} 0 \\ -gh\nabla b \end{bmatrix},$$

where h is the height of the water, which plays the role of density in the continuity equation, $\mathbf{q} = h\mathbf{v}$ is sometimes referred to as *discharge* and plays the role of momentum (hereafter, $\mathbf{q}$ will usually just be referred to as "momentum"), $\mathbf{v}$ is velocity, g is acceleration due to gravity, and b is the topography of the bottom terrain of the fluid body, hereafter referred to as the *bathymetry* function. Note that the shallow water equations are only valid in 1-D or 2-D, not 3-D, since they are depth-integrated equations.

The shallow water equations (SWE) are a popular model for flows in lakes, rivers, irrigation channels, and ocean shores, and thus are of great interest in hydrology, oceanography, and climate modeling [6][9].

Initial conditions are included if the problem is transient:

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}^0(\mathbf{x}) \quad \forall \mathbf{x} \in \mathcal{D}\,. \tag{2.5}$$

To complete the problem formulation, boundary conditions must be provided, some examples being Dirichlet boundary, open boundary, wall boundary, etc. One must be careful with specifying boundary conditions to have a well-posed problem for hyperbolic systems. In general a characteristic analysis is required; there is a large body of research addressing this area alone. For simplicity, problems in this work are chosen such that initial data never reaches the boundary or boundary conditions are implemented as natural conditions rather than using the method of characteristics. Appendix C gives some details on a selection of boundary conditions applicable to the shallow water equations.

2.1.2.1 Characteristics Analysis for the Shallow Water Equations

In this section, characteristics are considered for the shallow water equations.

The 1-D shallow water equations (with flat bottom topography) can be expressed as

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{0}\,, \quad \mathbf{u} = \begin{bmatrix} h \\ q_x \end{bmatrix}, \quad \mathbf{F}(\mathbf{u}) = \begin{bmatrix} q_x \\ \frac{q_x^2}{h} + \frac{1}{2}gh^2 \end{bmatrix}. \tag{2.6}$$

The spatial derivative of the flux function can be evaluated using chain rule, giving

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{A}(\mathbf{u})\frac{\partial \mathbf{u}}{\partial x} = \mathbf{0}\,, \quad \mathbf{A}(\mathbf{u}) \equiv \frac{\partial \mathbf{F}}{\partial \mathbf{u}} = \begin{bmatrix} 0 & 1 \\ a^2 - u^2 & 2u \end{bmatrix}, \tag{2.7}$$

where $\mathbf{A}(\mathbf{u})$ is referred to as the Jacobian matrix, and $a \equiv \sqrt{gh}$. The eigenvalues

and corresponding eigenvectors of this matrix are

$$\lambda_1(\mathbf{u}) = u - a\,, \quad \lambda_2(\mathbf{u}) = u + a\,, \tag{2.8}$$

$$\mathbf{k}_1(\mathbf{u}) = \begin{bmatrix} 1 \\ u-a \end{bmatrix}\,, \quad \mathbf{k}_2(\mathbf{u}) = \begin{bmatrix} 1 \\ u+a \end{bmatrix}\,. \tag{2.9}$$

Forming a matrix $\mathbf{K}(\mathbf{u})$ using the eigenvectors as columns and then taking its inverse gives

$$\mathbf{K}(\mathbf{u}) = \begin{bmatrix} 1 & 1 \\ u-a & u+a \end{bmatrix}\,, \quad \mathbf{K}(\mathbf{u})^{-1} = \begin{bmatrix} \frac{u+a}{2a} & -\frac{1}{2a} \\ \frac{a-u}{2a} & \frac{1}{2a} \end{bmatrix}\,. \tag{2.10}$$

Applying $\mathbf{K}(\mathbf{u})^{-1}$ from the left to Equation (2.6) gives

$$\mathbf{K}(\mathbf{u})^{-1}\frac{\partial \mathbf{u}}{\partial t} + \mathbf{K}(\mathbf{u})^{-1}\mathbf{A}(\mathbf{u})\mathbf{K}(\mathbf{u})\mathbf{K}(\mathbf{u})^{-1}\frac{\partial \mathbf{u}}{\partial x} = \mathbf{0}\,. \tag{2.11}$$

Multiplying the first equation by $-\frac{2g}{a}$ and the second by $\frac{2g}{a}$ gives

$$\frac{\partial \mathbf{w}}{\partial t} + \mathbf{\Lambda}(\mathbf{u})\frac{\partial \mathbf{w}}{\partial x} = \mathbf{0}\,, \tag{2.12}$$

where $\mathbf{w}$ is the vector of characteristic variables, for which $\frac{\partial \mathbf{w}}{\partial t} \equiv \mathbf{K}(\mathbf{u})^{-1}\frac{\partial \mathbf{u}}{\partial t}$ and $\frac{\partial \mathbf{w}}{\partial x} \equiv \mathbf{K}(\mathbf{u})^{-1}\frac{\partial \mathbf{u}}{\partial x}$, and $\mathbf{\Lambda}(\mathbf{u})$ is a diagonal matrix with the eigenvalues as the diagonal entries:

$$\mathbf{w}(\mathbf{u}) = \begin{bmatrix} u-2a \\ u+2a \end{bmatrix}\,, \quad \mathbf{\Lambda}(\mathbf{u}) = \mathbf{K}(\mathbf{u})^{-1}\mathbf{A}(\mathbf{u})\mathbf{K}(\mathbf{u}) = \begin{bmatrix} \lambda_1(\mathbf{u}) & 0 \\ 0 & \lambda_2(\mathbf{u}) \end{bmatrix}\,. \tag{2.13}$$

The 2-D shallow water equations (with flat bottom topography) can be expressed

as

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot [\mathbf{F}(\mathbf{u}), \mathbf{G}(\mathbf{u})] = \mathbf{0}\,, \quad \mathbf{u} = \begin{bmatrix} h \\ q_x \\ q_y \end{bmatrix}\,, \tag{2.14a}$$

$$\mathbf{F}(\mathbf{u}) = \begin{bmatrix} q_x \\ \frac{q_x^2}{h} + \frac{1}{2}gh^2 \\ \frac{q_x q_y}{h} \end{bmatrix}\,, \quad \mathbf{G}(\mathbf{u}) = \begin{bmatrix} q_y \\ \frac{q_x q_y}{h} \\ \frac{q_y^2}{h} + \frac{1}{2}gh^2 \end{bmatrix}\,. \tag{2.14b}$$

The spatial derivatives of the flux functions can be evaluated using chain rule, giving

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{A}(\mathbf{u})\frac{\partial \mathbf{u}}{\partial x} + \mathbf{B}(\mathbf{u})\frac{\partial \mathbf{u}}{\partial y} = \mathbf{0}\,, \tag{2.15}$$

where $\mathbf{A}(\mathbf{u}) \equiv \frac{\partial \mathbf{F}}{\partial \mathbf{u}}$ is the Jacobian matrix in the x-direction, and $\mathbf{B}(\mathbf{u}) \equiv \frac{\partial \mathbf{G}}{\partial \mathbf{u}}$ is the Jacobian matrix in the y-direction. Unfortunately, these Jacobians cannot be diagonalized simultaneously; instead, here the projection of the flux $[\mathbf{F}(\mathbf{u}), \mathbf{G}(\mathbf{u})]$ in the unit direction $\mathbf{n}$ is considered. The Jacobian of $\mathbf{F}_n(\mathbf{u}) \equiv [\mathbf{F}(\mathbf{u}), \mathbf{G}(\mathbf{u})] \cdot \mathbf{n}$ is denoted by $\mathbf{A}_n(\mathbf{u})$ and is given by

$$\mathbf{A}_n(\mathbf{u}) \equiv \frac{\partial \mathbf{F}_n}{\partial \mathbf{u}} = n_x \mathbf{A}(\mathbf{u}) + n_y \mathbf{B}(\mathbf{u})\,, \tag{2.16a}$$

$$\mathbf{A}_n(\mathbf{u}) \equiv \begin{bmatrix} 0 & n_x & n_y \\ \left(gh - \frac{q_x^2}{h^2}\right) n_x - \frac{q_x q_y}{h^2} n_y & \frac{q_y}{h} n_y + \frac{2q_x}{h} n_x & \frac{q_y}{h} n_y \\ \left(gh - \frac{q_y^2}{h^2}\right) n_y - \frac{q_x q_y}{h^2} n_x & \frac{q_y}{h} n_x & \frac{q_x}{h} n_x + \frac{2q_y}{h} n_y \end{bmatrix}\,. \tag{2.16b}$$

The eigenvalues and corresponding eigenvectors of this matrix are

$$\lambda_1(\mathbf{u}) = \mathbf{v} \cdot \mathbf{n} + a\,, \quad \lambda_2(\mathbf{u}) = \mathbf{v} \cdot \mathbf{n}\,, \quad \lambda_3(\mathbf{u}) = \mathbf{v} \cdot \mathbf{n} - a\,, \tag{2.17}$$

$$\mathbf{k}_1(\mathbf{u}) = \begin{bmatrix} 1 \\ u + an_x \\ v + an_y \end{bmatrix}, \quad \mathbf{k}_2(\mathbf{u}) = \begin{bmatrix} 0 \\ -an_y \\ an_x \end{bmatrix}, \quad \mathbf{k}_3(\mathbf{u}) = \begin{bmatrix} 1 \\ u - an_x \\ v - an_y \end{bmatrix}. \tag{2.18}$$

Forming a matrix $\mathbf{K}(\mathbf{u})$ using the eigenvectors as columns and then taking its inverse gives

$$\mathbf{K}(\mathbf{u}) = \begin{bmatrix} 1 & 0 & 1 \\ u + an_x & -an_y & u - an_x \\ v + an_y & an_x & v - an_y \end{bmatrix}, \tag{2.19a}$$

$$\mathbf{K}(\mathbf{u})^{-1} = \frac{1}{2a} \begin{bmatrix} a - v_n & n_x & n_y \\ 2(n_y u - n_x v) & -2n_y & 2n_x \\ a + v_n & -n_x & -n_y \end{bmatrix}. \tag{2.19b}$$

The characteristic variables $\mathbf{w}$ for the direction $\mathbf{n}$ are then such that $\frac{\partial \mathbf{w}}{\partial t} \equiv \mathbf{K}(\mathbf{u})^{-1} \frac{\partial \mathbf{u}}{\partial t}$ and $\frac{\partial \mathbf{w}}{\partial x} \equiv \mathbf{K}(\mathbf{u})^{-1} \frac{\partial \mathbf{u}}{\partial x}$. This gives the characteristic variables to be

$$\mathbf{w}(\mathbf{u}) = \begin{bmatrix} v_n + 2a \\ n_y u - n_x v \\ v_n - 2a \end{bmatrix}, \tag{2.20}$$

where $v_n \equiv \mathbf{v} \cdot \mathbf{n}$.

2.2 Discretization

2.2.1 Spatial Discretization for Scalar Conservation Laws

The continuous Galerkin (CG) finite element method (FEM) is used for spatial discretization. In this research, linear piecewise polynomials are used to approximate

the solution; formally, the approximation space is the following:

$$\mathcal{U}_h = \left\{ v \in C^0(\mathcal{D}; \mathbb{R});\, v|_K \circ \Phi_K \in \mathbb{Q}_1 \quad \forall K \in \mathcal{K}_h \right\} , \tag{2.21}$$

where Φ_K is a map from the reference element to an element K, and $\mathcal{K}_h$ is the triangulation. When the incoming flux boundary condition of Equation (2.3) is strongly imposed, the approximation space reduces to

$$\mathcal{U}_h^{\text{inc}} = \left\{ v \in \mathcal{U}_h;\, v(\mathbf{x}) = u^{\text{inc}}(\mathbf{x}) \quad \forall \mathbf{x} \in \partial\mathcal{D}^- \right\} , \tag{2.22}$$

where $u^{\text{inc}}(\mathbf{x})$ is the incoming flux function. The approximate solution is an expansion of basis functions $\varphi_j(\mathbf{x})$:

$$\tilde{u}\left(\mathbf{x}, t\right) = \sum_j U_j(t) \varphi_j(\mathbf{x}) \,, \tag{2.23}$$

where the coefficients $U_j(t)$ are the basis function expansion coefficients at time t. Substituting the approximate solution into Equation (2.1) and testing with basis function $\varphi_i(\mathbf{x})$ gives

$$\int\limits_{S_i} \frac{\partial \tilde{u}}{\partial t} \varphi_i(\mathbf{x})\, d\mathbf{x} + \int\limits_{S_i} \left(\nabla \cdot \mathbf{f}(\tilde{u}) + \sigma(\mathbf{x}) \tilde{u}\left(\mathbf{x}, t\right) \right) \varphi_i(\mathbf{x})\, d\mathbf{x} = \int\limits_{S_i} q\left(\mathbf{x}, t\right) \varphi_i(\mathbf{x})\, d\mathbf{x} \,, \tag{2.24}$$

where S_i is the support of $\varphi_i(\mathbf{x})$. If the flux function $\mathbf{f}(u)$ is linear with respect to u, i.e., $\mathbf{f}(u) = \mathbf{v}u$ for some uniform velocity field $\mathbf{v}$, then the system to be solved is linear:

$$\mathbf{M}^C \frac{d\mathbf{U}}{dt} + \mathbf{A}\mathbf{U}(t) = \mathbf{b}(t) \,, \tag{2.25}$$

with the elements of $\mathbf{A}$ being the following:

$$A_{i,j} \equiv \int_{S_{i,j}} \left(\mathbf{v} \cdot \nabla \varphi_j(\mathbf{x}) + \sigma(\mathbf{x}) \varphi_j(\mathbf{x})\right) \varphi_i(\mathbf{x}) \, d\mathbf{x} \,, \tag{2.26}$$

where $S_{i,j}$ is the shared support of $\varphi_i(\mathbf{x})$ and $\varphi_j(\mathbf{x})$. Figure 2.1 illustrates an example of this definition. If the flux function $\mathbf{f}(u)$ is nonlinear, then the system is nonlinear,

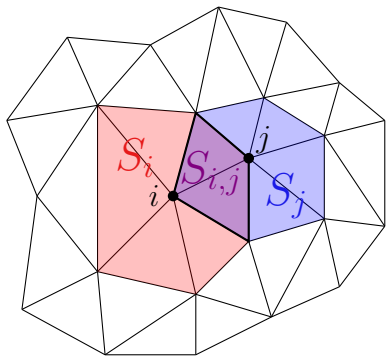


Figure 2.1: Illustration of Shared Support between Test Functions

but it may be expressed in a quasi-linear form:

$$\mathbf{M}^C \frac{d\mathbf{U}}{dt} + \mathbf{A}(\tilde{u})\mathbf{U}(t) = \mathbf{b}(t) \,, \tag{2.27}$$

where $\tilde{u}\,(\mathbf{x}, t)$ is the numerical solution, and the quasi-linear matrix (i.e., the Jacobian matrix) entries are

$$A_{i,j}(\tilde{u}) \equiv \int_{S_{i,j}} \left(\mathbf{f}'(\tilde{u}) \cdot \nabla \varphi_j(\mathbf{x}) + \sigma(\mathbf{x}) \varphi_j(\mathbf{x})\right) \varphi_i(\mathbf{x}) \, d\mathbf{x} \,. \tag{2.28}$$

The elements of $\mathbf{b}(t)$ are

$$b_i(t) \equiv \int_{S_i} q\,(\mathbf{x}, t)\,\varphi_i(\mathbf{x})\,d\mathbf{x}\,. \tag{2.29}$$

$\mathbf{M}^C$ is the consistent mass matrix, which has the entries

$$M^C_{i,j} \equiv \int_{S_{i,j}} \varphi_j(\mathbf{x})\varphi_i(\mathbf{x})\,d\mathbf{x}\,. \tag{2.30}$$

Similarly, for the linear steady-state case, the linear system is

$$\mathbf{AU} = \mathbf{b}\,, \tag{2.31}$$

or for the nonlinear case,

$$\mathbf{A}(\tilde{u})\mathbf{U} = \mathbf{b}\,. \tag{2.32}$$

2.2.1.1 Implementation of Incoming Flux Boundary Conditions

In this section, some options for the implementation of the incoming flux boundary condition, given by Equation (2.3), are presented. This research considers three options for this task:

1. Strongly impose the Dirichlet boundary conditions,
2. Weakly impose the Dirichlet boundary conditions, and
3. Weakly impose the Dirichlet boundary conditions with the boundary penalty method.

Most of the scalar results in this dissertation will use strongly imposed Dirichlet boundary conditions, but for some cases, the other two approaches are considered.

Consider the linear system $\mathbf{BU} = \mathbf{r}$. To strongly impose Dirichlet boundary conditions on this linear system, the system matrix and system right-hand-side vector are directly modified:

$$\tilde{\mathbf{B}}\mathbf{U} = \tilde{\mathbf{r}}\,, \tag{2.33a}$$

$$\tilde{B}_{i,j} = \begin{cases} \begin{cases} \alpha_i & j = i \\ 0 & j \neq i \end{cases} & i \in \mathcal{I}^{\text{inc}} \\ B_{i,j} & i \notin \mathcal{I}^{\text{inc}} \end{cases}, \tag{2.33b}$$

$$\tilde{r}_i = \begin{cases} \alpha_i u_i^{\text{inc}} & i \in \mathcal{I}^{\text{inc}} \\ r_i & i \notin \mathcal{I}^{\text{inc}} \end{cases}, \tag{2.33c}$$

where α_i is some positive value, u_i^{inc} is the incoming value to be imposed, and $\mathcal{I}^{\text{inc}}$ is the set of indices of degrees of freedom (DoF) for which incoming boundary conditions apply. To summarize, for an incoming DoF i, the row i of the system matrix $\mathbf{B}$ is zeroed, except for the diagonal value $\tilde{B}_{i,i}$, which is set to some positive value α_i. The right-hand-side value r_i is then replaced with $\tilde{r}_i = \alpha u_i^{\text{inc}}$. In this way, the ith equation is replaced with the equation $\alpha_i U_i = \alpha_i u_i^{\text{inc}}$. Optionally, one may also eliminate all of the off-diagonal entries corresponding to i by bringing them over to the right-hand-side:

$$\tilde{B}_{i,j} = \begin{cases} \begin{cases} \alpha_i & j = i \\ 0 & j \neq i \\ 0 & j \in \mathcal{I}^{\text{inc}} \end{cases} & i \in \mathcal{I}^{\text{inc}} \\ \begin{cases} B_{i,j} & j \notin \mathcal{I}^{\text{inc}} \end{cases} & i \notin \mathcal{I}^{\text{inc}} \end{cases}. \tag{2.34a}$$

$$\tilde{r}_i = \begin{cases} \alpha_i u_i^{\text{inc}} & i \in \mathcal{I}^{\text{inc}} \\ r_i - \sum_{j \in \mathcal{I}^{\text{inc}}} B_{i,j} u_j^{\text{inc}} & i \notin \mathcal{I}^{\text{inc}} \end{cases} . \tag{2.34b}$$

This approach has the advantage of preserving symmetry of the original matrix $\mathbf{B}$.

To weakly impose the incoming boundary conditions for a degree of freedom $i \in \mathcal{I}^{\text{inc}}$, one starts by integrating the advection term in Equation (2.26) by parts:

$$\int_{S_{i,j}} \varphi_i \mathbf{v} \cdot \nabla \varphi_j \, d\mathbf{x} = -\int_{S_{i,j}} \varphi_j \mathbf{v} \cdot \nabla \varphi_i \, d\mathbf{x} + \int_{\partial S_{i,j}} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n} dA \,, \tag{2.35}$$

$$= -\int_{S_{i,j}} \varphi_j \mathbf{v} \cdot \nabla \varphi_i \, d\mathbf{x} + \sum_{F \subset \partial S_{i,j}} \int_F \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n} dA \,, \tag{2.36}$$

where $\partial S_{i,j}$ denotes the boundary of the shared support $S_{i,j}$. The interior face terms cancel with their skew-symmetric counterparts and can thus be ignored, leaving only face terms for exterior faces. Let $\partial S_{i,j}^{\text{ext}}$ denote the portion of the boundary of the shared support $\partial S_{i,j}$ that lies on the external boundary of the triangulation: $\partial S_{i,j}^{\text{ext}} = \partial S_{i,j} \cap \partial \mathcal{K}_h$. Then the advection term above can be expressed as

$$\int_{S_{i,j}} \varphi_i \mathbf{v} \cdot \nabla \varphi_j \, d\mathbf{x} = -\int_{S_{i,j}} \varphi_j \mathbf{v} \cdot \nabla \varphi_i \, d\mathbf{x} + \sum_{F \subset \partial S_{i,j}^{\text{ext}}} \int_F \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n} dA \,. \tag{2.37}$$

Furthermore, let $\partial S_{i,j}^-$ denote the portion of $\partial S_{i,j}$ on the *incoming* portion of the boundary of the triangulation and $\partial S_{i,j}^+$ denote the portion on the *outgoing* portion:

$$\partial S_{i,j}^- \equiv \{\mathbf{x} \in \partial S_{i,j}^{\text{ext}} \, : \, \mathbf{v} \cdot \mathbf{n}(\mathbf{x}) < 0\} \,, \tag{2.38}$$

$$\partial S_{i,j}^+ \equiv \{\mathbf{x} \in \partial S_{i,j}^{\text{ext}} \, : \, \mathbf{v} \cdot \mathbf{n}(\mathbf{x}) > 0\} \,. \tag{2.39}$$

Then Equation (2.37) becomes

$$\int_{S_{i,j}} \varphi_i \mathbf{v} \cdot \nabla \varphi_j \, d\mathbf{x} = -\int_{S_{i,j}} \varphi_j \mathbf{v} \cdot \nabla \varphi_i \, d\mathbf{x} + \sum_{F^- \subset \partial S^-_{i,j}} \int_{F^-} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n}_{F^-} dA + \sum_{F^+ \subset \partial S^+_{i,j}} \int_{F^+} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n}_{F^+} dA \,. \quad (2.40)$$

The complete expression for $A_{i,j}$ is then

$$A_{i,j} = \int_{S_{i,j}} \left(-\varphi_j \mathbf{v} \cdot \nabla \varphi_i + \sigma \varphi_j \varphi_i\right) d\mathbf{x} + \sum_{F^- \subset \partial S^-_{i,j}} \int_{F^-} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n}_{F^-} dA + \sum_{F^+ \subset \partial S^+_{i,j}} \int_{F^+} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n}_{F^+} dA \quad (2.41)$$

One then brings the incoming boundary terms to the right-hand-side, multiplied by the boundary data evaluated at $\mathbf{x}_j$. In summary, the modified steady-state matrix and steady-state right-hand-side vector are

$$\tilde{A}_{i,j} = \int_{S_{i,j}} \left(-\varphi_j \mathbf{v} \cdot \nabla \varphi_i + \sigma \varphi_j \varphi_i\right) d\mathbf{x} + \sum_{F^+ \subset \partial S^+_{i,j}} \int_{F^+} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n}_{F^+} dA \,, \quad (2.42a)$$

$$\tilde{b}_i = b_i - \sum_j \left(\sum_{F^- \subset \partial S^-_{i,j}} \int_{F^-} \varphi_j \varphi_i \mathbf{v} \cdot \mathbf{n}_{F^-} dA \right) u_j^{\text{inc}} \,, \quad (2.42b)$$

where u_j^{inc} denotes $u^{\text{inc}}(\mathbf{x}_j)$. Note that since $\mathbf{v} \cdot \mathbf{n} < 0$ for incoming boundaries and the boundary data u^{inc} is assumed to be non-negative, $\tilde{A}_{i,j} \geq A_{i,j}$ and $\tilde{b}_i \geq b_i$.

For the boundary penalty method, to impose a Dirichlet boundary condition for degree of freedom i, instead of replacing equation i with $\alpha_i U_i = \alpha_i u_i^{\text{inc}}$, as is done with strongly imposed Dirichlet boundary conditions, this equation is *added* to the

existing equation. The strength of the imposition of the boundary condition increases with increasing α_i. The modified system matrix and right-hand-side vector are the following:

$$\tilde{B}_{i,j} = \begin{cases} \begin{cases} B_{i,i} + \alpha_i & j = i \\ B_{i,j} & j \neq i \end{cases} & i \in \mathcal{I}^{\text{inc}} \\ B_{i,j} & i \notin \mathcal{I}^{\text{inc}} \end{cases}, \tag{2.43a}$$

$$\tilde{r}_i = \begin{cases} r_i + \alpha_i u_i^{\text{inc}} & i \in \mathcal{I}^{\text{inc}} \\ r_i & i \notin \mathcal{I}^{\text{inc}} \end{cases}. \tag{2.43b}$$

This approach may be used with or without applying weak Dirichlet boundary conditions.

2.2.2 Spatial Discretization for Conservation Law Systems

This section gives the spatial discretization for conservation law systems. Again, CGFEM is used for spatial discretization, so the numerical solution (now vector-valued: $\tilde{\mathbf{u}}$) is again approximated using an expansion of basis functions. However, since there are now multiple degrees of freedom at each node, some discussion on notation is given here to distinguish between node indexing and degree of freedom indexing. First, let the number of *scalar* solution components be denoted by m. Thus, as an example, the 2-D shallow water equations, which consist of a continuity equation and a conservation of momentum equation, have $m = 3$ because the multi-dimensional conservation of momentum equation is comprised of two scalar conservation of momentum equations. There are a number of ways to view the basis function expansion of the approximate solution. For example, one may use

vector-valued basis functions:

$$\tilde{\mathbf{u}}\left(\mathbf{x},t\right)=\sum_{j=1}^{N_{\text{dof}}}U_j(t)\boldsymbol{\Phi}_j(\mathbf{x})\,, \tag{2.44}$$

where the coefficients $U_j(t)$ are the basis function expansion coefficients at time t, and $\boldsymbol{\Phi}_j(\mathbf{x})$ are the vector-valued basis functions

$$\boldsymbol{\Phi}_j(\mathbf{x})=\hat{\mathbf{e}}_{k(j)}\varphi_{k(j)}(\mathbf{x})\,, \tag{2.45}$$

where $k(j)$ returns the component index associated with degree of freedom j, and $k(j)$ returns the node index associated with degree of freedom j, and $\hat{\mathbf{e}}_{k(j)}$ is the unit vector for dimension $k(j)$. The basis function $\boldsymbol{\Phi}_j$ is thus zero for all components except the one corresponding to j which uses the scalar component basis function at node $k(j)$, $\varphi_{k(j)}(\mathbf{x})$. One may view each solution component as its own expansion:

$$\tilde{u}^k\left(\mathbf{x},t\right)=\sum_{k=1}^{N_{\text{node}}}U_{j(k,k)}(t)\varphi_k(\mathbf{x})\,, \tag{2.46}$$

where $j(k,k)$ is the global degree of freedom index associated with node k and solution component k.

Remark Alternatively, one may consider vector-valued degrees of freedom $\mathbf{U}_j(t)$ with scalar test functions:

$$\tilde{\mathbf{u}}\left(\mathbf{x},t\right)=\sum_{j=1}^{N_{\text{node}}}\mathbf{U}_j(t)\varphi_j(\mathbf{x})\,. \tag{2.47}$$

This is often more convenient in describing schemes and theory; however, in a typical implementation, the former view is used since the solution vector is typically stored as

a single vector of scalars instead of a vector of vectors. When there is doubt regarding which view is used in this dissertation, the view used will be stated explicitly.

As opposed to the scalar conservation law case, the vector case interpolates the conservation law flux between nodal values:

$$\mathbf{F}(\mathbf{u}\,(\mathbf{x},t)) \rightarrow \Pi\mathbf{F}(\mathbf{u}\,(\mathbf{x},t)) \equiv \sum_j \mathbf{\Phi}_j(\mathbf{x})\mathbf{f}(\mathbf{u}(\mathbf{x}_{k(j)},t))\,, \tag{2.48}$$

where hereafter the nodal flux values used as interpolation values, $\mathbf{f}(\mathbf{u}(\mathbf{x}_{k(j)},t))$, will be denoted as $\mathrm{F}_j(t)$. Substituting the approximate solution into the general conservation law system equation,

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{u}) = \mathbf{0}\,, \tag{2.49}$$

and testing with basis function $\mathbf{\Phi}_i(\mathbf{x})$ gives

$$\sum_j \int\limits_{S_{i,j}} \mathbf{\Phi}_i^{\mathrm{T}}(\mathbf{x})\mathbf{\Phi}_j(\mathbf{x})\,d\mathbf{x}\frac{dU_j}{dt} + \sum_j \int\limits_{S_{i,j}} \mathbf{\Phi}_i^{\mathrm{T}}(\mathbf{x})\nabla\mathbf{\Phi}_j(\mathbf{x})\,d\mathbf{x}\cdot \mathrm{F}_j(t) = \mathbf{0}\,. \tag{2.50}$$

Again, a mass matrix $\mathbf{M}^C$ is defined:

$$M^C_{i,j} \equiv \int\limits_{S_{i,j}} \mathbf{\Phi}_i^{\mathrm{T}}(\mathbf{x})\mathbf{\Phi}_j(\mathbf{x})\,d\mathbf{x} \tag{2.51}$$

as well a 3rd-order tensor $\mathbf{C}$, which will here be viewed as a matrix with vector entries:

$$\mathbf{c}_{i,j} \equiv \int\limits_{S_{i,j}} \mathbf{\Phi}_i^{\mathrm{T}}(\mathbf{x})\nabla\mathbf{\Phi}_j(\mathbf{x})\,d\mathbf{x}\,. \tag{2.52}$$

Making these substitutions into Equation (2.50) gives

$$\sum_j M^C_{i,j} \frac{dU_j}{dt} + \sum_j \mathbf{c}_{i,j} \cdot \mathrm{F}_j(t) = \mathbf{0}\,, \tag{2.53}$$

and expressing this as a system gives

$$\mathbf{M}^C \frac{d\mathbf{U}}{dt} + \mathbf{C}\mathrm{F}(t) = \mathbf{0}\,. \tag{2.54}$$

where $\mathrm{F}(t)$ is the vector of nodal flux interpolant values $\mathrm{F}_j(t)$.

2.2.2.1 The Shallow Water Equations

Rearranging the continuity equation, substituting the approximate FEM solution and testing with a test function φ_i^h gives its weak form for degree of freedom i:

$$\int_{\mathcal{D}} \varphi_i^h \frac{\partial \tilde{h}}{\partial t}\, d\mathbf{x} = -\int_{\mathcal{D}} \varphi_i^h \nabla \cdot \tilde{\mathbf{q}}\, d\mathbf{x}\,. \tag{2.55}$$

Integrating by parts gives

$$\int_{\mathcal{D}} \varphi_i^h \frac{\partial \tilde{h}}{\partial t}\, d\mathbf{x} = \int_{\mathcal{D}} \nabla \varphi_i^h \cdot \tilde{\mathbf{q}}\, d\mathbf{x} - \int_{\partial\mathcal{D}} \varphi_i^h \tilde{\mathbf{q}} \cdot \mathbf{n}\, dA\,. \tag{2.56}$$

Rearranging the momentum equation for the d direction, substituting the approximate FEM solution, and testing with a test function $\varphi_i^{q_d}$ gives its weak form for degree of freedom i:

$$\int_{\mathcal{D}} \varphi_i^{q_d} \frac{\partial \tilde{q}_d}{\partial t}\, d\mathbf{x} = -\int_{\mathcal{D}} \varphi_i^{q_d} \nabla \cdot \left(\frac{\tilde{q}_d}{h} \tilde{\mathbf{q}} + \frac{1}{2} g \tilde{h}^2 \hat{\mathbf{e}}_d \right)\, d\mathbf{x} - \int_{\mathcal{D}} \varphi_i^{q_d} g \tilde{h} \frac{\partial b}{\partial x_d}\, d\mathbf{x}\,. \tag{2.57}$$

Integrating by parts for the flux terms gives

$$\int_{\mathcal{D}} \varphi_i^{q_d} \frac{\partial \tilde{q}_d}{\partial t}\, d\mathbf{x} = + \int_{\mathcal{D}} \nabla \varphi_i^{q_d} \cdot \left(\frac{\tilde{q}_d}{h} \tilde{\mathbf{q}} + \frac{1}{2} g \tilde{h}^2 \hat{\mathbf{e}}_d \right)\, d\mathbf{x} - \int_{\partial \mathcal{D}} \varphi_i^{q_d} \left(\frac{\tilde{q}_d}{h} \tilde{\mathbf{q}} + \frac{1}{2} g \tilde{h}^2 \hat{\mathbf{e}}_d \right) \cdot \mathbf{n}\, dA - \int_{\mathcal{D}} \varphi_i^{q_d} g \tilde{h} \frac{\partial b}{\partial x_d}\, d\mathbf{x}\,. \quad (2.58)$$

In a more compact vector format, the momentum equations may be expressed as

$$\int_{\mathcal{D}} \varphi_i^{\mathbf{q}} \cdot \frac{\partial \tilde{\mathbf{q}}}{\partial t}\, d\mathbf{x} = \int_{\mathcal{D}} \nabla \varphi_i^{\mathbf{q}} : \left(\tilde{\mathbf{q}} \otimes \tilde{\mathbf{v}} + \frac{1}{2} g \tilde{h}^2 \mathbf{I} \right)\, d\mathbf{x} - \int_{\partial \mathcal{D}} \varphi_i^{\mathbf{q}} \cdot \left(\tilde{\mathbf{q}} \otimes \tilde{\mathbf{v}} + \frac{1}{2} g \tilde{h}^2 \mathbf{I} \right) \cdot \mathbf{n}\, dA - \int_{\mathcal{D}} \varphi_i^{\mathbf{q}} \cdot g \tilde{h} \nabla b\, d\mathbf{x}\,. \quad (2.59)$$

This yields a discrete system

$$\mathbf{M}^C \frac{d\mathbf{U}}{dt} = \mathbf{r}\,, \quad (2.60)$$

where $\mathbf{M}^C$ is the mass matrix and the steady-state residual $\mathbf{r}$ is given by

$$r_i = \int_{\mathcal{D}} \nabla \varphi_i^h \cdot \tilde{\mathbf{q}}\, d\mathbf{x} - \int_{\partial \mathcal{D}} \varphi_i^h \tilde{\mathbf{q}} \cdot \mathbf{n}\, dA + \int_{\mathcal{D}} \nabla \varphi_i^{\mathbf{q}} : \left(\tilde{\mathbf{q}} \otimes \tilde{\mathbf{v}} + \frac{1}{2} g \tilde{h}^2 \mathbf{I} \right)\, d\mathbf{x} - \int_{\partial \mathcal{D}} \varphi_i^{\mathbf{q}} \cdot \left(\tilde{\mathbf{q}} \otimes \tilde{\mathbf{v}} + \frac{1}{2} g \tilde{h}^2 \mathbf{I} \right) \cdot \mathbf{n}\, dA - \int_{\mathcal{D}} \varphi_i^{\mathbf{q}} \cdot g \tilde{h} \nabla b\, d\mathbf{x} \quad (2.61)$$

2.2.3 Temporal Discretization

2.2.3.1 Explicit Euler Scheme

Considering a time step from time t^n to time t^{n+1} with time step size $\Delta t \equiv t^{n+1} - t^n$, the semi-discrete equation given by Equation (2.25) is discretized using the

explicit Euler:

$$\mathbf{M}^C \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} + \mathbf{A}\mathbf{U}^n = \mathbf{b}^n, \tag{2.62}$$

where $\mathbf{U}^{n+1}$ is the Galerkin solution at time t^{n+1}.

2.2.3.2 Strong Stability-Preserving Runge-Kutta (SSPRK) Schemes

Strong Stability-Preserving Runge Kutta (SSPRK) methods [10][23] are a subclass of Runge Kutta methods that offer high-order accuracy while preserving stability. Suppose that one wants to integrate the following ODE system:

$$\frac{d\mathbf{U}}{dt} = \mathbf{G}(t, \mathbf{U}(t)) \,. \tag{2.63}$$

For example, the function $\mathbf{G}(t, \mathbf{U}(t))$ corresponding to Equation (2.25) is

$$\mathbf{G}(t, \mathbf{U}(t)) = (\mathbf{M}^C)^{-1} \left(\mathbf{b}(t) - \mathbf{A}\mathbf{U}(t)\right) \,. \tag{2.64}$$

In α-β notation, SSPRK methods can be described by the following steps:

$$\hat{\mathbf{U}}^0 = \mathbf{U}^n \,, \tag{2.65a}$$

$$\hat{\mathbf{U}}^i = \sum_{j=1}^{i-1} \left[\alpha_{i,j}\hat{\mathbf{U}}^{j-1} + \Delta t \beta_{i,j} \mathbf{G}(t^n + c_j \Delta t, \hat{\mathbf{U}}^{j-1})\right] \,, \quad i = 1, \ldots, s \,, \tag{2.65b}$$

$$\mathbf{U}^{n+1} = \hat{\mathbf{U}}^s \,, \tag{2.65c}$$

where s is the number of stages of the method, and α, β, and c are coefficient arrays corresponding to the particular SSPRK method. Results in this dissertation use two SSPRK methods: the forward (explicit) Euler method, and the 3-stage, 3rd-order-accurate Shu-Osher scheme, hereafter referred to as SSPRK33. The SSPRK33

method has the following coefficients:

$$\alpha = \begin{bmatrix} 1 & & \\ \frac{3}{4} & \frac{1}{4} & \\ \frac{1}{3} & 0 & \frac{2}{3} \end{bmatrix}, \quad \beta = \begin{bmatrix} 1 & & \\ 0 & \frac{1}{4} & \\ 0 & 0 & \frac{2}{3} \end{bmatrix}, \quad c = \begin{bmatrix} 0 \\ 1 \\ \frac{1}{2} \end{bmatrix}. \tag{2.66}$$

Alternatively, the SSPRK methods used in this dissertation can be expressed in the form

$$\hat{\mathbf{U}}^0 = \mathbf{U}^n \,, \tag{2.67a}$$

$$\hat{\mathbf{U}}^i = \gamma_i \mathbf{U}^n + \zeta_i \left[\hat{\mathbf{U}}^{i-1} + \Delta t \mathbf{G}(t^n + c_i \Delta t, \hat{\mathbf{U}}^{i-1}) \right], \quad i = 1, \ldots, s \,, \tag{2.67b}$$

$$\mathbf{U}^{n+1} = \hat{\mathbf{U}}^s \,. \tag{2.67c}$$

This form makes it clear that these methods can be expressed as a linear combination of steps resembling a forward Euler step, except that quantities with explicit time dependence are not always evaluated at the time corresponding to the beginning of the step. This allows the methodology developed for forward Euler in this research to be directly extended to these SSPRK methods. This includes, for example, the FCT methodology described in Sections 2.7 and 2.8. In the γ-ζ notation, the coefficient arrays for SSPRK33 are

$$\gamma = \begin{bmatrix} 0 \\ \frac{3}{4} \\ \frac{1}{3} \end{bmatrix}, \quad \zeta = \begin{bmatrix} 1 \\ \frac{1}{4} \\ \frac{2}{3} \end{bmatrix}, \quad c = \begin{bmatrix} 0 \\ 1 \\ \frac{1}{2} \end{bmatrix}. \tag{2.68}$$

2.2.3.3 Theta Schemes

Discretizing Equation (2.25) with the θ scheme gives

$$\mathbf{M}^C \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}\mathbf{U}^n + \theta\mathbf{A}\mathbf{U}^{n+1} = (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1}. \tag{2.69}$$

Note that $\theta = 0$ corresponds to the explicit Euler method, $\theta = 1$ corresponds to the implicit Euler method, and $\theta = \frac{1}{2}$ corresponds to the Crank-Nicolson method.

2.3 Low-Order Scheme for Scalar Conservation Laws

In this section, a first-order, positivity-preserving scheme for scalar conservation laws will be described. This scheme is adapted from Guermond [11], in which a general scalar conservation law of the form

$$\frac{\partial u}{\partial t} + \nabla \cdot \mathbf{f}(u) = 0\,, \tag{2.70}$$

is considered; this section additionally considers the presence of a reaction term and source term as given by Equation (2.1). This scheme will be shown to preserve non-negativity of the solution and to satisfy a local discrete maximum principle (DMP). DMP-satisfaction is a key property in the prevention of spurious oscillations.

This section is organized as follows: Section 2.3.1 makes some definitions that are necessary to define the low-order scheme, including a low-order artificial viscosity and a local graph-theoretic viscous bilinear form. Section 2.3.2 defines the low-order system in a number of temporal discretizations. Section 2.3.3 gives a theoretical proof that the definitions of the low-order scheme yield an inverse-positive system matrix, thus proving positivity-preservation. Finally, Section 2.3.5 lists and proves local DMPs for each considered temporal discretization.

2.3.1 Low-Order Viscosity

In this section, definitions of a graph-theoretic local viscous bilinear form and a low-order viscosity from Guermond [11] are given. First, some preliminary definitions are given.

Let $\mathcal{I}_K$ denote the set of degree of freedom indices associated with cell K, which is defined to be those degrees of freedom j for which the corresponding test function φ_j has nonzero support on cell K:

$$\mathcal{I}_K \equiv \{j \in \{1, \ldots, N\} : |S_j \cap \mathcal{D}_K| \neq 0\}, \tag{2.71}$$

where $\mathcal{D}_K$ is the domain of cell K. An illustration of this definition is given in Figure 2.2. Let n_K denote the number of elements in the set $\mathcal{I}_K$; for example, in Figure 2.2, $n_K = 3$. Let $\mathcal{K}(S_{i,j})$ denote the set of cell indices corresponding to cells that lie in

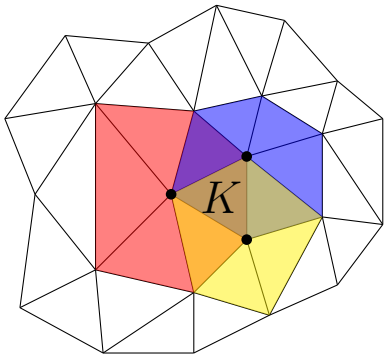


Figure 2.2: Illustration of Cell Degree of Freedom Indices $\mathcal{I}_K$

the shared support $S_{i,j}$:

$$\mathcal{K}(S_{i,j}) \equiv \{K : \mathcal{D}_K \subset S_{i,j}\}. \tag{2.72}$$

For example, in Figure 2.1, $\mathcal{K}(S_{i,j})$ would consist of the indices of the two cells in

$S_{i,j}$.

The following graph-theoretic local viscous bilinear form from [11] is employed in computation of the artificial diffusion terms, which are expressed in matrix form in Section 2.3.2:

Definition 2.3.1 (Local Viscous Bilinear Form) *The local viscous bilinear form for cell K is defined as follows:*

$$b_K(\varphi_j, \varphi_i) \equiv \begin{cases} -\frac{1}{n_K - 1}|\mathcal{D}_K|\,, & i \neq j\,, \quad i,j \in \mathcal{I}_K\,, \\ |\mathcal{D}_K|\,, & i = j\,, \quad i,j \in \mathcal{I}_K\,, \quad , \\ 0\,, & \textit{otherwise}\,, \end{cases} \tag{2.73}$$

with $|\mathcal{D}_K|$ defined as the volume of cell K.

Note some properties of this definition: the diagonal entries $b_K(\varphi_i, \varphi_i)$ are positive, the off-diagonal entries are negative, and the row-sum $\sum_j b_K(\varphi_i, \varphi_j)$ is zero. The signs of the entries are important in Section 2.3.3, where this knowledge is invoked in the proof of inverse-positivity of the system matrix. The zero row-sum is important in proving that the method is conservative, and it is also used when defining antidiffusive fluxes in the FCT scheme in Section 2.7.1; specifically, it allows the antidiffusive source for a node i to be decomposed into skew-symmetric antidiffusive fluxes between adjacent nodes.

The definition of the low-order viscosity, also taken from [11], follows. The resulting piecewise viscosity is constant over each cell. This definition is designed to introduce the smallest amount of artificial diffusion possible such that the inverse-positivity of the system matrix can be guaranteed; specifically, this definition allows Lemma 2.3.1 in Section 2.3.3 to be proven.

Definition 2.3.2 (Low-Order Viscosity) *The low-order viscosity for cell K is*

defined as follows:

$$\nu_K^{L,n} \equiv \max_{i \neq j \in \mathcal{I}_K} \frac{\max(0, A_{i,j}^n)}{-\sum\limits_{T \in \mathcal{K}(S_{i,j})} b_T(\varphi_j, \varphi_i)}\,, \tag{2.74}$$

where $A_{i,j}^n$ is the i,jth entry of the Galerkin steady-state matrix given by Equation (2.26).

2.3.2 Low-Order System

This section gives the low-order system for each of the considered temporal discretizations. First, a low-order artificial diffusion matrix is defined, which is built upon the definitions of the local viscous bilinear form and low-order viscosity defined in Section (2.3.1).

Definition 2.3.3 (Low-Order Artificial Diffusion Matrix) *The low-order artificial diffusion matrix $\mathbf{D}^{L,n}$ is assembled using the low-order viscosity and local viscous bilinear form introduced in Section 2.3.1:*

$$D_{i,j}^{L,n} \equiv \sum_{K \in \mathcal{K}(S_{i,j})} \nu_K^{L,n} b_K(\varphi_j, \varphi_i)\,. \tag{2.75}$$

The low-order system matrix incorporates the low-order artificial diffusion matrix and is given in the following definition.

Definition 2.3.4 (Low-Order Steady-State System Matrix) *The low-order steady-state system matrix is the sum of the inviscid steady-state system matrix $\mathbf{A}^n$ and the low-order diffusion matrix $\mathbf{D}^{L,n}$:*

$$\mathbf{A}^{L,n} \equiv \mathbf{A}^n + \mathbf{D}^{L,n}\,. \tag{2.76}$$

For the low-order system, the mass matrix is lumped: $\mathbf{M}^C \rightarrow \mathbf{M}^L$, where

$$M_{i,j}^L = \begin{cases} \sum_k M_{i,k}^C\,, & j = i \\ 0\,, & \text{otherwise} \end{cases}\,, \tag{2.77}$$

and the low-order steady-state system matrix defined in Equation (2.76) is used. The low-order system is given here for different time discretizations:

Steady-state scheme:

$$\mathbf{A}^L\mathbf{U}^L = \mathbf{b} \tag{2.78}$$

Semi-discrete scheme:

$$\mathbf{M}^L\frac{d\mathbf{U}^L}{dt} + \mathbf{A}^L(t)\mathbf{U}^L(t) = \mathbf{b}(t) \tag{2.79}$$

Explicit Euler scheme:

$$\mathbf{M}^L\frac{\mathbf{U}^L - \mathbf{U}^n}{\Delta t} + \mathbf{A}^{L,n}\mathbf{U}^n = \mathbf{b}^n \tag{2.80}$$

Theta scheme:

$$\mathbf{M}^L\frac{\mathbf{U}^L - \mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}^{L,n}\mathbf{U}^n + \theta\mathbf{A}^{L,n+1}\mathbf{U}^L = (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1} \tag{2.81}$$

Section 2.3.5 will prove that each of the fully discrete schemes above satisfy a local discrete maximum principle.

2.3.3 M-Matrix Property

In this section, it will be shown that the low-order steady-state system matrix $\mathbf{A}^L$ defined in Equation (2.76) is an M-matrix, also called an inverse-positive matrix or a monotone matrix. An inverse-positive matrix $\mathbf{A}$ has the property

$$\mathbf{A}\mathbf{x} \geq \mathbf{0} \Rightarrow \mathbf{x} \geq \mathbf{0}\,. \tag{2.82}$$

Thus if one has a linear system $\mathbf{Ax} = \mathbf{b}$ with $\mathbf{b} \geq \mathbf{0}$, then the solution $\mathbf{x}$ is known to be non-negative. This property is used in Section 2.3.4 to prove the positivity-preservation of the low-order scheme in each temporal discretization.

The non-positivity off-diagonal elements of $\mathbf{A}^L$ and positivity of its diagonal elements are proven in the following lemmas. These properties are used in the proof that $\mathbf{A}^L$ is an M-matrix, and are also used in Section 2.3.5 to prove local discrete maximum principles for each temporal discretization of the low-order scheme.

Lemma 2.3.1 (Non-Positivity of Off-Diagonal Elements) *The off-diagonal elements of the matrix* $\mathbf{A}^{L,n}$ *are non-positive:*

$$A_{i,j}^{L,n} \leq 0\,, \quad j \neq i\,, \quad \forall i\,.$$

Proof This proof begins by bounding the term $D_{i,j}^{L,n}$:

$$\begin{aligned} D_{i,j}^{L,n} &= \sum_{K \in \mathcal{K}(S_{i,j})} \nu_K^{L,n} b_K(\varphi_j, \varphi_i) \\ &= \sum_{K \in \mathcal{K}(S_{i,j})} \max_{k \neq \ell \in \mathcal{I}_K} \left(\frac{\max(0, A_{k,\ell}^n)}{-\sum\limits_{T \in \mathcal{K}(S_{k,\ell})} b_T(\varphi_\ell, \varphi_k)} \right) b_K(\varphi_j, \varphi_i)\,. \end{aligned}$$

Recall $b_K(\varphi_j, \varphi_i) < 0$ for $j \neq i$. For an arbitrary quantity $c_{k,\ell} \geq 0\,, \forall k \neq \ell \in \mathcal{I}$, the following is true for $i \neq j \in \mathcal{I}$: $\max\limits_{k \neq \ell \in \mathcal{I}} c_{k,\ell} \geq c_{i,j}$, and thus for $a \leq 0$, $a \max\limits_{k \neq \ell \in \mathcal{I}} c_{k,\ell} \leq a c_{i,j}$.

Thus,

$$\begin{aligned} D_{i,j}^{L,n} &\leq \sum_{K\in\mathcal{K}(S_{i,j})} \frac{\max(0, A_{i,j}^n)}{-\sum\limits_{T\in\mathcal{K}(S_{i,j})} b_T(\varphi_j,\varphi_i)} b_K(\varphi_j,\varphi_i)\,, \\ &= -\max(0, A_{i,j}^n) \frac{\sum\limits_{K\in\mathcal{K}(S_{i,j})} b_K(\varphi_j,\varphi_i)}{\sum\limits_{T\in\mathcal{K}(S_{i,j})} b_T(\varphi_j,\varphi_i)}\,, \\ &= -\max(0, A_{i,j}^n)\,, \\ &\leq -A_{i,j}^n\,. \end{aligned}$$

Applying this inequality to Equation (2.76) gives

$$\begin{aligned} A_{i,j}^{L,n} &= A_{i,j}^n + D_{i,j}^{L,n}\,, \\ A_{i,j}^{L,n} &\leq A_{i,j}^n - A_{i,j}^n\,, \\ A_{i,j}^{L,n} &\leq 0\,. \quad \blacksquare \end{aligned}$$

Remark If boundary conditions are weakly imposed, as discussed in Section 2.2.1.1, then the steady-state system matrix is modified and $\tilde{A}_{i,j} \geq A_{i,j}$. In this case, the low-order viscosity is computed with the *modified* steady-state matrix. Then Lemma 2.3.1 still holds.

Lemma 2.3.2 (Non-Negativity of Row Sums) *The row-sums of the matrix* $\mathbf{A}^{L,n}$ *are non-negative:*

$$\sum_j A_{i,j}^{L,n} \geq 0\,, \quad \forall i\,.$$

Proof Using the fact that $\sum_j \varphi_j(\mathbf{x}) = 1$ and $\sum_j b_K(\varphi_j, \varphi_i) = 0$,

$$\begin{aligned}
\sum_j A_{i,j}^{L,n} &= \sum_j \int_{S_{i,j}} \left(\mathbf{f}'(\tilde{u}^n) \cdot \nabla\varphi_j + \sigma\varphi_j\right) \varphi_i \, d\mathbf{x} + \sum_j \sum_{K \in \mathcal{K}(S_{i,j})} \nu_K^{L,n} b_K(\varphi_j, \varphi_i) \,, \\
&= \int_{S_i} \left(\mathbf{f}'(\tilde{u}^n) \cdot \nabla \sum_j \varphi_j(\mathbf{x}) + \sigma(\mathbf{x}) \sum_j \varphi_j(\mathbf{x}) \right) \varphi_i(\mathbf{x}) \, d\mathbf{x} \,, \\
&= \int_{S_i} \sigma(\mathbf{x}) \varphi_i(\mathbf{x}) \, d\mathbf{x} \,, \\
&\geq 0 \,. \quad \blacksquare
\end{aligned}$$

Remark If boundary conditions are weakly imposed, as discussed in Section 2.2.1.1, then the steady-state system matrix is modified and $\tilde{A}_{i,j} \geq A_{i,j}$. Thus

$$\sum_j \tilde{\mathbf{A}}_{i,j} \geq \sum_j \mathbf{A}_{i,j} \geq 0 \,, \tag{2.83}$$

and Lemma 2.3.2 still holds.

Lemma 2.3.3 (Non-Negativity of Diagonal Elements) *The diagonal elements of the matrix* $\mathbf{A}^{L,n}$ *are non-negative:*

$$A_{i,i}^{L,n} \geq 0 \,, \quad \forall i \,.$$

Proof Using Lemma 2.3.2,

$$\sum_j A_{i,j}^{L,n} \geq 0 \,.$$

Thus,

$$A_{i,i}^{L,n} \geq -\sum_{j \neq i} A_{i,j}^{L,n} \,.$$

From Lemma 2.3.1, the off-diagonal elements are known to be non-positive: $A_{i,j}^{L,n} \leq 0$.

Thus, $-A_{i,j}^{L,n} \geq 0$ and finally,

$$A_{i,i}^{L,n} \geq 0\,. \quad \blacksquare$$

Lemma 2.3.4 (Diagonal Dominance) *The matrix* $\mathbf{A}^{L,n}$ *is strictly diagonally dominant:*

$$\left|A_{i,i}^{L,n}\right| \geq \sum_{j\neq i} \left|A_{i,j}^{L,n}\right|\,, \quad \forall i\,.$$

Proof Using the inequalities $\sum\limits_{j} A_{i,j}^{L,n} \geq 0$ and $A_{i,j}^{L,n} \leq 0, j \neq i$, it is proven that $\mathbf{A}^{L,n}$ is strictly diagonally dominant:

$$\begin{aligned}
\sum_{j} A_{i,j}^{L,n} &\geq 0\,,\\
\sum_{j\neq i} A_{i,j}^{L,n} + A_{i,i}^{L,n} &\geq 0\,,\\
\left|A_{i,i}^{L,n}\right| &\geq \sum_{j\neq i} -A_{i,j}^{L,n}\,,\\
\left|A_{i,i}^{L,n}\right| &\geq \sum_{j\neq i} \left|A_{i,j}^{L,n}\right|\,. \quad \blacksquare
\end{aligned}$$

Theorem 2.3.1 (M-Matrix) *The matrix* $\mathbf{A}^{L,n}$ *is an M-Matrix.*

Proof To prove that a matrix is an M-Matrix, it is sufficient to prove that the following 3 statements are true:

1. $A_{i,j}^{L,n} \leq 0\,, \quad j \neq i\,, \forall i,$
2. $A_{i,i}^{L,n} \geq 0\,, \quad \forall i,$
3. $\left|A_{i,i}^{L,n}\right| \geq \sum\limits_{j\neq i} \left|A_{i,j}^{L,n}\right|\,, \quad \forall i.$

These conditions are proven by Lemmas 2.3.1, 2.3.3, and 2.3.4, respectively. $\blacksquare$

2.3.4 Positivity Preservation

In this section, it will be shown that the low-order scheme for each temporal discretization preserves non-negativity of the solution, given that a CFL-like time step size condition is satisfied. This section builds upon the results of Section 2.3.3, which proved that the low-order system matrix $\mathbf{A}^{L,n}$ is an M-matrix, which to recall Equation (2.82), has the property

$$\mathbf{A}\mathbf{x} \geq \mathbf{0} \Rightarrow \mathbf{x} \geq \mathbf{0}\,,$$

and thus for a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$, proof of non-negativity of the right-hand-side vector proves non-negativity of the solution $\mathbf{x}$. For each temporal discretization, it will be shown that the system matrix inverted for the corresponding low-order system is also an M-matrix and that the right-hand-side vector for each system is non-negative. Thus positivity-preservation of the solution will be proven.

Theorem 2.3.2 (Non-Negativity of the Steady-State Low-Order Solution) *The solution of the steady-state low-order system given by Equation* (2.78) *is non-negative:*

$$U_i^L \geq 0\,, \quad \forall i\,.$$

Proof By Theorem 2.3.1, the system matrix $\mathbf{A}^L$ is an M-matrix, and by assumption in Section 2.1.1, the source q is non-negative, and thus the steady-state right-hand-side vector entries b_i are non-negative. Invoking the M-matrix property concludes the proof. ∎

Theorem 2.3.3 (Non-Negativity Preservation of the Explicit Euler Low-Order Solution) *If the old solution* $\mathbf{U}^n$ *is non-negative and the time step size* Δt

satisfies

$$\Delta t \leq \frac{M_{i,i}^{L}}{A_{i,i}^{L,n}}\,, \quad \forall i\,, \tag{2.84}$$

then the new solution $\mathbf{U}^{L,n+1}$ *of the explicit Euler low-order system given by Equation* (2.80) *is non-negative:*

$$U_i^{L,n+1} \geq 0\,, \quad \forall i\,.$$

Proof Rearranging Equation (2.80),

$$\mathbf{M}^{L}\mathbf{U}^{L,n+1} = \Delta t \mathbf{b}^{n} + \mathbf{M}^{L}\mathbf{U}^{n} + \Delta t \mathbf{A}^{L,n}\mathbf{U}^{n}\,.$$

Thus the system matrix to invert is the lumped mass matrix, which is an M-matrix since it is diagonal and positive. The right-hand-side vector $\mathbf{y}$ of this system has the entries

$$y_i = \Delta t b_i^{n} + \left(M_{i,i}^{L} - \Delta t A_{i,i}^{L,n}\right) U_i^{n} - \Delta t \sum_{j \neq i} A_{i,j}^{L,n} U_j^{n}\,.$$

It now just remains to prove that these entries are non-negative. As stated previously, the source function q is assumed to be non-negative and thus the steady-state right-hand-side vector is non-negative. Due to the time step size assumption given by Equation (2.84) and Lemma 2.3.3,

$$M_{i,i}^{L} - \Delta t A_{i,i}^{L,n} \geq 0\,,$$

and by Lemma 2.3.1, the off-diagonal sum term is also non-negative. Thus y_i is a sum of non-negative terms. Invoking the M-matrix property concludes the proof. ∎

Theorem 2.3.4 (Non-Negativity Preservation of the Theta Low-Order So-

lution) *If the old solution* $\mathbf{U}^n$ *is non-negative and the time step size* Δt *satisfies*

$$\Delta t \leq \frac{M_{i,i}^L}{(1-\theta)A_{i,i}^{L,n}}\,, \quad \forall i\,, \tag{2.85}$$

then the new solution $\mathbf{U}^{L,n+1}$ *of the Theta low-order system given by Equation* (2.81) *is non-negative:*

$$U_i^{L,n+1} \geq 0\,, \quad \forall i\,.$$

Proof Rearranging Equation (2.81),

$$\left(\mathbf{M}^L + \theta\Delta t\mathbf{A}^{L,n+1}\right)\mathbf{U}^{L,n+1} = \Delta t\left((1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1}\right) + \mathbf{M}^L\mathbf{U}^n - (1-\theta)\Delta t\mathbf{A}^{L,n}\mathbf{U}^n\,.$$

Thus the system matrix to invert is $\mathbf{M}^L + \Delta t\theta\mathbf{A}^{L,n+1}$, which is an M-matrix since it is a linear combination of two M-matrices. The right-hand-side vector $\mathbf{y}$ of this system has the entries

$$y_i = \Delta t\left((1-\theta)b_i^n + \theta b_i^{n+1}\right) + \left(M_{i,i}^L - (1-\theta)\Delta t A_{i,i}^{L,n}\right)U_i^n - (1-\theta)\Delta t\sum_{j\neq i} A_{i,j}^{L,n}U_j^n\,.$$

It now just remains to prove that these entries are non-negative. As stated previously, the source function q is assumed to be non-negative and thus the steady-state right-hand-side vector is non-negative. Due to the time step size assumption given by Equation (2.85) and Lemma 2.3.3,

$$M_{i,i}^L - (1-\theta)\Delta t A_{i,i}^{L,n} \geq 0\,,$$

and by Lemma 2.3.1, the off-diagonal sum term is also non-negative. Thus y_i is a sum of non-negative terms. Invoking the M-matrix property concludes the proof. ∎

2.3.5 Local Discrete Maximum Principles

In this section, the low-order schemes described in Section 2.3.2 are each shown to satisfy a local discrete maximum principle (DMP). The DMP is analogous to local extremum diminishing (LED) constraints such as

$$U^n_{\min,i} \leq U^{n+1}_i \leq U^n_{\max,i}\,, \tag{2.86}$$

where $U^n_{\min,i}$ and $U^n_{\max,i}$ are the minimum and maximum, respectively, of the old solution in the neighborhood of i. Thus if i corresponds to a local minimum, it cannot shrink, and if it corresponds to a local maximum, it cannot grow. However, this particular constraint does not apply to general scalar conservation law given by Equation (2.1) due to the presence of the reaction term and source term; without these terms, the DMP given in this section for explicit Euler will reduce to this constraint. The DMP gives proof that the solution local minima will not decrease without a reaction term and that the solution local maxima will not increase without a source. In addition, the lower DMP bound can provide proof that a particular scheme is positivity-preserving.

Theorem 2.3.5 (Steady-State Scheme Local Discrete Maximum Principle) *The solution of the steady-state low-order system given by Equation* (2.78) *satisfies the following local discrete maximum principle:*

$$W_i^{\mathrm{DMP},-}(\mathbf{U}^L) \leq U_i^L \leq W_i^{\mathrm{DMP},+}(\mathbf{U}^L) \quad \forall i\,, \tag{2.87a}$$

$$W_i^{\mathrm{DMP},\pm}(\mathbf{U}^L) \equiv -\frac{1}{A^L_{i,i}} \sum_{j\neq i} A^L_{i,j} U^L_{\substack{\max\\ \min},j\neq i} + \frac{b_i}{A^L_{i,i}}\,, \tag{2.87b}$$

where

$$U^L_{\min,j\neq i} \equiv \min_{j\neq i\in\mathcal{I}(S_i)} U^L_j\,, \quad U^L_{\max,j\neq i} \equiv \max_{j\neq i\in\mathcal{I}(S_i)} U^L_j\,.$$

Proof

$$\begin{aligned}
\sum_j A^L_{i,j}U^L_j &= b_i\,,\\
A^L_{i,i}U^L_i &= \sum_{j\neq i} -A^L_{i,j}U^L_j + b_i\,,\\
A^L_{i,i}U^L_i &\leq \left(\sum_{j\neq i} -A^L_{i,j}\right) U^L_{\max,j\neq i} + b_i\,,\\
U^L_i &\leq -\frac{1}{A^L_{i,i}}\sum_{j\neq i} A^L_{i,j}U^L_{\max,j\neq i} + \frac{b_i}{A^L_{i,i}}\,.
\end{aligned}$$

A similar analysis is performed to prove the lower bound for U^L_i. ∎

Theorem 2.3.6 (Explicit Euler Scheme Local Discrete Maximum Principle) *If the time step size Δt satisfies Equation* (2.84)*,*

$$\Delta t \leq \frac{M^L_{i,i}}{A^{L,n}_{i,i}}\,, \quad \forall i\,,$$

then the solution of the explicit Euler low-order system given by Equation (2.80) *satisfies the following local discrete maximum principle:*

$$W^{\mathrm{DMP},-}_i \leq U^{L,n+1}_i \leq W^{\mathrm{DMP},+}_i\,, \quad \forall i\,, \tag{2.88a}$$

$$W^{\mathrm{DMP},\pm}_i \equiv \left(1 - \frac{\Delta t}{M^L_{i,i}}\sum_j A^{L,n}_{i,j}\right) U^n_{\substack{\max\\\min},i} + \frac{\Delta t}{M^L_{i,i}} b^n_i\,, \tag{2.88b}$$

where $U^n_{\substack{\max\\\min},i} = \min\limits^{\max}_{j\in\mathcal{I}(S_i)} U^n_j$.

Proof Evaluating row i of the low-order Explicit Euler system, given by Equation

(2.80), and rearranging,

$$U_i^{L,n+1} = U_i^n - \frac{\Delta t}{M_{i,i}^L} \sum_j A_{i,j}^{L,n} U_j^n + \frac{\Delta t}{M_{i,i}^L} b_i^n \,,$$

where $M_{i,i}^L$ is the ith element of the lumped mass matrix. Rearranging this equation,

$$U_i^{L,n+1} = \left(1 - \frac{\Delta t}{M_{i,i}^L} A_{i,i}^{L,n}\right) U_i^n - \frac{\Delta t}{M_{i,i}^L} \sum_{j \neq i} A_{i,j}^{L,n} U_j^n + \frac{\Delta t}{M_{i,i}^L} b_i^n \,,$$

The time step size condition in Equation (2.84) gives that $1 - \frac{\Delta t}{M_{i,i}^L} A_{i,i}^{L,n} \geq 0$, and by Lemma 2.3.1, it is known that the off-diagonal elements $A_{i,j}^{L,n}, j \neq i$, are non-positive. Thus, the following inequality is able to be applied:

$$U_i^{L,n+1} \leq \left(1 - \frac{\Delta t}{M_{i,i}^L} \sum_j A_{i,j}^{L,n}\right) U_{\max,i}^n + \frac{\Delta t}{M_{i,i}^L} b_i^n \,,$$

and similarly for the lower bound. ∎

Theorem 2.3.7 (Theta Scheme Local Discrete Maximum Principle) *If the time step size Δt satisfies Equation* (2.85)*,*

$$\Delta t \leq \frac{M_{i,i}^L}{(1-\theta) A_{i,i}^{L,n}} \,, \quad \forall i \,,$$

then the solution of the Theta low-order system given by Equation (2.81) *satisfies the following local discrete maximum principle:*

$$W_i^{\mathrm{DMP},-}(\mathbf{U}^{L,n+1}) \leq U_i^{L,n+1} \leq W_i^{\mathrm{DMP},+}(\mathbf{U}^{L,n+1}) \quad \forall i \,, \tag{2.89a}$$

$$W_i^{\mathrm{DMP},\pm}(\mathbf{U}^{L,n+1}) \equiv \frac{1}{1+\frac{\theta\Delta t}{M_{i,i}^L}A_{i,i}^{L,n+1}}\Bigg[\left(1-\frac{(1-\theta)\Delta t}{M_{i,i}^L}A_{i,i}^{L,n}\right)U_i^n$$
$$-\frac{\Delta t}{M_{i,i}^L}\left((1-\theta)\sum_{j\neq i}A_{i,j}^{L,n}U^n_{\substack{\max\\\min},j\neq i}+\theta\sum_{j\neq i}A_{i,j}^{L,n+1}U^{n+1}_{\substack{\max\\\min},j\neq i}\right)$$
$$+\frac{\Delta t}{M_{i,i}^L}\left((1-\theta)b_i^n+\theta b_i^{n+1}\right)\Bigg]. \quad (2.89b)$$

where

$$U^n_{\substack{\max\\\min},j\neq i} \equiv \overset{\max}{\underset{j\neq i\in\mathcal{I}(S_i)}{\min}} U_j^n\,.$$

Proof Evaluating row i of the low-order Theta system, given by Equation (2.81),

$$M_{i,i}^L\frac{U_i^{L,n+1}-U_i^n}{\Delta t}+(1-\theta)\sum_j A_{i,j}^{L,n}U_j^n+\theta\sum_j A_{i,j}^{L,n+1}U_j^{L,n+1}=(1-\theta)b_i^n+\theta b_i^{n+1},$$

and rearranging,

$$U_i^{L,n+1}=\frac{1}{1+\frac{\theta\Delta t}{M_{i,i}^L}A_{i,i}^{L,n+1}}\Bigg[\left(1-\frac{(1-\theta)\Delta t}{M_{i,i}^L}A_{i,i}^{L,n}\right)U_i^n$$
$$-\frac{\Delta t}{M_{i,i}^L}\left((1-\theta)\sum_{j\neq i}A_{i,j}^{L,n}U_j^n+\theta\sum_{j\neq i}A_{i,j}^{L,n+1}U_j^{n+1}\right)+\frac{\Delta t}{M_{i,i}^L}\left((1-\theta)b_i^n+\theta b_i^{n+1}\right)\Bigg].$$

By Lemma 2.3.1, it is known that the off-diagonal elements $A_{i,j}^{L,n}, j\neq i$, are non-positive, and by Equation (2.85) the term $1-\frac{(1-\theta)\Delta t}{M_{i,i}^L}A_{i,i}^{L,n}$ is non-negative. Thus, the

following inequality is able to be applied:

$$U_i^{L,n+1} \leq \frac{1}{1+\frac{\theta\Delta t}{M_{i,i}^L}A_{i,i}^{L,n+1}}\Bigg[\left(1-\frac{(1-\theta)\Delta t}{M_{i,i}^L}A_{i,i}^{L,n}\right)U_i^n \\ -\frac{\Delta t}{M_{i,i}^L}\left((1-\theta)\sum_{j\neq i}A_{i,j}^{L,n}U_{\max,j\neq i}^n+\theta\sum_{j\neq i}A_{i,j}^{L,n+1}U_{\max,j\neq i}^{n+1}\right) \\ +\frac{\Delta t}{M_{i,i}^L}\left((1-\theta)b_i^n+\theta b_i^{n+1}\right)\Bigg].$$

and similarly for the lower bound. ∎

2.4 Low-Order Scheme for Conservation Law Systems

This section describes a low-order scheme for *systems* of conservation laws since the scalar methodology described in Section 2.3 no longer applies. The low-order scheme is adapted from recent work by Guermond [14], which focuses on a property called *domain-invariance*, which is described in Section 2.4.1. Section 2.4.2 summarizes the low-order system. In contrast to the scalar case, only explicit Euler and SSPRK time discretizations are considered for the system case.

2.4.1 Invariant Domain

In the case of *systems* of conservation laws, the local discrete maximum principle is no longer valid. Instead, the desired property is *domain-invariance*. Thus the low-order scheme for conservation law systems is designed around this property. The approach given in this section is taken from recent work by Guermond [14]. This section will begin by making definitions necessary to describe the domain-invariance property of the low-order scheme. Subsequent sections will define the scheme, including the low-order diffusion terms necessary to ensure the invariant domain property.

It is desired that the solution process produce admissible (physical, entropy-

satisfying) solutions; let the space of these solutions be $\mathcal{A} \subset \mathbb{R}^m$, where m is the number of components in the system. The following definition of an *invariant set* comes from [14]:

Definition 2.4.1 (Invariant Set) *Consider the following Riemann initial value problem (IVP):*

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial}{\partial x}(\mathbf{F}(\mathbf{u}) \cdot \mathbf{n}) = 0\,, \quad (x,t) \in \mathbb{R} \times \mathbb{R}_+\,, \quad \mathbf{u}(x,0) = \begin{cases} \mathbf{u}_L\,, & x \leq 0\,, \\ \mathbf{u}_R\,, & x > 0\,, \end{cases} \tag{2.90}$$

where $\mathbf{u} \in \mathbb{R}^m$ *is the* m*-component solution,* $\mathbf{F}(\mathbf{u}) \equiv [\mathbf{f}(\mathbf{u})_1, \ldots, \mathbf{f}(\mathbf{u})_m]^T$ *is a matrix of size* $m \times d$*, where each row* i *is the* d*-dimensional conservation law flux for component* i*, and* $\mathbf{n}$ *is a unit vector in* $\mathbb{R}^d$*. This problem has a unique solution which is denoted by* $\mathbf{u}(\mathbf{u}_L, \mathbf{u}_R, \mathbf{n})(x,t)$*. A set* $A \subset \mathcal{A} \subset \mathbb{R}^m$ *is called an invariant set for the Riemann problem given by Equation* (2.90) *if and only if* $\forall (\mathbf{u}_L, \mathbf{u}_R) \in A \times A$*,* $\forall \mathbf{n}$ *on the unit sphere, and* $\forall t > 0$*, the average of the entropy solution* $\mathbf{u}(\mathbf{u}_L, \mathbf{u}_R, \mathbf{n})$ *over the Riemann fan, i.e.,*

$$\bar{\mathbf{u}} \equiv \frac{1}{t(\lambda_m^+ - \lambda_1^-)} \int_{\lambda_1^- t}^{\lambda_m^+ t} \mathbf{u}(\mathbf{u}_L, \mathbf{u}_R, \mathbf{n})(x,t)dx\,, \tag{2.91}$$

is an element in A*.*

To summarize the definition above, an invariant set is one in which a 1-D Riemann problem using any two elements in the set as left and right data in any direction produces an entropy solution that remains in that set.

Before defining an invariant domain, first recall the following definition for a *convex set*:

Definition 2.4.2 (Convex Set) *A convex set* A *is a set such that for any two elements in the set, the line connecting the two remains completely in the set. As*

a consequence, any convex combination of elements in the set remains in the set: $\sum_k a_k \mathbf{b}_k \in A$, *where* $\mathbf{b}_k \in A \; \forall k$, $a_k \geq 0 \; \forall k$, *and* $\sum_k a_k = 1$.

Now the definition for an *invariant domain* is made:

Definition 2.4.3 (Invariant Domain) *Let R be defined as the discrete solution process, which produces each subsequent approximate solution: $\mathbf{u}^{n+1} = R(\mathbf{u}^n)$. A convex invariant set A is called an invariant domain for the process R if and only if $\forall \mathbf{u} \in A$, $R(\mathbf{u}) \in A$.*

Now all of the necessary definitions have been made. Proving the invariant domain property with respect to the discrete process R takes the following approach:

1. Assume the initial data $\mathbf{u}^0 \in A$, where A is a convex invariant set.
2. Prove that the discrete scheme R is such that $\mathbf{u}^{n+1} \equiv R(\mathbf{u}^n)$ can be expressed a convex combination of elements in A: $\mathbf{u}^{n+1} = \sum_k a_k \mathbf{b}_k$.
   (a) Prove $\sum_k a_k = 1$.
   (b) Prove $a_k \geq 0 \, \forall k$, which requires conditions on the time step size.
   (c) Prove $\mathbf{b}_k \in A \; \forall k$.
3. Invoke the definition of a convex set to prove that A is an invariant domain for the process R.

These proofs are given in detail in [14] and will not be reproduced here. The time step size requirement is the following:

$$\Delta t \leq \frac{1}{2} \frac{\underline{\Delta x}}{\lambda_{\max}(A)} \frac{\mu_{\min} \vartheta_{\min}}{\mu_{\max}} \,, \tag{2.92a}$$

$$\underline{\Delta x} \equiv \min_K \underline{\Delta x}_K \,, \quad \underline{\Delta x}_K \equiv \frac{1}{\max\limits_{i \neq j \in \mathcal{I}_K} \|\nabla \varphi_i\|_{L^\infty(S_{i,j})}} \,, \tag{2.92b}$$

$$\vartheta_{\min} \equiv \min_K \vartheta_K \,, \quad \vartheta_K \equiv \frac{1}{n_K - 1} \,, \tag{2.92c}$$

$$\mu_{\min} \equiv \min_K \min_{i \in \mathcal{I}_K} \frac{1}{|\mathcal{D}_K|} \int_K \varphi_i \, d\mathbf{x} \,, \quad \mu_{\max} \equiv \max_K \max_{i \in \mathcal{I}_K} \frac{1}{|\mathcal{D}_K|} \int_K \varphi_i \, d\mathbf{x} \,, \tag{2.92d}$$

$$\lambda^{\max}(A) \equiv \max_{\mathbf{n} \in S^d(\mathbf{0},1)} \max_{\mathbf{u}_L, \mathbf{u}_R \in A} \lambda^{\max}(\mathbf{n}, \mathbf{u}_L, \mathbf{u}_R) \,, \tag{2.92e}$$

where $S^d(\mathbf{0}, 1)$ is the unit sphere in dimension d. For example, in 1-D, $\underline{\Delta x} = \Delta x_{\min} \equiv \min_K \Delta x_K$, and $\vartheta_{\min} = \mu_{\min} = \mu_{\max} = 1$, so the time step size requirement becomes

$$\Delta t \leq \frac{1}{2} \frac{\Delta x_{\min}}{\lambda_{\max}(A)} \,. \tag{2.93}$$

From this expression, it is easier to see the underlying requirement: for a given time step step, the elementary waves generated from each Riemann problem between neighboring nodal data must not travel far enough such that they can interact. Each of these Riemann problems is separated by a distance $\geq \Delta x_{\min}$, so if for example for a pair of adjacent Riemann problems, a wave is traveling to the right from the left Riemann problem and to the left from the right Riemann problem, then they could interact at half the distance between them. In practice, to determine $\lambda_{\max}(A)$ for a transient, one needs to loop over all pairs of initial data $\mathbf{u}_i$ and $\mathbf{u}_j$ for which the associated test functions have nonempty shared support $S_{i,j}$, and compute the maximum wave speeds from the associated Riemann problem (or some upper bound to them).

2.4.2 Low-Order System

Starting with Equation (2.53), the low-order scheme lumps the mass matrix and adds a low-order artificial diffusion term:

$$M_{i,i}^{L}\frac{\mathbf{U}_i^{L,n+1}-\mathbf{U}_i^{n}}{\Delta t}+\sum_j \mathbf{c}_{i,j}\cdot \mathrm{F}_j^{n}+\sum_j D_{i,j}^{L,n}\mathbf{U}_j^{n}=\mathbf{b}_i^{n}\,, \tag{2.94}$$

where the quantities $\mathbf{c}_{i,j}$ and F_j are defined in Section 2.2.2 and the low-order artificial diffusion matrix $\mathbf{D}^L$ is given in the following definition, which is designed in [14] to be the smallest amount of artificial possible to give the invariant domain property described in Section 2.4.1:

Definition 2.4.4 (Low-Order Artificial Diffusion Matrix) *The low-order artificial diffusion matrix $\mathbf{D}^{L,n}$ is defined as*

$$D_{i,j}^{L,n}\equiv \max(\lambda_{i,j}^{max}\left\|\mathbf{c}_{i,j}\right\|_{L^2},\lambda_{j,i}^{max}\left\|\mathbf{c}_{j,i}\right\|_{L^2}) \quad j\neq i\,, \quad D_{i,i}^{L,n}\equiv -\sum_{j\neq i} D_{i,j}^{L,n}\,, \tag{2.95}$$

where $\lambda_{i,j}^{max}\equiv\lambda^{max}(\mathbf{n}_{i,j},\mathbf{U}_i^n,\mathbf{U}_j^n)$ is the maximum wave speed in the 1-D Riemann problem in the direction $\mathbf{n}_{i,j}\equiv \mathbf{c}_{i,j}/\left\|\mathbf{c}_{i,j}\right\|_{L^2}$ with left state $\mathbf{U}_i^n$ and right state $\mathbf{U}_j^n$.

2.5 High-Order Scheme for Scalar Conservation Laws

In this section, the entropy viscosity method introduced by Guermond [13][12] is extended to the scalar transport model given by Equation (2.1). Section 2.5.1 defines the entropy viscosity for the scalar transport model, and Section 2.5.2 summarizes the high-order system for each of the time discretizations considered in this dissertation.

2.5.1 Entropy Viscosity

To construct a high-order scheme, the concept of entropy viscosity [13] is used in conjunction with the local viscous bilinear form introduced in Equation (2.73). The

high-order viscosity $\nu_K^{H,n}$ is computed as the minimum of the low-order viscosity $\nu_K^{L,n}$ and the entropy viscosity $\nu_K^{\eta,n}$:

$$\nu_K^{H,n} = \min(\nu_K^{L,n}, \nu_K^{\eta,n}) \,, \tag{2.96}$$

where the entropy viscosity is defined as

$$\nu_K^{\eta,n} = \frac{c_{\mathcal{R}} \mathcal{R}_K^n + c_{\mathcal{J}} \mathcal{J}_K^n}{\|\eta(\tilde{u}^n) - \bar{\eta}(\tilde{u}^n)\|_{L^\infty(\mathcal{D})}} \,. \tag{2.97}$$

The entropy is defined to be some convex function of u such as $\eta(u) = \frac{1}{2}u^2$. The entropy residual $\mathcal{R}_K^n$ is the following:

$$\mathcal{R}_K^n \equiv \left\| \mathcal{R}(\tilde{u}^n, \tilde{u}^{n-1}) \right\|_{L^\infty(K)} \,, \tag{2.98}$$

$$\mathcal{R}(\tilde{u}^n, \tilde{u}^{n-1}) \equiv \frac{\eta(\tilde{u}^n) - \eta(\tilde{u}^{n-1})}{\Delta t^n} + \eta'(\tilde{u}^n) \left(\nabla \cdot \mathbf{f}(\tilde{u}^n) + \sigma \tilde{u}^n - q \right) \,, \tag{2.99}$$

where the $L^\infty(K)$ norm is approximated as the maximum of the norm operand evaluated at each quadrature point on K. Because the entropy residual only measures cell-wise entropy production, it is useful to include entropy flux *jumps* in the definition of the entropy viscosity, since these jumps are a measure of edge-wise entropy production. The entropy viscosity definition uses the largest jump found on any of the faces of the cell K:

$$\mathcal{J}_K^n \equiv \max_{F \in \partial K} \mathcal{J}_F(\tilde{u}^n) \,, \tag{2.100}$$

where the jump $\mathcal{J}_F$ for a face F measures the jump in the normal component of the entropy flux across the cell interface:

$$\mathcal{J}_F(u) \equiv \|\mathbf{f}'(u) \cdot \mathbf{n}_F [\![\partial_n \eta(u)]\!]_F \|_{L^\infty(F)} \,, \tag{2.101}$$

where $\mathbf{n}_F$ is the outward unit vector for face F, the $L^\infty(F)$ norm is approximated as the maximum of the norm operand evaluated at each quadrature point on F, and the term $[\![\partial_n \eta(\tilde{u}^n)]\!]_F$ is computed as

$$\begin{aligned} [\![\partial_n \eta(\tilde{u}^n)]\!]_F &= [\![\nabla \eta(\tilde{u}^n) \cdot \mathbf{n}_F]\!] && (2.102) \\ &= [\![\eta'(\tilde{u}^n) \nabla \tilde{u}^n \cdot \mathbf{n}_F]\!] && (2.103) \\ &= (\eta'(\tilde{u}^n|_K) \nabla \tilde{u}^n|_K - \eta'(\tilde{u}^n|_{K'}) \nabla \tilde{u}^n|_{K'}) \cdot \mathbf{n}_F && (2.104) \end{aligned}$$

where $a|_K$ denotes the computation of a from cell K, and $a|_{K'}$ denotes the computation of a from the neighbor cell K' sharing the face F.

2.5.2 High-Order System

The high-order steady-state system matrix $\mathbf{A}^H$ is defined as the sum of the inviscid steady-state matrix $\mathbf{A}$ and a high-order artificial diffusion matrix $\mathbf{D}^H$:

$$\mathbf{A}^{H,n} = \mathbf{A} + \mathbf{D}^{H,n}\,, \tag{2.105}$$

where the high-order diffusion matrix is assembled in an identical manner as the low-order diffusion matrix but using the high-order viscosity defined in Equation (2.96) instead of the low-order viscosity:

$$D_{i,j}^{H,n} = \sum_{K \in \mathcal{K}(S_{i,j})} \nu_K^{H,n} b_K(\varphi_j, \varphi_i)\,. \tag{2.106}$$

Alternatively, one could choose to use no viscosity for the high-order scheme, i.e., use the standard CGFEM scheme, in which case the diffusion matrix would be a zero matrix; however, this approach is not recommended for general use for the reasons discussed in Section 2.5.1.

Unlike the low-order system, the high-order system does not lump the mass matrix, and it uses the high-order steady-state system matrix defined in Equation (2.105). The high-order system for different time discretizations follows:

Steady-state scheme:

$$\mathbf{A}^H \mathbf{U}^H = \mathbf{b} \tag{2.107}$$

Semi-discrete scheme:

$$\mathbf{M}^C \frac{d\mathbf{U}^H}{dt} + \mathbf{A}^H(t)\mathbf{U}^H(t) = \mathbf{b}(t) \tag{2.108}$$

Explicit Euler scheme:

$$\mathbf{M}^C \frac{\mathbf{U}^H - \mathbf{U}^n}{\Delta t} + \mathbf{A}^{H,n}\mathbf{U}^n = \mathbf{b}^n \tag{2.109}$$

Theta scheme:

$$\mathbf{M}^C \frac{\mathbf{U}^H - \mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}^{H,n}\mathbf{U}^n + \theta\mathbf{A}^{H,n+1}\mathbf{U}^H = (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1} \tag{2.110}$$

2.6 High-Order Scheme for Conservation Law Systems

In this section, the entropy viscosity method introduced by Guermond [13][12] is extended to the shallow water equations. Section 2.6.1 defines the entropy viscosity for the shallow water equations, and Section 2.6.2 summarizes the high-order system for explicit Euler (and thus SSPRK) time discretizations.

2.6.1 Entropy Viscosity

The entropy function for the shallow water equations is defined to be the sum of the kinetic and potential energy terms, in terms of the conservative variables and the bathymetry function b:

$$\eta(\mathbf{u}, b) = \frac{1}{2}\frac{\mathbf{q}\cdot\mathbf{q}}{h} + \frac{1}{2}gh\left(h + b\right) . \tag{2.111}$$

Remark Omission of the bathymetry term in the potential energy term in the entropy definition has no effect on either the entropy residual or entropy jump due to the assumption that b is not a function of time. Thus in implementation, the following definition can be used:

$$\eta(\mathbf{u}) = \frac{1}{2}\frac{\mathbf{q}\cdot\mathbf{q}}{h} + \frac{1}{2}gh^2\,, \tag{2.112}$$

which is often more convenient since it is a function of the conservative variables only.

The entropy flux, derived in Appendix B, is the following:

$$\mathbf{f}^{\eta}(\mathbf{u}, b) = g(h+b)\mathbf{q} + \frac{1}{2}\frac{(\mathbf{q}\cdot\mathbf{q})\,\mathbf{q}}{h^2}\,. \tag{2.113}$$

The entropy residual is defined to be the left hand side of Equation (B.6):

$$\mathcal{R}(\mathbf{u}^n, \mathbf{u}^{n-1}) \equiv \frac{\eta(\mathbf{u}^n) - \eta(\mathbf{u}^{n-1})}{\Delta t^{n-1}} + \nabla\cdot\mathbf{f}^{\eta}(\mathbf{u}^n, b)\,. \tag{2.114}$$

As in the scalar case, it can be helpful to define an *entropy jump* for an interior face F:

$$\mathcal{J}_F(\mathbf{u}) \equiv |(\llbracket\nabla\mathbf{f}^{\eta}(\mathbf{u})\rrbracket\cdot\mathbf{n}_F)\cdot\mathbf{n}_F|\,, \tag{2.115}$$

which can be interpreted as the jump in the gradient in the normal direction of the normal component of the entropy flux.

To compute the high-order diffusion matrix, one option is to follow the approach of the scalar case and compute a cell-wise entropy viscosity as in Equation (2.97):

$$\nu_K^{\eta,n} = \frac{c_{\mathcal{R}}\mathcal{R}_K^n + c_{\mathcal{J}}\mathcal{J}_K^n}{\hat{\eta}_K^n}\,, \tag{2.116}$$

where here $\hat{\eta}_K^n$ represents a more general normalization coefficient for a cell K than the global normalization presented for the scalar case,

$$\hat{\eta}_K^n = \hat{\eta}^n = \|\eta(\tilde{u}^n) - \bar{\eta}(\tilde{u}^n)\|_{L^\infty(\mathcal{D})} \,. \tag{2.117}$$

For the shallow water equations, a possible alternative normalization is the following, which unlike the above normalization, is *local*:

$$\hat{\eta}_K^n = \max_{\mathbf{x}_q \in \mathcal{Q}_K} g\tilde{h}_q^2 \,, \tag{2.118}$$

where $\mathbf{x}_q$ denotes a quadrature point in the set $\mathcal{Q}_K$ of quadrature points in a cell K and $\tilde{h}_q$ denotes $\tilde{h}(\mathbf{x}_q)$. The high-order diffusion matrix could then be assembled as in Equation (2.106).

Alternatively, one could compute an artificial diffusion matrix without computing cell-wise entropy viscosities, as was presented for the low-order case for systems in Equation (2.95). For example, one could define an entry of the high-order diffusion matrix as

$$D_{i,j}^{H,n} \equiv \min\left(D_{i,j}^{\eta,n}, D_{i,j}^{L,n}\right) \,, \tag{2.119}$$

where similar to before, the low-order diffusion is used as an upper bound of the entropy diffusion $D_{i,j}^{\eta,n}$, which could be defined in the form

$$D_{i,j}^{\eta,n} \equiv \frac{c_{\mathcal{R}}\mathcal{R}_{i,j}^n + c_{\mathcal{J}}\mathcal{J}_{i,j}^n}{\hat{\eta}_{i,j}^n} \,, \tag{2.120}$$

where for example $\mathcal{R}_{i,j}^n$ might be defined as

$$\mathcal{R}_{i,j}^n \equiv \left| \int_{S_{i,j}} \mathcal{R}(\mathbf{u}^n, \mathbf{u}^{n-1}) \varphi_i(\mathbf{x}) \varphi_j(\mathbf{x}) \, d\mathbf{x} \right| . \tag{2.121}$$

Possible definitions of $\mathcal{J}_{i,j}^n$ and $\hat{\eta}_{i,j}^n$ are omitted here. The results obtained in this dissertation used the first approach, in which cell-wise entropy viscosities were computed.

2.6.2 High-Order System

The high-order steady-state system matrix $\mathbf{A}^H$ is defined as the sum of the inviscid steady-state matrix $\mathbf{A}$ and a high-order artificial diffusion matrix $\mathbf{D}^H$:

$$\mathbf{A}^{H,n} = \mathbf{A}^n + \mathbf{D}^{H,n} , \tag{2.122}$$

Unlike the low-order system, the high-order system does not lump the mass matrix, and it uses the high-order steady-state system matrix defined in Equation (2.122). The high-order system for explicit Euler time discretization is the following:

$$\sum_j M_{i,j}^C \frac{\mathbf{U}_j^H - \mathbf{U}_j^n}{\Delta t} + \sum_j \mathbf{c}_{i,j} \cdot \mathrm{F}_j^n + \sum_j D_{i,j}^{H,n} \mathbf{U}_j^n = \mathbf{b}_i^n . \tag{2.123}$$

2.7 FCT Scheme for Scalar Conservation Laws

In this section the FCT scheme for scalar conservation laws is presented. This section is organized as follows. Section 2.7.1 introduces the FCT scheme and defines the antidiffusion correction fluxes, Section 2.7.2 discusses the solution bounds to impose on the FCT solution, Section 2.7.3 discusses the corresponding antidiffusion bounds, and Section 2.7.4 defines limiting coefficient definitions for the antidiffusive

fluxes.

2.7.1 FCT System

The crux of the flux-corrected transport scheme is to define an antidiffusive correction flux $\mathbf{p}$ from a monotone, low-order scheme to a high-order scheme. Thus to define $\mathbf{p}$ for a particular temporal discretization, $\mathbf{p}$ is added as a source to the respective low-order system given in Section 2.3.2, except that the solution at $n+1$ is no longer the low-order solution $\mathbf{U}^L$, but instead, the high-order solution $\mathbf{U}^H$. The systems defining $\mathbf{p}$ for each temporal discretization follow.

Steady-state scheme:

$$\mathbf{A}^L\mathbf{U}^H = \mathbf{b} + \mathbf{p} \tag{2.124}$$

Semi-discrete scheme:

$$\mathbf{M}^L\frac{d\mathbf{U}^H}{dt} + \mathbf{A}^L\mathbf{U}^H(t) = \mathbf{b}(t) + \mathbf{p} \tag{2.125}$$

Explicit Euler scheme:

$$\mathbf{M}^L\frac{\mathbf{U}^H - \mathbf{U}^n}{\Delta t} + \mathbf{A}^L\mathbf{U}^n = \mathbf{b}^n + \mathbf{p} \tag{2.126}$$

Theta scheme:

$$\mathbf{M}^L\frac{\mathbf{U}^H - \mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}^L\mathbf{U}^n + \theta\mathbf{A}^L\mathbf{U}^H = (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1} + \mathbf{p} \tag{2.127}$$

The object of FCT is to limit these correction fluxes to satisfy physically-motivated bounds imposed on the solution. To adopt the limiting procedure used in this dissertation, it is necessary to decompose the correction flux for a node i into a sum of correction fluxes coming into node i from its neighbors. These decomposed fluxes are conveniently represented by a correction flux matrix, denoted by $\mathbf{P}$, e.g., entry $P_{i,j}$ is the correction flux going into node i from node j, and $\sum_j P_{i,j} = p_i$. Thus the

question remains of how to distribute, or *decompose*, the correction flux p_i among its neighbors. A convenient decomposition reveals itself when the correction flux definitions given by Equations (2.124) - (2.127) are combined with the respective high-order system equations given by Equations (2.107) - (2.110). This yields new expressions for $\mathbf{p}$, which follow.

Steady-state scheme:

$$\mathbf{p} \equiv \left(\mathbf{D}^L - \mathbf{D}^H\right) \mathbf{U}^H \tag{2.128}$$

Semi-discrete scheme:

$$\mathbf{p} \equiv -\left(\mathbf{M}^C - \mathbf{M}^L\right) \frac{d\mathbf{U}^H}{dt} + \left(\mathbf{D}^L - \mathbf{D}^H(t)\right) \mathbf{U}^H(t) \tag{2.129}$$

Explicit Euler scheme:

$$\mathbf{p} \equiv -\left(\mathbf{M}^C - \mathbf{M}^L\right) \frac{\mathbf{U}^H - \mathbf{U}^n}{\Delta t^{n+1}} + \left(\mathbf{D}^L - \mathbf{D}^{H,n}\right) \mathbf{U}^n \tag{2.130}$$

Theta scheme:

$$\mathbf{p} \equiv -\left(\mathbf{M}^C - \mathbf{M}^L\right) \frac{\mathbf{U}^H - \mathbf{U}^n}{\Delta t^{n+1}} + (1-\theta)\left(\mathbf{D}^L - \mathbf{D}^{H,n}\right) \mathbf{U}^n + \theta\left(\mathbf{D}^L - \mathbf{D}^{H,n+1}\right) \mathbf{U}^H \tag{2.131}$$

These definitions suggest convenient decompositions because $\mathbf{M}^C - \mathbf{M}^L$ and $\mathbf{D}^L - \mathbf{D}^H$ are symmetric and feature zero row sums:

$$\sum_j \left(M^L_{i,j} - M^C_{i,j}\right) = 0\,,$$

$$M^L_{i,i} - \sum_j M^C_{i,j} = 0\,,$$

and

$$\sum_j \left(D^L_{i,j} - D^H_{i,j}\right) = 0\,.$$

The following decompositions result for each temporal discretization:

Steady-state scheme:

$$P_{i,j} = \left(D^L_{i,j} - D^H_{i,j}\right)\left(U^H_j - U^H_i\right) \tag{2.132}$$

Semi-discrete scheme:

$$P_{i,j} = -M^C_{i,j}\left(\frac{dU^H_j}{dt} - \frac{dU^H_i}{dt}\right) + \left(D^L_{i,j} - D^H_{i,j}(t)\right)\left(U^H_j(t) - U^H_i(t)\right) \tag{2.133}$$

Explicit Euler scheme:

$$P_{i,j} = -M^C_{i,j}\left(\frac{U^H_j - U^n_j}{\Delta t} - \frac{U^H_i - U^n_i}{\Delta t}\right) + \left(D^L_{i,j} - D^{H,n}_{i,j}\right)\left(U^n_j - U^n_i\right) \tag{2.134}$$

Theta scheme:

$$\begin{aligned} P_{i,j} = -M^C_{i,j}\left(\frac{U^H_j - U^n_j}{\Delta t} - \frac{U^H_i - U^n_i}{\Delta t}\right) &+ (1-\theta)\left(D^L_{i,j} - D^{H,n}_{i,j}\right)\left(U^n_j - U^n_i\right) \\ &+ \theta\left(D^L_{i,j} - D^{H,n+1}_{i,j}\right)\left(U^H_j - U^H_i\right) \end{aligned} \tag{2.135}$$

Note that the decompositions for the time-dependent schemes above use the fact

$$\sum_j M^C_{i,j}\frac{dU_i}{dt} = M^L_{i,i}\frac{dU_i}{dt}\,.$$

Remark If Dirichlet boundary conditions are strongly imposed on a degree of freedom i, then antidiffusive flux decompositions above do not apply. The total antidiffusive flux into i is $p_i = 0$, and the antidiffusive flux decomposition for degrees of freedom j neighboring i are no longer valid because there is no longer an equal and opposite antidiffusive flux $P_{i,j}$ to cancel $P_{j,i}$. Therefore, one must either accept the lack of the conservation property, or one must completely cancel $P_{j,i}$.

Up until this point, no limiting has been applied; using the schemes as defined in Equations (2.124) - (2.127) would simply reproduce the high-order solution $\mathbf{U}^H$. As stated previously, FCT applies a limiting procedure to the antidiffusive correction fluxes to satisfy the bounds that are imposed. This is achieved by assigning each *internodal* correction flux $P_{i,j}$ its own limiting coefficient $L_{i,j}$, which is applied as a scaling factor. Again, it is convenient to store these limiting coefficients in a matrix $\mathbf{L}$. Instead of adding the full correction flux to a node i, $p_i = \sum_j P_{i,j}$, the limited correction flux sum $\sum_j L_{i,j} P_{i,j}$ is added. In vector form, this row-wise product is denoted by $\mathbf{L} \cdot \mathbf{P}$, i.e., $(\mathbf{L} \cdot \mathbf{P})_i = \sum_j L_{i,j} P_{i,j}$. The FCT scheme for each temporal discretization follows.

Steady-state scheme:

$$\mathbf{A}^L \mathbf{U} = \mathbf{b} + \mathbf{L} \cdot \mathbf{P} \tag{2.136}$$

Semi-discrete scheme:

$$\mathbf{M}^L \frac{d\mathbf{U}}{dt} + \mathbf{A}^L \mathbf{U}(t) = \mathbf{b}(t) + \mathbf{L} \cdot \mathbf{P} \tag{2.137}$$

Explicit Euler scheme:

$$\mathbf{M}^L \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} + \mathbf{A}^L \mathbf{U}^n = \mathbf{b}^n + \mathbf{L} \cdot \mathbf{P} \tag{2.138}$$

Theta scheme:

$$\mathbf{M}^L \frac{\mathbf{U}^{n+1} - \mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}^L \mathbf{U}^n + \theta \mathbf{A}^L \mathbf{U}^{n+1} = (1-\theta)\mathbf{b}^n + \theta \mathbf{b}^{n+1} + \mathbf{L} \cdot \mathbf{P} \tag{2.139}$$

Each of these limiting coefficients is a number from zero to one; if the coefficient is one, then no limiting is applied, and if it is zero, then full limiting is applied, i.e., the internodal correction flux $P_{i,j}$ is completely canceled. If all of the limiting coefficients are zero, then the low-order solution $\mathbf{U}^L$ is recovered, and if all of the

limiting coefficients are one, then the high-order solution $\mathbf{U}^H$ is recovered. The definition of the limiting coefficients is given in Section 2.7.4.

2.7.2 Solution Bounds

The main idea of the FCT algorithm is to enforce some physically-motivated bounds on the each solution degree of freedom:

$$W_i^- \leq U_i^{n+1} \leq W_i^+ \,, \qquad \forall i \,. \tag{2.140}$$

For example, one may use the discrete maximum principle (DMP) bounds derived for the low-order schemes in Section 2.3.5 as FCT solution bounds. Alternatively, if the physics are relatively simple, one could use the method of characteristics to derive solution bounds. This approach is performed for the scalar transport equation in Appendix A. For solution variables that are physically non-negative, an obvious lower bound is zero: $W_i^- = 0$; if one only wants to prevent negativities, then one can simply set the upper bound to be an arbitrary large number $W_i^+ = c_{\text{large}}$. However, often one wants other properties such as monotonicity, in which case it is necessary to use more restrictive bounds.

Equation numbers for the local discrete maximum principle bounds for the different considered time discretizations, as well as the analytic local discrete maximum principle bounds, are given in Table 2.1.

2.7.3 Antidiffusion Bounds

In this section, the solution bounds discussed in Section 2.7.2 will be translated into bounds on the antidiffusive fluxes defined in Section 2.7.1. These antidiffusion bounds, along with the antidiffusive fluxes, are the input to the limiter (described in Section 2.7.4), which computes the limiting coefficient associated with each antidif-

Table 2.1: Discrete Maximum Principles

Case	*Equation*
Steady-State	Equation (2.87)
Explicit Euler	Equation (2.88)
Theta Method	Equation (2.89)
Analytic	Equation (A.9)

fusive flux. This section is organized as follows. Section 2.7.3.1 gives the definition of the antidiffusive bounds, derived from the imposed solution bounds, and Section 2.7.3.2 discusses some requirements on these bounds.

2.7.3.1 Definition of Antidiffusion Bounds

Theorem 2.7.1 (Antidiffusion Bounds for Steady-State Scheme) *Using the steady-state FCT scheme given by Equation* (2.136)*, the following antidiffusion bounds* $Q_i^{\pm}$ *correspond to the solution bounds* $W_i^- \leq U_i \leq W_i^+$*:*

$$Q_i^- \leq \sum_j L_{i,j} P_{i,j} \leq Q_i^+ \,, \tag{2.141a}$$

$$Q_i^{\pm} \equiv A_{i,i}^L W_i^{\pm} + \sum_{j \neq i} A_{i,j}^L U_j - b_i \,. \tag{2.141b}$$

Proof Starting with row i of Equation (2.136),

$$\sum_j A_{i,j}^L U_j = b_i + \sum_j L_{i,j} P_{i,j} \,.$$

Solving for $\sum_j L_{i,j} P_{i,j}$ gives

$$\sum_j L_{i,j} P_{i,j} = \sum_j A_{i,j}^L U_j - b_i \,.$$

The solution bounds for i are

$$W_i^- \leq U_i \leq W_i^+ .$$

Through addition/subtraction and multiplication/division operations, this principle can be made to look like the following:

$$\begin{aligned} A_{i,i}^L W_i^- + \sum_{j \neq i} A_{i,j}^L U_j - b_i = Q_i^- \\ \leq \sum_j A_{i,j}^L U_j - b_i = \sum_j L_{i,j} P_{i,j} \\ \leq A_{i,i}^L W_i^+ + \sum_{j \neq i} A_{i,j}^L U_j - b_i = Q_i^+ . \quad \blacksquare \end{aligned}$$

Theorem 2.7.2 (Antidiffusion Bounds for Explicit Euler Scheme) *Using the Explicit Euler FCT scheme given by Equation* (2.138)*, the following antidiffusion bounds $Q_i^\pm$ correspond to the solution bounds $W_i^- \leq U_i^{n+1} \leq W_i^+$:*

$$Q_i^- \leq \sum_j L_{i,j} P_{i,j} \leq Q_i^+ , \tag{2.142a}$$

$$Q_i^\pm \equiv M_{i,i}^L \frac{W_i^\pm - U_i^n}{\Delta t} + \sum_j A_{i,j}^L U_j^n - b_i^n . \tag{2.142b}$$

Proof Starting with row i of Equation (2.138),

$$M_{i,i}^L \frac{U_i^{n+1} - U_i^n}{\Delta t} + \sum_j A_{i,j}^L U_j^n = b_i^n + \sum_j L_{i,j} P_{i,j} .$$

Solving for $\sum_j L_{i,j}P_{i,j}$ gives

$$\sum_j L_{i,j}P_{i,j} = M^L_{i,i}\frac{U_i^{n+1}-U_i^n}{\Delta t} + \sum_j A^L_{i,j}U_j^n - b_i^n\,.$$

The solution bounds for degree of freedom i are

$$W_i^- \le U_i^{n+1} \le W_i^+\,.$$

Through addition/subtraction and multiplication/division operations, this principle can be made to look like the following:

$$\begin{aligned} M^L_{i,i}\frac{W_i^- - U_i^n}{\Delta t} + \sum_j A^L_{i,j}U_j^n - b_i^n &= Q_i^- \\ &\le M^L_{i,i}\frac{U_i^{n+1}-U_i^n}{\Delta t} + \sum_j A^L_{i,j}U_j^n - b_i^n = \sum_j L_{i,j}P_{i,j} \\ &\le M^L_{i,i}\frac{W_i^+ - U_i^n}{\Delta t} + \sum_j A^L_{i,j}U_j^n - b_i^n = Q_i^+\,. \end{aligned}$$ ∎

Theorem 2.7.3 (Antidiffusion Bounds for Theta Scheme) *Using the Theta FCT scheme given by Equation* (2.139)*, the following antidiffusion bounds $Q_i^\pm$ correspond to the solution bounds $W_i^- \le U_i^{n+1} \le W_i^+$:*

$$Q_i^- \le \sum_j L_{i,j}P_{i,j} \le Q_i^+\,, \tag{2.143a}$$

$$Q_i^\pm \equiv \left(\frac{M^L_{i,i}}{\Delta t} + \theta A^L_{i,i}\right)W_i^\pm + \left((1-\theta)A^L_{i,i} - \frac{M^L_{i,i}}{\Delta t}\right)U_i^n + \sum_{j\ne i} A^L_{i,j}U_j^\theta - b_i^\theta\,, \tag{2.143b}$$

$$U_j^\theta \equiv \theta U_j^{n+1} + (1-\theta)U_j^n\,, \tag{2.143c}$$

$$b_i^\theta \equiv \theta b_i^{n+1} + (1-\theta) b_i^n \,. \tag{2.143d}$$

Proof Starting with row i of Equation (2.139),

$$\left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) U_i^{n+1} + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta = b_i^\theta + \sum_j L_{i,j} P_{i,j} \,.$$

Solving for $\sum_j L_{i,j} P_{i,j}$ gives

$$\sum_j L_{i,j} P_{i,j} = \left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) U_i^{n+1} + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta \,.$$

The solution bounds for degree of freedom i are

$$W_i^- \leq U_i^{n+1} \leq W_i^+ \,.$$

Through addition/subtraction and multiplication/division operations, this principle can be made to look like the following:

$$\begin{aligned}
&\left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) W_i^- + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta = Q_i^- \\
&\leq \left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) U_i^{n+1} + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta = \sum_j L_{i,j} P_{i,j} \\
&\leq \left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) W_i^+ + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta = Q_i^+ \,. \qquad \blacksquare
\end{aligned}$$

2.7.3.2 Conditions on Antidiffusion Bounds

The FCT algorithm requires that the antidiffusion bounds $Q_i^\pm$,

$$Q_i^- \leq \sum_j L_{i,j} P_{i,j} \leq Q_i^+ \,,$$

must have the following properties:

$$Q_i^- \leq 0\,, \quad Q_i^+ \geq 0\,. \tag{2.144}$$

Otherwise, there would not be a fail-safe definition of the limiting coefficients; the FCT solution could not necessarily revert to the low-order solution, which corresponds to $L_{i,j} = 0\ \forall i,j$. If either of the conditions are violated, then the default choice $L_{i,j} = 0\ \forall i,j$ does not even satisfy the antidiffusion bounds, so there is no guarantee that there is any combination of limiting coefficients that could satisfy the bounds.

Most limiters assume these conditions are satisfied; if they are used despite the conditions not being satisfied, then negative limiting coefficients may be produced. While one may decide that negative limiting coefficients are acceptable, this is usually considered undesirable because it is usually the goal to produce limiting coefficients between 0 and 1:

$$0 \leq L_{i,j} \leq 1\,, \quad \forall i,j\,, \tag{2.145}$$

representing a scale between complete rejection and complete acceptance of an antidiffusion flux, respectively. A negative limiting coefficient would represent a *reversal* of an antidiffusive flux.

The following theorems give conditions under which Equation (2.144) is satisfied for each temporal discretization.

Theorem 2.7.4 (Signs of Antidiffusion Bounds for Steady-State Scheme) *The antidiffusion bounds $Q_i^\pm$ for the steady-state scheme, given by Equation* (2.141)*, satisfy the conditions given by Equation* (2.144) *when the DMP solution bounds given by Equation* (2.87) *are used and both the antidiffusion bounds and DMP solution*

bounds are evaluated with the same solution $\mathbf{U}$*:*

$$Q_i^+ = A_{i,i}^L W_i^{\mathrm{DMP},+}(\mathbf{U}) + \sum_{j \neq i} A_{i,j}^L U_j - b_i \geq 0\,, \tag{2.146a}$$

$$Q_i^- = A_{i,i}^L W_i^{\mathrm{DMP},-}(\mathbf{U}) + \sum_{j \neq i} A_{i,j}^L U_j - b_i \leq 0\,. \tag{2.146b}$$

Proof Starting with Equation (2.141),

$$Q_i^\pm \equiv A_{i,i}^L W_i^\pm + \sum_{j \neq i} A_{i,j}^L U_j - b_i\,.$$

Evaluating Equation (2.87) with an arbitrary solution $\mathbf{U}$ instead of the low-order solution $\mathbf{U}^L$ gives

$$W_i^{\mathrm{DMP},\pm}(\mathbf{U}) \equiv -\frac{1}{A_{i,i}^L} \sum_{j \neq i} A_{i,j}^L U_{\substack{\max\\\min},j\neq i} + \frac{b_i}{A_{i,i}^L}\,,$$

where

$$U_{\min,j\neq i} \equiv \min_{j \neq i \in \mathcal{I}(S_i)} U_j\,, \quad U_{\max,j\neq i} \equiv \max_{j \neq i \in \mathcal{I}(S_i)} U_j\,.$$

Substituting this expression for $W_i^\pm$ into the antidiffusion bounds expression gives

$$Q_i^\pm = -\sum_{j \neq i} A_{i,j}^L (U_{\substack{\max\\\min},j\neq i} - U_j)\,.$$

By Lemma 2.3.1, the off-diagonal matrix entries $A_{i,j}^L$ are non-positive, and by definition, $U_{\max,j\neq i} \geq U_j$ and $U_{\min,j\neq i} \leq U_j$, so the signs of Equation (2.144) are verified. ∎

Theorem 2.7.5 (Signs of Antidiffusion Bounds for Explicit Euler Scheme)

If the time step size Δt satisfies Equation (2.84)*,*

$$\Delta t \leq \frac{M_{i,i}^L}{A_{i,i}^{L,n}}\,, \quad \forall i\,,$$

then the antidiffusion bounds $Q_i^{\pm}$ for the explicit Euler scheme, given by Equation (2.142)*, satisfy the conditions given by Equation* (2.144) *when the DMP solution bounds given by Equation* (2.88) *are used:*

$$Q_i^+ = M_{i,i}^L \frac{W_i^{\mathrm{DMP},+} - U_i^n}{\Delta t} + \sum_j A_{i,j}^L U_j^n - b_i^n \geq 0\,, \tag{2.147a}$$

$$Q_i^- = M_{i,i}^L \frac{W_i^{\mathrm{DMP},-} - U_i^n}{\Delta t} + \sum_j A_{i,j}^L U_j^n - b_i^n \leq 0\,. \tag{2.147b}$$

Proof Starting with Equation (2.142),

$$Q_i^{\pm} \equiv M_{i,i}^L \frac{W_i^{\pm} - U_i^n}{\Delta t} + \sum_j A_{i,j}^L U_j^n - b_i^n\,.$$

Substituting Equation (2.88) into this expression gives

$$Q_i^{\pm} = \frac{M_{i,i}^L}{\Delta t}\left(\left(1 - \frac{\Delta t}{M_{i,i}^L}\sum_j A_{i,j}^L\right) U_{\substack{\max\\\min},i}^n + \frac{\Delta t}{M_{i,i}^L} b_i^n - U_i^n\right) + \sum_j A_{i,j}^L U_j^n - b_i^n\,,$$

$$Q_i^{\pm} = \left(\frac{M_{i,i}^L}{\Delta t} - \sum_j A_{i,j}^L\right) U_{\substack{\max\\\min},i}^n - \frac{M_{i,i}^L}{\Delta t} U_i^n + \sum_j A_{i,j}^L U_j^n\,,$$

$$Q_i^{\pm} = \left(\frac{M_{i,i}^L}{\Delta t} - A_{i,i}^L\right)\left(U_{\substack{\max\\\min},i}^n - U_i^n\right) - \sum_{j\neq i} A_{i,j}^L \left(U_{\substack{\max\\\min},i}^n - U_j^n\right)\,.$$

Due to the time step size restriction of Equation (2.84),

$$\frac{M_{i,i}^L}{\Delta t} - A_{i,i}^L \geq 0\,,$$

and by Lemma 2.3.1, the off-diagonal matrix entries $A_{i,j}^L$ are non-positive. By definition, $U_{\max,i}^n \geq U_j^n$ and $U_{\min,i}^n \leq U_j^n$, so the signs of Equation (2.144) are verified. ∎

Theorem 2.7.6 (Signs of Antidiffusion Bounds for Theta Scheme) *If the time step size Δt satisfies Equation* (2.85)*,*

$$\Delta t \leq \frac{M_{i,i}^L}{(1-\theta)A_{i,i}^{L,n}}\,, \quad \forall i\,,$$

then the antidiffusion bounds $Q_i^{\pm}$ for the theta scheme, given by Equation (2.143)*, satisfy the conditions given by Equation* (2.144) *when the DMP solution bounds given by Equation* (2.89) *are used and both the antidiffusion bounds and DMP solution bounds are evaluated with the same solution $\mathbf{U}^{n+1}$:*

$$Q_i^+ = \left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) W_i^{\mathrm{DMP},+}(\mathbf{U}^{n+1}) + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta \geq 0\,, \quad \text{(2.148a)}$$

$$Q_i^- = \left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) W_i^{\mathrm{DMP},-}(\mathbf{U}^{n+1}) + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta \leq 0\,. \quad \text{(2.148b)}$$

Proof Starting with Equation (2.143),

$$Q_i^{\pm} = \left(\frac{M_{i,i}^L}{\Delta t} + \theta A_{i,i}^L\right) W_i^{\pm} + \left((1-\theta)A_{i,i}^L - \frac{M_{i,i}^L}{\Delta t}\right) U_i^n + \sum_{j\neq i} A_{i,j}^L U_j^\theta - b_i^\theta\,.$$

Substituting Equation (2.89) into this expression gives

$$Q_i^{\pm} = -\sum_{j \neq i} A_{i,j}^L \left(\theta \left(U^{n+1}_{\substack{\max \\ \min}, j \neq i} - U_j^{n+1} \right) + (1-\theta) \left(U^{n}_{\substack{\max \\ \min}, j \neq i} - U_j^{n} \right) \right) ,$$

and by Lemma 2.3.1, the off-diagonal matrix entries $A_{i,j}^L$ are non-positive. By definition, $U^n_{\max, j \neq i} \geq U_j^n$ and $U^n_{\min, j \neq i} \leq U_j^n$, so the signs of Equation (2.144) are verified. ∎

If one wishes to use solution bounds $W_i^{\pm}$ other than the discrete maximum principle bounds, then one needs to take care to ensure that the imposed bounds yield the properties given by Equation (2.144). One way to achieve this is to ensure that the solution bounds $W_i^{\pm}$ themselves bound the solution bounds for which the properties given by Equation (2.144) are known to hold:

$$W_i^- \leq W_i^{\mathrm{DMP},-} , \tag{2.149a}$$

$$W_i^+ \geq W_i^{\mathrm{DMP},+} . \tag{2.149b}$$

These conditions may be enforced by the following operation:

$$\tilde{W}_i^- \equiv \min(W_i^-, W_i^{\mathrm{DMP},-}) , \tag{2.150a}$$

$$\tilde{W}_i^+ \equiv \max(W_i^+, W_i^{\mathrm{DMP},+}) . \tag{2.150b}$$

Then when the antidiffusion bounds $Q_i^{\pm}$ are computed from the solution bounds $\tilde{W}_i^{\pm}$, the properties of Equation (2.144) will still hold. Similarly, a more direct approach to enforcing the properties given by Equation (2.144) is not to perform the operation given by Equation (2.150) but instead to compute $Q_i^{\pm}$ with the unmodified bounds

$W_i^{\pm}$ and then perform the following operation:

$$\tilde{Q}_i^- \equiv \min(Q_i^-, 0)\,, \tag{2.151a}$$

$$\tilde{Q}_i^+ \equiv \max(Q_i^+, 0)\,. \tag{2.151b}$$

Unless otherwise noted, all of the FCT results in this dissertation employ this operation.

2.7.4 Limiting Coefficients

In this section, a number of limiting strategies are described. Section 2.7.4.1 describes a classic FCT limiter by Zalesak [31], and Section 2.7.4.2 describes an approach to achieving more antidiffusion by taking multiple passes through a limiter.

2.7.4.1 Zalesak Limiter

A classic multi-dimensional limiter used in the FCT algorithm is the limiter of Zalesak [31]. Zalesak's limiter separately considers positive and negative antidiffusive fluxes $P_{i,j}$ for a degree of freedom i and makes the conservative choice that the upper antidiffusion bound Q_i^+ does not consider the possibility of negative fluxes when assigning limiting coefficients for the positive fluxes, and similarly for the lower antidiffusion bound Q_i^-. The following theorem states and proves that the classic Zalesak limiter produces limiting coefficients that satisfy the imposed solution bounds.

Theorem 2.7.7 (Zalesak Limiting Coefficients) *Suppose that that the solution bounds $W_i^+ \leq U_i^{n+1} \leq W_i^-$ correspond to the following inequality for the antidiffusion fluxes:*

$$Q_i^- \leq \sum_j L_{i,j} P_{i,j} \leq Q_i^+\,, \tag{2.152}$$

where $Q_i^{\pm}$ are bounds that depend on the temporal discretization and satisfy the con-

ditions of Equation (2.144)*:*

$$Q_i^- \leq 0\,, \quad Q_i^+ \geq 0\,.$$

Then the following limiting coefficient definitions satisfy the solution bounds:

$$p_i^+ \equiv \sum_j \max(0, P_{i,j})\,, \qquad p_i^- \equiv \sum_j \min(0, P_{i,j})\,, \tag{2.153}$$

$$L_i^\pm \equiv \begin{cases} 1 & p_i^\pm = 0 \\ \min\left(1, \frac{Q_i^\pm}{p_i^\pm}\right) & p_i^\pm \neq 0 \end{cases}\,, \tag{2.154}$$

$$L_{i,j} \equiv \begin{cases} \min(L_i^+, L_j^-) & P_{i,j} \geq 0 \\ \min(L_i^-, L_j^+) & P_{i,j} < 0 \end{cases}\,. \tag{2.155}$$

Proof First, note some properties of the above definitions:

$$p_i^+ \geq 0\,, \qquad p_i^- \leq 0\,,$$
$$Q_i^+ \geq 0\,, \qquad Q_i^- \leq 0\,,$$
$$0 \leq L_i^\pm \leq 1\,,$$
$$0 \leq L_{i,j} \leq 1\,.$$

The proof will be given for the upper bound.

$$\sum_j L_{i,j} P_{i,j} \leq \sum_{j:P_{i,j}\geq 0} L_{i,j} P_{i,j} = \sum_{j:P_{i,j}\geq 0} \min(L_i^+, L_j^-) P_{i,j} \leq \sum_{j:P_{i,j}\geq 0} L_i^+ P_{i,j} = L_i^+ p_i^+\,.$$

For the case $p_i^+ = 0$,

$$L_i^+ p_i^+ = 0 \leq Q_i^+\,.$$

For the case $p_i^+ \neq 0$,

$$L_i^+ p_i^+ \leq \frac{Q_i^+}{p_i^+} p_i^+ = Q_i^+ \, .$$

Thus,

$$\sum_j L_{i,j} P_{i,j} \leq Q_i^+ \, .$$

The lower bound is proved similarly. ∎

2.7.4.2 Multi-Pass Limiting

In this section, a novel approach for FCT limitation is described, in which multiple passes are taken through any limiter to maximize the amount of antidiffusion that is accepted without violating the imposed bounds.

To date, no practical limiter has been developed that perfectly solves the optimization problem; solving an optimization problem exactly would be too computationally expensive to be used in practical calculations. Thus all limiters are sub-optimal; additional antidiffusion can be accepted without violating the imposed bounds. For example, recall that the Zalesak limiter described in Section 2.7.4.1 makes the safe choice that when considering the upper antidiffusion bound Q_i^+ for node i, the limiting coefficients for the positive antidiffusive fluxes, L_i^+, are computed while assuming that there is no contribution to the limited antidiffusion sum $\bar{p}_i$ from *negative* antidiffusive fluxes:

$$L_i^+ = \min\left(1, \frac{Q_i^+}{p_i^+}\right) \, .$$

A more optimal limiter would consider the negative fluxes here, for example by the following approach:

$$L_i^+ = \min\left(1, \frac{Q_i^+ - \bar{p}_i^-}{p_i^+}\right) \, ,$$

where

$$\bar{p}_i^- = \sum_{j:P_{i,j}<0} L_{i,j} P_{i,j} \, .$$

Of course, the problem with this hypothetical limiter is that one does not know the limiting coefficients for the negative antidiffusive fluxes yet; thus Zalesak designed the limiter to assume those limiting coefficients are all zero. Zalesak's limiter is sub-optimal whenever positive antidiffusive fluxes are limited (this happens when $p_i^+ > Q^+$) and there are some nonzero limiting coefficients for the negative antidiffusive fluxes. Furthermore, in the end, symmetry is enforced on the limiting coefficients, and thus the actual limiting coefficients applied to the positive antidiffusive fluxes are not L_i^+ but instead, $L_{i,j}$, which is either lesser or equal: $L_{i,j} \leq L_i^+$ (see Equation (2.155) to see why). With these arguments considered (and they apply similarly to the lower bound), Zalesak's limiter has room for improvement.

The idea of multi-pass limiting is to take multiple passes through a limiter to overcome the sub-optimality. A limiter may be considered as a black box in which the inputs are the antidiffusive fluxes $\{P_{i,j}\}_{i,j}$ and the antidiffusion bounds $\{Q_i^{\pm}\}_i$ and the outputs are the limiting coefficients $\{L_{i,j}\}_{i,j}$. See Figure 2.155 for an illustration of this concept.

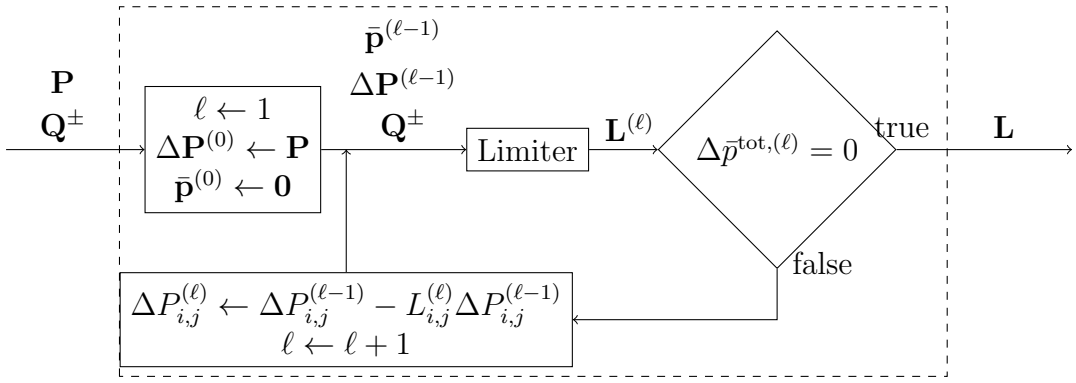
Figure 2.3: Limiter Input and Output

In multi-pass limiting, a loop is formed around the limiter. After each pass through the limiter, the remainder of the antidiffusive fluxes $\Delta\mathbf{P}$ is computed and passed back to the limiter in place of the original antidiffusive fluxes $\mathbf{P}$. In this

next iteration, some antidiffusion $\bar{p}_i$ has now already been accepted, and thus in the limiter, one does not start from $\bar{p}_i = 0$ when considering antidiffusion bounds but instead starts from $\bar{p}_i = \bar{p}_i^{(\ell-1)}$. For example if $\bar{p}_i > 0$, then the remaining positive antidiffusion fluxes have less room to be accepted, and the remaining negative fluxes have *more* room to be accepted. For example, Zalesak's limiter would be modified as

$$L_i^+ = \min\left(1, \frac{Q_i^+ - \bar{p}_i^{(\ell-1)}}{p_i^+}\right).$$

Perhaps a better approach is simply to pass into the limiter $\{Q_i^\pm - \bar{p}_i^{(\ell-1)}\}_i$ in place of $\{Q_i^\pm\}_i$. This achieves the same effect but avoids any modification of the limiter function. With each iteration, the accepted antidiffusion becomes smaller, and the iteration is terminated when the change in the total accepted antidiffusion source becomes sufficiently small. This is illustrated in Figure 2.4.



$$\Delta\bar{p}^{\mathrm{tot},(\ell)} \leftarrow \sum_{i,j} \left| L_{i,j}^{(\ell)} \Delta P_{i,j}^{(\ell-1)} \right|$$

Figure 2.4: Multi-Pass Limiting Diagram

2.8 FCT Scheme for Systems of Conservation Laws

2.8.1 Introduction to FCT for Systems

Recall that for scalar conservation laws, a discrete maximum principle (DMP) can be derived for the low-order scheme and used for the evaluation of the imposed FCT bounds. In the case of *systems* of conservation laws, no DMP is available to use for this purpose. Thus the bounds to impose on the FCT solution are unclear. The invariant domain property of the low-order scheme for systems is not useful in the manner that the DMP property was for scalar conservation laws; this is because while one can prove that the low-order solution is in some invariant domain, one does not necessarily have knowledge of the extent of the domain itself and thus it cannot be translated to solution bounds.

Direct extension of scalar FCT methodology to the systems case is often met with poor results because this extension is typically not physically valid. Kuzmin states that limitation on alternative sets of variables such as primitive variables or characteristic variables, rather than conservative variables, typically produces better quality FCT results [19]. For example, for the shallow water equations, one may consider the following sets of variables:

- Conservative: $\mathbf{u} \equiv [h, hu]^{\mathrm{T}}$,
- Primitive: $\check{\mathbf{u}} \equiv [h, u]^{\mathrm{T}}$,
- Characteristic: $\hat{\mathbf{u}} \equiv [u - 2a, u + 2a]^{\mathrm{T}}$, where $a \equiv \sqrt{gh}$.

Section 2.8.2 gives the general FCT scheme for systems, with applicability to those FCT schemes limiting non-conservative sets of variables.

2.8.2 General FCT Scheme for Systems

The general FCT strategy is the same in the systems case as in the scalar case. One first defines antidiffusive correction fluxes such that

$$M_{i,i}^L \frac{\mathbf{U}_i^H - \mathbf{U}_i^n}{\Delta t} + \sum_j \mathbf{c}_{i,j} \cdot \mathrm{F}_j^n + \sum_j D_{i,j}^{L,n} \mathbf{U}_j^n = \mathbf{b}_i^n + \mathbf{p}_i \,. \tag{2.156}$$

Then subtracting the high-order scheme equation from this gives the definition of $\mathbf{p}$:

$$\mathbf{p}_i \equiv M_{i,i}^L \frac{\mathbf{U}_i^H - \mathbf{U}_i^n}{\Delta t} - \sum_j M_{i,j}^C \frac{\mathbf{U}_j^H - \mathbf{U}_j^n}{\Delta t} + \sum_j (D_{i,j}^{L,n} - D_{i,j}^{H,n}) \mathbf{U}_j^n \,. \tag{2.157}$$

As in the scalar case, these fluxes are decomposed into internodal fluxes $\mathbf{P}_{i,j}$ such that $\sum_j \mathbf{P}_{i,j} = \mathbf{p}_i$:

$$\mathbf{P}_{i,j} = -M_{i,j}^C \left(\frac{\mathbf{U}_j^H - \mathbf{U}_j^n}{\Delta t} - \frac{\mathbf{U}_i^H - \mathbf{U}_i^n}{\Delta t} \right) + (D_{i,j}^{L,n} - D_{i,j}^{H,n})(\mathbf{U}_j^n - \mathbf{U}_i^n) \,. \tag{2.158}$$

Applying a limiting coefficient to each internodal antidiffusive correction flux gives

$$M_{i,i}^L \frac{\mathbf{U}_i^{n+1} - \mathbf{U}_i^n}{\Delta t} + \sum_j \mathbf{c}_{i,j} \cdot \mathrm{F}_j^n + \sum_j D_{i,j}^{L,n} \mathbf{U}_j^n = \mathbf{b}_i^n + \sum_j \mathbf{L}_{i,j} \odot \mathbf{P}_{i,j} \,, \tag{2.159}$$

where the notation $\mathbf{L}_{i,j} \odot \mathbf{P}_{i,j}$ denotes an element-wise multiplication of $\mathbf{L}_{i,j}$ and $\mathbf{P}_{i,j}$: $(\mathbf{L}_{i,j} \odot \mathbf{P}_{i,j})^k = L_{i,j}^k P_{i,j}^k$.

As discussed in Section 2.8.1, FCT limitation for systems of conservation laws may benefit from transformations to other sets of variables such as primitive or characteristic variables. Consider some set of variables $\hat{\mathbf{u}}$, which is produced using a transformation matrix $\mathbf{T}(\mathbf{u})$: $\hat{\mathbf{u}} = \mathbf{T}^{-1}(\mathbf{u})\mathbf{u}$. For example, with the shallow water equations, to transform to the characteristic variables, one uses the matrix of right

eigenvectors $\mathbf{K}(\mathbf{u})$ of the Jacobian $\mathbf{A}(\mathbf{u})$ as the transformation matrix: $\mathbf{T}(\mathbf{u}) = \mathbf{K}(\mathbf{u})$. Applying a local transformation $\mathbf{T}^{-1}(\mathbf{U}_i^n)$ to Equation (2.156) gives

$$M_{i,i}^L \frac{\hat{\mathbf{U}}_i^{H,n+1} - \hat{\mathbf{U}}_i^n}{\Delta t} + \sum_j \mathbf{c}_{i,j} \cdot \hat{\mathrm{F}}_j^n + \sum_j D_{i,j}^{L,n} \hat{\mathbf{U}}_j^n = \hat{\mathbf{b}}_i^n + \sum_j \hat{\mathbf{P}}_{i,j} \,, \tag{2.160}$$

where accents denote transformed quantities:

$$\hat{\mathbf{U}}_j^{H,n+1} = \mathbf{T}^{-1}(\mathbf{U}_i^n)\mathbf{U}_j^{H,n+1} \,, \tag{2.161}$$

$$\hat{\mathbf{U}}_j^n = \mathbf{T}^{-1}(\mathbf{U}_i^n)\mathbf{U}_j^n \,, \tag{2.162}$$

$$\hat{\mathrm{F}}_j^n = \mathbf{T}^{-1}(\mathbf{U}_i^n)\mathrm{F}_j^n \,, \tag{2.163}$$

$$\hat{\mathbf{b}}_i^n = \mathbf{T}^{-1}(\mathbf{U}_i^n)\mathbf{b}_i^n \,. \tag{2.164}$$

$$\hat{\mathbf{P}}_{i,j} = \mathbf{T}^{-1}(\mathbf{U}_i^n)\mathbf{P}_{i,j} \,. \tag{2.165}$$

Applying a limiter $\hat{\mathbf{L}}$ produces some solution $\hat{\mathbf{U}}_i^{n+1}$:

$$M_{i,i}^L \frac{\hat{\mathbf{U}}_i^{n+1} - \hat{\mathbf{U}}_i^n}{\Delta t} + \sum_j \mathbf{c}_{i,j} \cdot \hat{\mathrm{F}}_j^n + \sum_j D_{i,j}^{L,n} \hat{\mathbf{U}}_j^n = \hat{\mathbf{b}}_i^n + \sum_j \hat{\mathbf{L}}_{i,j} \odot \hat{\mathbf{P}}_{i,j} \,, \tag{2.166}$$

One can then choose the limiting coefficients $\hat{\mathbf{L}}_{i,j}$ to satisfy bounds on the transformed variables:

$$\hat{W}_i^- \leq \hat{\mathbf{U}}_i^{n+1} \leq \hat{W}_i^+ \quad \forall i \,. \tag{2.167}$$

Imposition of these bounds corresponds to bounding the transformed antidiffusion flux sums:

$$\hat{\mathbf{Q}}_i^- \leq \sum_j \hat{\mathbf{L}}_{i,j} \odot \hat{\mathbf{P}}_{i,j} \leq \hat{\mathbf{Q}}_i^+ \,, \tag{2.168}$$

where the bounds are obtained by performing some algebra on the transformed sys-

tem:

$$\hat{\mathbf{Q}}_i^{\pm} \equiv M_{i,i}^L \frac{\hat{\mathbf{W}}_i^{\pm} - \hat{\mathbf{U}}_i^n}{\Delta t} + \sum_j \mathbf{c}_{i,j} \cdot \hat{\mathrm{F}}_j^n + \sum_j D_{i,j}^{L,n} \hat{\mathbf{U}}_j^n - \hat{\mathbf{b}}_i^n \,. \tag{2.169}$$

Zalesak's limiter then takes the same form as in the scalar case, but now the transformed quantities $\hat{\mathbf{Q}}^{\pm}$ and $\hat{\mathbf{P}}$ are used instead of $\mathbf{Q}^{\pm}$ and $\mathbf{P}$:

$$\hat{p}_i^{-,k} \equiv \sum_{j:\hat{P}_{i,j}^k<0} \hat{P}_{i,j}^k \,, \qquad \hat{p}_i^{+,k} \equiv \sum_{j:\hat{P}_{i,j}^k>0} \hat{P}_{i,j}^k \,, \tag{2.170}$$

$$\hat{L}_i^{\pm,k} \equiv \begin{cases} 1 & \hat{p}_i^{\pm,k} = 0 \\ \min\left(1, \frac{\hat{Q}_i^{\pm,k}}{\hat{p}_i^{\pm,k}}\right) & \hat{p}_i^{\pm,k} \neq 0 \end{cases} , \tag{2.171}$$

$$\hat{L}_{i,j}^k \equiv \begin{cases} \min(\hat{L}_i^{+,k}, \hat{L}_j^{-,k}) & \hat{P}_{i,j}^k \geq 0 \\ \min(\hat{L}_i^{-,k}, \hat{L}_j^{+,k}) & \hat{P}_{i,j}^k < 0 \end{cases} . \tag{2.172}$$

Kuzmin states that for the systems case, the limiting coefficients may require synchronization between the components, such as

$$\hat{L}_{i,j}^k \leftarrow\!\shortmid \min_{k'} \hat{L}_{i,j}^{k'} \quad \forall k \,. \tag{2.173}$$

Otherwise, antidiffusive fluxes in one component can violate the conditions of another component [19].

After these limiting coefficients $\hat{\mathbf{L}}_{i,j}$ for the transformed system are computed, one simply uses $\mathbf{L}_{i,j} = \hat{\mathbf{L}}_{i,j}$ in the original, conservative scheme given by Equation (2.159). In this way, the transformed fluxes $\hat{\mathbf{P}}_{i,j}$, which do not have the conservative property $\hat{\mathbf{P}}_{i,j} = -\hat{\mathbf{P}}_{j,i}$, are only used to compute $\mathbf{L}_{i,j}$, and the resulting scheme is still conservative.

2.9 Implementation

2.9.1 Nonlinear Iteration

In this research, implicit and steady-state temporal discretizations are considered for the scalar case only. The examples given in this section thus correspond to the scalar case; however, the same methodology could be used for systems of conservation laws.

The nonlinear systems considered in this research can be written in a quasi-linear form:

$$\mathbf{B}(\mathbf{U})\mathbf{U} = \mathbf{s}(\mathbf{U})\,. \tag{2.174}$$

The right-hand-side $\mathbf{s}(\mathbf{U})$ is a function of the solution for FCT schemes since in general the limiting coefficients are functions of the solution. A system in which Dirichlet boundary conditions are strongly imposed requires modification of the matrix and right-hand-side vector: $\mathbf{B} \rightarrow \tilde{\mathbf{B}}$ and $\mathbf{s} \rightarrow \tilde{\mathbf{s}}$. In the remainder of this section, the Dirichlet-modified notation is used to be general; if strong Dirichlet boundary conditions are not applied, then the Dirichlet-modified notation can simply be ignored. Subsequent sections will define the matrix $\mathbf{B}$ and $\mathbf{s}$ for different nonlinear schemes.

To solve this system, fixed point iteration will be used:

$$\tilde{\mathbf{B}}^{(\ell)}\mathbf{U}^{(\ell+1)} = \tilde{\mathbf{s}}^{(\ell)}\,, \tag{2.175}$$

where $\mathbf{B}^{(\ell)} \equiv \mathbf{B}(\mathbf{U}^{(\ell)})$ and $\mathbf{s}^{(\ell)} \equiv \mathbf{s}(\mathbf{U}^{(\ell)})$. However, in implementation, this will be expressed as a defect correction scheme:

$$\mathbf{r}^{(\ell)} = \tilde{\mathbf{s}}^{(\ell)} - \tilde{\mathbf{B}}^{(\ell)}\mathbf{U}^{(\ell)}\,, \tag{2.176}$$

$$\tilde{\mathbf{B}}^{(\ell)}\Delta\mathbf{U}^{(\ell+1)} = \mathbf{r}^{(\ell)}\,, \tag{2.177}$$

$$\mathbf{U}^{(\ell+1)} = \mathbf{U}^{(\ell)} + \Delta\mathbf{U}^{(\ell+1)}\,. \tag{2.178}$$

There are a number of advantages to this approach. Firstly this approach uses the linear residual $\mathbf{r}^{(\ell)}$, which is advantageous when checking for convergence; for example, there are a number of caveats associated with checking convergence based on some measure of the difference between solution iterates $\mathbf{U}^{(\ell)}$ and $\mathbf{U}^{(\ell+1)}$.

Another advantage of the defect correction approach is the ease in implementing a relaxation parameter α:

$$\mathbf{U}^{(\ell+1)} = \mathbf{U}^{(\ell)} + \alpha\Delta\mathbf{U}^{(\ell+1)}\,. \tag{2.179}$$

The pseudo-code for this defect correction scheme is given in Algorithm 1. The notation $\|\mathbf{r}\|_X$ denotes some norm X of $\mathbf{r}$. The initial guess is denoted by $\mathbf{U}^{\text{guess}}$. For transient calculations, the previous time step solution $\mathbf{U}^n$ is often used as the guess, and for steady-state calculations, one could use a zero vector as the initial guess.

Algorithm 1 Defect Correction Algorithm

$\mathbf{U}^{(0)} \leftarrow \mathbf{U}^{\text{guess}}$

`converged` $\leftarrow$ `FALSE`

for $\ell \leftarrow 0, \ell_{\max}$ **do**

  $\mathbf{B}^{(\ell)} \leftarrow \mathbf{B}(\mathbf{U}^{(\ell)})$

  $\mathbf{s}^{(\ell)} \leftarrow \mathbf{s}(\mathbf{U}^{(\ell)})$

  $\mathbf{B}^{(\ell)} \rightarrow \tilde{\mathbf{B}}^{(\ell)}$, $\mathbf{s}^{(\ell)} \rightarrow \tilde{\mathbf{s}}^{(\ell)}$,

  $\mathbf{r}^{(\ell)} \leftarrow \tilde{\mathbf{s}}^{(\ell)} - \tilde{\mathbf{B}}^{(\ell)}\mathbf{U}^{(\ell)}$

  if $\|\mathbf{r}^{(\ell)}\|_X < \epsilon$ **then**

    `converged` $\leftarrow$ `TRUE`

    break

  end if

  $\Delta\mathbf{U}^{(\ell+1)} \leftarrow \left[\tilde{\mathbf{B}}^{(\ell)}\right]^{-1}\mathbf{r}^{(\ell)}$

  $\mathbf{U}^{(\ell+1)} \leftarrow \mathbf{U}^{(\ell)} + \alpha\Delta\mathbf{U}^{(\ell+1)}$

end for

if not `converged` **then**

  error: Solution did not converge within $\ell_{\max}$ iterations

end if

Table 2.2 gives the definitions of the system matrix and right-hand-side vector given by Equation (2.174) for a number of different nonlinear schemes. These schemes include schemes that use entropy viscosity and those that use FCT. For the equations of transient schemes, the solution vector $\mathbf{U}^{n+1}$ uses $\mathbf{U}$ as its notation; the superscript $n+1$ is understood. This is to be consistent with the convention that $\mathbf{U}$ is the vector iterated upon in the nonlinear scheme.

Note that in Table 2.2, the low-order steady-state matrix $\mathbf{A}^L$ is assumed to be independent of the solution. This is true for conservation laws with *linear* conservation law flux functions $\mathbf{f}(u)$. For nonlinear $\mathbf{f}(u)$, this is not the case, and the nonlinear system matrix $\mathbf{B}$ for the FCT schemes given in the table becomes a function of the solution and thus must be recomputed in each iteration.

Table 2.2: Nonlinear System Matrix and Right-hand-side Vector for Different Schemes

Steady-State Entropy Viscosity Scheme	
Equation	$\mathbf{A}^H(\mathbf{U})\mathbf{U} = \mathbf{b}$
Matrix	$\mathbf{B}(\mathbf{U}) \equiv \mathbf{A}^H(\mathbf{U})$
Right-hand-side	$\mathbf{s} \equiv \mathbf{b}$
Steady-State FCT Scheme	
Equation	$\mathbf{A}^L\mathbf{U} = \mathbf{b} + \mathbf{L}(\mathbf{U}) \cdot \mathbf{P}$
Matrix	$\mathbf{B} \equiv \mathbf{A}^L$
Right-hand-side	$\mathbf{s}(\mathbf{U}) \equiv \mathbf{b} + \mathbf{L}(\mathbf{U}) \cdot \mathbf{P}$
Theta Entropy Viscosity Scheme	
Equation	$\mathbf{M}^C\frac{\mathbf{U}-\mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}^{H,n}\mathbf{U}^n + \theta\mathbf{A}^H(\mathbf{U})\mathbf{U} = (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1}$
Matrix	$\mathbf{B}(\mathbf{U}) \equiv \mathbf{M}^C + \theta\Delta t\mathbf{A}^H(\mathbf{U})$
Right-hand-side	$\mathbf{s} \equiv \mathbf{M}^C\mathbf{U}^n - (1-\theta)\Delta t\mathbf{A}^{H,n}\mathbf{U}^n + (1-\theta)\Delta t\mathbf{b}^n + \theta\Delta t\mathbf{b}^{n+1}$
Theta FCT Scheme	
Equation	$\mathbf{M}^L\frac{\mathbf{U}-\mathbf{U}^n}{\Delta t} + (1-\theta)\mathbf{A}^L\mathbf{U}^n + \theta\mathbf{A}^L\mathbf{U} = (1-\theta)\mathbf{b}^n + \theta\mathbf{b}^{n+1} + \mathbf{L}(\mathbf{U}) \cdot \mathbf{P}$
Matrix	$\mathbf{B} \equiv \mathbf{M}^L + \theta\Delta t\mathbf{A}^L$
Right-hand-side	$\mathbf{s}(\mathbf{U}) \equiv \mathbf{M}^L\mathbf{U}^n - (1-\theta)\Delta t\mathbf{A}^L\mathbf{U}^n + (1-\theta)\Delta t\mathbf{b}^n + \theta\Delta t\mathbf{b}^{n+1} + \Delta t\mathbf{L}(\mathbf{U}) \cdot \mathbf{P}$

2.9.2 FCT Nonlinear Iteration

For implicit time discretizations, such as the Theta discretization and steady-state, the FCT algorithm is nonlinear in general because the solution bounds are implicit in general. However, if one chooses solution bounds that do not depend on the FCT solution, then the resulting system of equations is only nonlinear if both the conservation law is nonlinear and an implicit time discretization is used.

This section considers nonlinear FCT systems in which the solution bounds depend on the FCT solution. With the presence of conditional statements in the computation of the limiting coefficients, the computation of the Jacobian of the nonlinear system is complicated, and thus instead of a Newton iteration, a defect correction scheme such as that described in Section 2.9.1 is considered here.

Suppose that in a particular iteration of a nonlinear FCT scheme, one aims to find the FCT solution iterate $\mathbf{U}^{(\ell+1)}$. In the nonlinear FCT case, the solution bounds depend on the solution: $W_i^{\pm}(\mathbf{U})$, so one needs to consider which bounds are actually being imposed in that iteration. Ideally, one would like to impose the solution bounds $W_i^{\pm}(\mathbf{U}^{(\ell+1)})$; however, $\mathbf{U}^{(\ell+1)}$ is not available. Therefore one must employ previous solution iterates such as $\mathbf{U}^{(\ell)}$. Thus the imposed solution bounds are the following:

$$W_i^{-}(\mathbf{U}^{(\ell)}) \leq U_i^{(\ell+1)} \leq W_i^{+}(\mathbf{U}^{(\ell)})\,. \tag{2.180}$$

As shown in Section 2.7.3, the antidiffusion bounds $Q_i^{\pm}$ are defined by the time discretization and are functions of the solution bounds. As an example, the steady-state antidiffusion bounds from Equation (2.141) for an iteration would be

$$Q_i^{\pm} = A_{i,i}^L W_i^{\pm}(\mathbf{U}^{(\ell)}) + \sum_{j \neq i} A_{i,j}^L U_j^{(\ell+1)} - b_i\,. \tag{2.181}$$

However, this reveals an issue: the antidiffusion bounds definition depends on $\mathbf{U}^{(\ell+1)}$, which again is not available in a fixed-point-type iteration scheme. Thus one must use another definition for the antidiffusion bounds, such as the following for steady-state:

$$Q_i^{\pm,(\ell)} \equiv A_{i,i}^L W_i^{\pm}(\mathbf{U}^{(\ell)}) + \sum_{j \neq i} A_{i,j}^L U_j^{(\ell)} - b_i \,. \tag{2.182}$$

Unfortunately, the transition from Equation (2.181) to (2.182) implies that there is no longer a guarantee that the FCT bounds at each iteration, given by Equation (2.180), are satisfied. Effectively, the modified antidiffusion bounds given by Equation (2.182) correspond to different solution bounds:

$$\tilde{W}_i^{-,(\ell)} \leq U_i^{(\ell+1)} \leq \tilde{W}_i^{+,(\ell)} \,, \tag{2.183a}$$

$$\tilde{W}_i^{\pm,(\ell)} \equiv W_i^{\pm}(\mathbf{U}^{(\ell)}) + \sum_{j \neq i} A_{i,j}^L (U_j^{(\ell)} - U_j^{(\ell+1)}) \,. \tag{2.183b}$$

However, upon convergence, the original solution bounds are satisfied because upon convergence, Equations (2.181) and (2.182) are equal.

3. RESULTS

3.1 Overview

This chapter presents the results of the methods described in this dissertation. The methods were implemented using the `deal.II` finite element library[4]. Section 3.2 gives results for the scalar transport equation, and Section 3.3 gives results for the shallow water equations.

3.2 Scalar Transport

3.2.1 Overview

This section presents results for scalar transport. Section 3.2.2 shows convergence results that demonstrate second-order spatial accuracy for the entropy viscosity (EV) method, as well as the FCT scheme. Results in subsequent sections give various results exploring a number of dimensions of EV and FCT schemes.

The FCT algorithm can be built on either the entropy viscosity method or the standard Galerkin high-order scheme given in Section 2.2.1. The resulting schemes will be referred to in this section as the EV-FCT scheme and the Galerkin-FCT scheme, respectively.

Incoming flux boundary conditions can be applied in a number of different ways, as discussed in Section 2.2.1.1. When these boundary conditions are applied as strong Dirichlet, recall from Section 2.7.1 that the antidiffusion flux decomposition does not apply in the neighborhood of nodes for which Dirichlet boundary conditions are strongly imposed. Therefore, to keep the conservation property, antidiffusive fluxes

involving Dirichlet nodes must be completely canceled:

$$L_{i,j} \leftarrow 0\,, \quad \forall j\,, \forall i \in \mathcal{I}^{\text{inc}}\,. \tag{3.1}$$

Recall from Section 2.7.4.1 that Zalesak's limiter computes upper bounds on the limiting coefficients for all positive and negative antidiffusive fluxes for a node i: L_i^+ and L_i^-. Then for a positive antidiffusive flux $P_{i,j}$, the limiting coefficient is $L_{i,j} = \min(L_i^+, L_j^-)$. Thus to cancel the antidiffusive fluxes for Dirichlet nodes, one can perform the following operation:

$$L_i^+ \leftarrow 0\,, \quad L_i^- \leftarrow 0\,, \quad \forall i \in \mathcal{I}^{\text{inc}}\,. \tag{3.2}$$

If one does not deem the conservation property important, then one may instead decide that antidiffusive fluxes from Dirichlet nodes should not necessarily be canceled and thus may instead perform the following:

$$L_i^+ \leftarrow 1\,, \quad L_i^- \leftarrow 1\,, \quad \forall i \in \mathcal{I}^{\text{inc}}\,. \tag{3.3}$$

This has the advantage that more antidiffusion can be accepted and thus that the solution may have greater accuracy. These two paradigms are considered for a number of test problems and will typically be labeled as "strong Dirichlet BC with $L_i^- = L_i^+ = 0$" and "strong Dirichlet BC with $L_i^- = L_i^+ = 1$", respectively. Other options considered include weak Dirichlet and weak Dirichlet with a boundary penalty. When results in this section use a boundary penalty, the penalty coefficient (as described in Section 2.2.1.1 takes the value $\alpha = 1000$.

A number of options for the solution bounds to impose in FCT will be considered. Firstly, the low-order DMP bounds given in Section 2.3.5 for each tempo-

ral discretization are considered. Alternatively, analytic bounds derived from the method of characteristics (MoC) in Appendix A are used. The analytic bounds are derived along an upwind line segment $L(\mathbf{x}, \mathbf{x} - v\Delta t\boldsymbol{\Omega})$; this is referred to as the "*upwind* analytic solution bounds". However, a more relaxed set of solution bounds are used as the default solution bounds, which considers a spherical neighborhood around each node instead of just the upwind line segment; this will be referred to as the "analytic solution bounds". Certain sections will also consider a modification of these bounds, referred to as the "modified analytic solution bounds", which as of yet has no analytic proof. These modified bounds are discussed when used. Note that the corresponding antidiffusion bounds $Q_i^{\pm}$ for a given set of solution bounds $W_i^{\pm}$ must obey the conditions discussed in Section 2.7.3.2, and thus the following operation given by Equation (2.151) is performed for all results in this section:

$$Q_i^- \leftarrow \min(Q_i^-, 0)\,,$$

$$Q_i^+ \leftarrow \max(Q_i^+, 0)\,.$$

Some results in this section also try the multi-pass limiting procedure described in Section 2.7.4.2.

For transient simulations, the time step size used is given as a "CFL" number ν, which is defined in terms of the maximum time step size for Explicit Euler time discretization of the low-order scheme, given by Equation (2.84):

$$\nu \equiv \frac{\Delta t}{\Delta t_{\text{CFL}}}\,, \quad \Delta t_{\text{CFL}} \equiv \min_i \frac{M_{i,i}^L}{A_{i,i}^{L,n}}\,. \tag{3.5}$$

Note that the time step size for the Theta method is less restrictive for $\theta > 0$ (see Equation (2.85)); however, for Theta method results, the definition of ν above is still

used.

In summary, results in this section have the following dimensions, where the default options in this section are underlined:

- temporal discretization
- time step size/CFL number
- mesh size
- BC method:
  - strong Dirichlet BC with $L_i^- = L_i^+ = 0$
  - <u>strong Dirichlet BC with $L_i^- = L_i^+ = 1$</u>
  - weak Dirichlet BC
  - weak Dirichlet BC with a boundary penalty
- FCT type:
  - <u>EV-FCT</u>
  - Galerkin-FCT
- FCT solution bounds:
  - low-order DMP bounds
  - <u>analytic bounds from method of characteristics</u>
  - *modified* analytic bounds from method of characteristics
  - upwind analytic bounds from method of characteristics
- FCT limiter:

  - single-pass Zalesak
  - multi-pass Zalesak

- Initial guess for implicit FCT and steady-state FCT solutions:
  - zeroes: $u^{(0)} = 0$
  - low-order solution: $u^{(0)} = u^L$
  - high-order solution: $u^{(0)} = u^H$

Section 3.2.12 will attempt to make some conclusions about some of these options.

3.2.2 Convergence Studies

3.2.2.1 Overview

A number of test problems were used to evaluate spatial and temporal convergence rates. Before describing these problems, the methodology for evaluating these rates will be described. The error in the numerical solution $\tilde{u}(\mathbf{x}, t)$ has a number of components:

- Spatial discretization error,
- Temporal discretization error, and
- Computer precision (round-off) error.

Round-off error arises from limited precision of floating point numbers and becomes relevant here only when measuring errors on the order of the computer precision, $\sim \mathcal{O}(10^{-15})$; this becomes the bottleneck of improvement of convergence at fine refinements. For larger magnitude error, the important components are thus spatial and temporal discretization error.

Of course, for steady-state problems, there is no temporal discretization error. Omitting higher order terms, for a steady-state problem, the error of the approximate solution in some norm has the form

$$e = c_x \Delta x^m \,, \tag{3.6}$$

where Δx is the spatial element size, m is the spatial convergence rate, and c_x is the leading coefficient. Taking the logarithm of this equation gives

$$\log(e) = m \log(\Delta x) + c \,, \tag{3.7}$$

where c is some constant. Thus one can make two measurements $(\Delta x_i, e_i)$ and $(\Delta x_{i+1}, e_{i+1})$ to compute a slope $m_{i+\frac{1}{2}}$ on a log-log plot to estimate the convergence rate:

$$m_{i+\frac{1}{2}} = \frac{\log(e_{i+1}) - \log(e_i)}{\log(\Delta x_{i+1}) - \log(\Delta x_i)} \,. \tag{3.8}$$

For time-dependent solutions, the situation is more complicated:

$$e = c_x \Delta x^m + c_t \Delta t^p \,, \tag{3.9}$$

where Δt is the time step size, p is the temporal convergence rate, and c_t is the leading coefficient for temporal error. Taking the logarithm of both sides of this equation does not yield the same linear relationship as in the steady-state case:

$$\log(e) = \log(c_x \Delta x^m + c_t \Delta t^p) \,. \tag{3.10}$$

One may be interested in measuring the spatial convergence rate m, but temporal errors may prevent the rate from being recovered. Similarly, One may be interested

in measuring the temporal convergence rate p, but spatial errors may prevent the rate from being recovered. There are three main strategies for overcoming this difficulty:

- **For measuring spatial/temporal convergence rates, choose a test problem such that no temporal/spatial error arises.** For example, for measuring convergence rates, one can choose a problem with a solution that is not a function of time or that is linear in time, since the time discretization should be able to exactly integrate linear functions of time. Similarly, if one knows that a spatial discretization can exactly approximate a linear solution, then there is no spatial error and thus temporal convergence rates can be measured.

- **For measuring spatial/temporal convergence rates, use a very fine temporal/spatial refinement level.** This is an obvious approach; however, this is often undesirable because it is a computationally costly approach.

- **Refine both space and time using knowledge of expected convergence rates.** The idea of this approach is to use a certain relation between mesh size and time step size to recover either the spatial or temporal convergence rate. If one would like to recover the spatial convergence rate m, then one can assume that time step size has the relation

$$\Delta t^p = \Delta x^m \,, \tag{3.11}$$

so that $\Delta t = \Delta x^{m/p}$. Then Equation (3.10) becomes

$$\log(e) = \log(c_x \Delta x^m + c_t \Delta x^m) = m \log(\Delta x) + c \,. \tag{3.12}$$

Similarly one can use the relation $\Delta x = \Delta t^{p/m}$ to recover

$$\log(e) = \log(c_x \Delta t^p + c_t \Delta t^p) = p \log(\Delta t) + c\,. \tag{3.13}$$

Then rates can be measured just as given in Equation (3.8). For example, suppose that one wishes to measure temporal convergence rate and refines mesh size by a factor of $\frac{1}{2}$ in each cycle: $\Delta x_{i+1} = \frac{1}{2}\Delta x_i$. Suppose that the spatial discretization is of order m and the temporal discretization is supposedly of order p. Then using $\Delta t_{i+1} = \Delta x_{i+1}^{m/p} = (\frac{1}{2}\Delta x_i)^{m/p} = (\frac{1}{2})^{m/p}\Delta t_i$. Therefore the temporal refinement factor should be $(\frac{1}{2})^{m/p}$. Then the temporal convergence rate can be measured:

$$p_{i+\frac{1}{2}} = \frac{\log(e_{i+1}) - \log(e_i)}{\log(\Delta t_{i+1}) - \log(\Delta t_i)}\,. \tag{3.14}$$

The first and third of these approaches are used in sections that follow.

3.2.2.2 Convergence Study 1

This test problem uses a time-independent manufactured solution to isolate spatial errors: $u(x) = \sin(\pi x)$. The problem summary is given in Table 3.1.

This problem was run in steady-state to avoid temporal error so that spatial convergence rates could accurately be measured. The coarsest mesh size in this study uses 8 cells, and each successive mesh size is halved, with the finest mesh in the study using 256 cells. The run parameters for this test problem are given in Table 3.2.

Figure 3.1 shows a comparison of the solutions with 32 cells, and Figure 3.2 compares the viscosity profiles. Only the low-order (DMP) method does not match the exact solution well and suffers a defect at the outflow (right) boundary. The

Table 3.1: Convergence Test Problem 1 Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0, 1)$
Boundary Conditions	$u(x) = 0, \quad x \in \partial\mathcal{D}^-,$ $\partial\mathcal{D}^- = \{x \in \partial\mathcal{D} : \mathbf{n}(x) \cdot \mathbf{\Omega} < 0\}$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(x) = 1$
Source	$q(x) = \pi \cos(\pi x) + \sin(\pi x)$
Speed	$v = 1$
Exact Solution	$u(x) = \sin(\pi x)$

Table 3.2: Convergence Test Problem 1 Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 8, 16, \mathbf{32}, 64, 128, 256$
Time Discretization	Steady-State
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	**Analytic**, DMP
FCT Initial Guess	$u^{(0)} = u^H$

entropy viscosity and Galerkin FCT methods use the analytic solution bounds given by Equation (A.10). Since in the steady-state case the FCT solution bounds are implicit, the system is nonlinear and requires iteration. To produce the results that follow, the high-order solution was used as the initial guess in this iteration. It should be noted that for this test problem, using other guesses, such as zero or the low-order solution, leads to serious nonlinear convergence issues, for which a remedy is not currently known.

Figure 3.3 shows a comparison of errors with different methods. The DMP low-

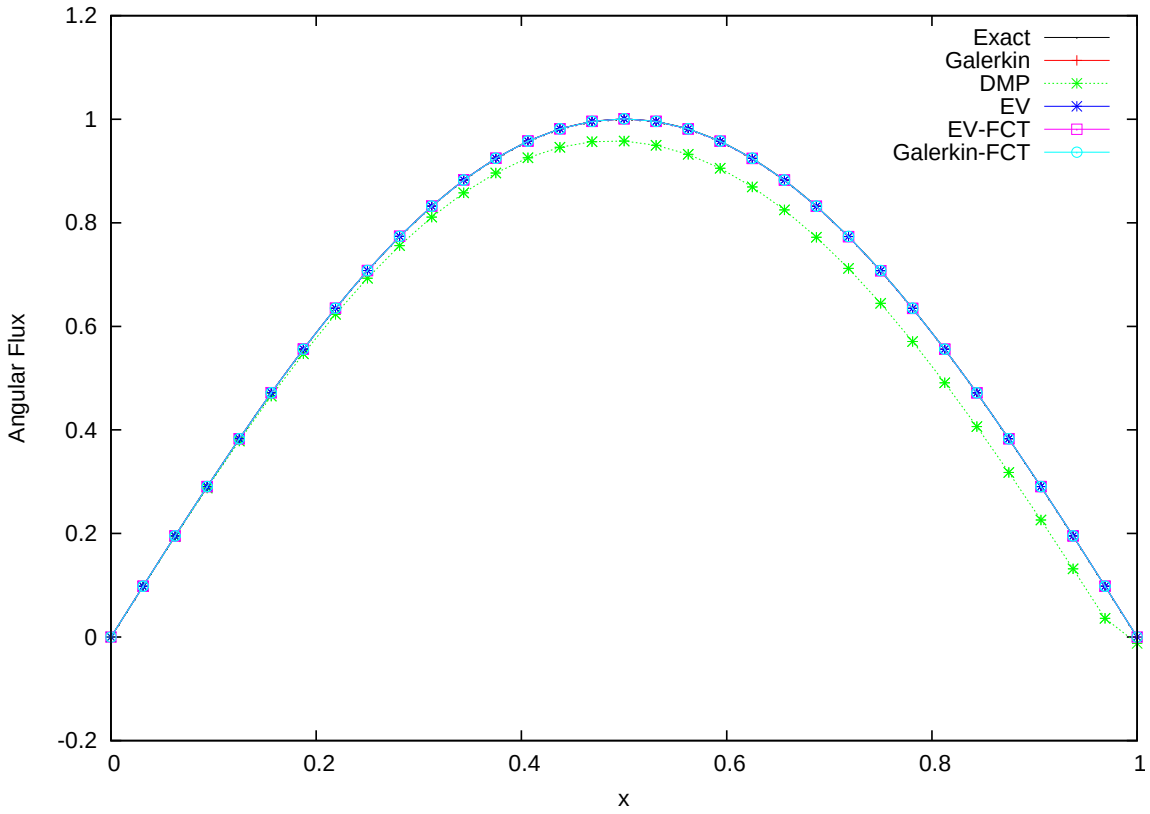


Figure 3.1: Comparison of Solutions for Convergence Test Problem 1 with 32 Cells

order method achieves first-order spatial convergence as expected, and all other methods achieve second-order spatial convergence. The entropy viscosity method and the FCT method employing it both start with more error for the coarsest mesh than the Galerkin method, but upon refinement, the differences in error diminish.

If the FCT methods use the low-order discrete maximum principle expressions given by Equation (2.87) as solution bounds instead of the analytic solution bounds, then results are much different for the FCT schemes. Figure 3.4 shows a comparison of the entropy viscosity FCT solutions with 32 cells using the analytic bounds vs. using the low-order DMP bounds, and Figure 3.5 shows a comparison of the errors. For this test problem, using the low-order DMP for the FCT solution bounds results

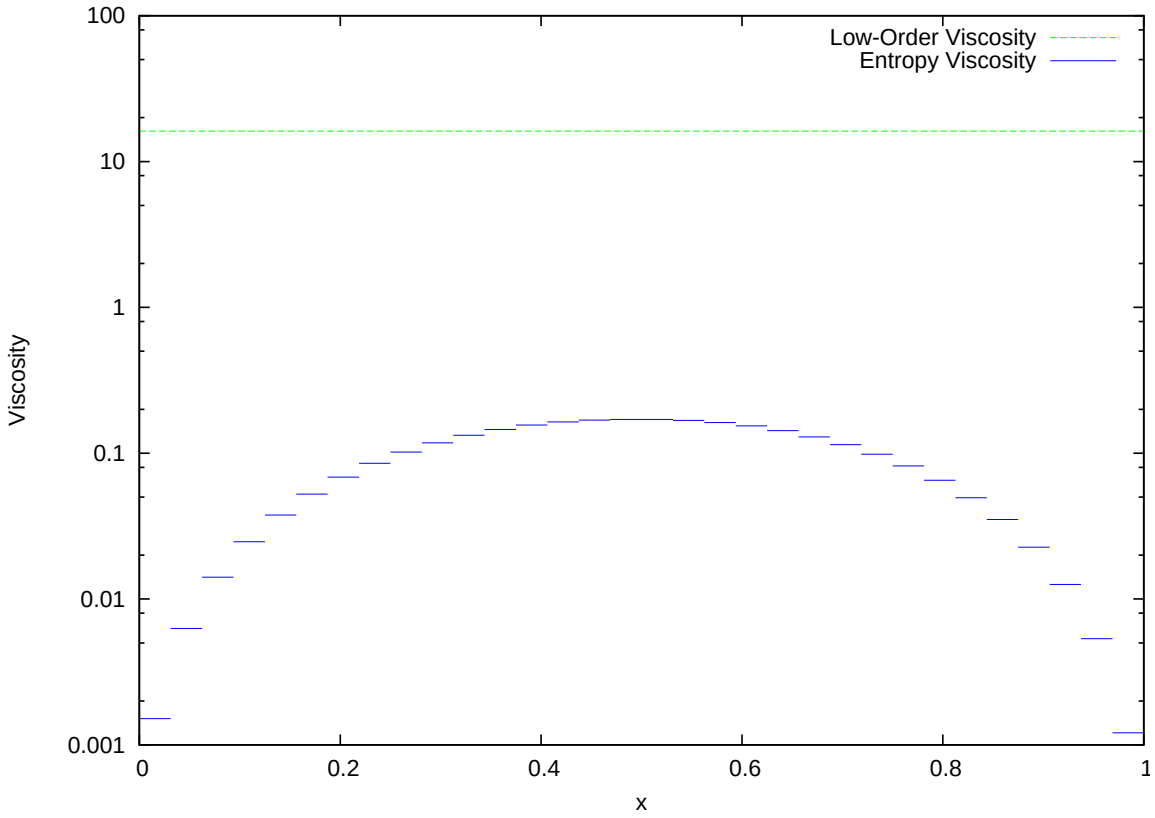


Figure 3.2: Viscosity Profiles for Convergence Test Problem 1 with 32 Cells

in reducing the FCT solution to the low-order solution. From this example it is evident that the selection of solution bounds is critical to the success of the FCT algorithm.

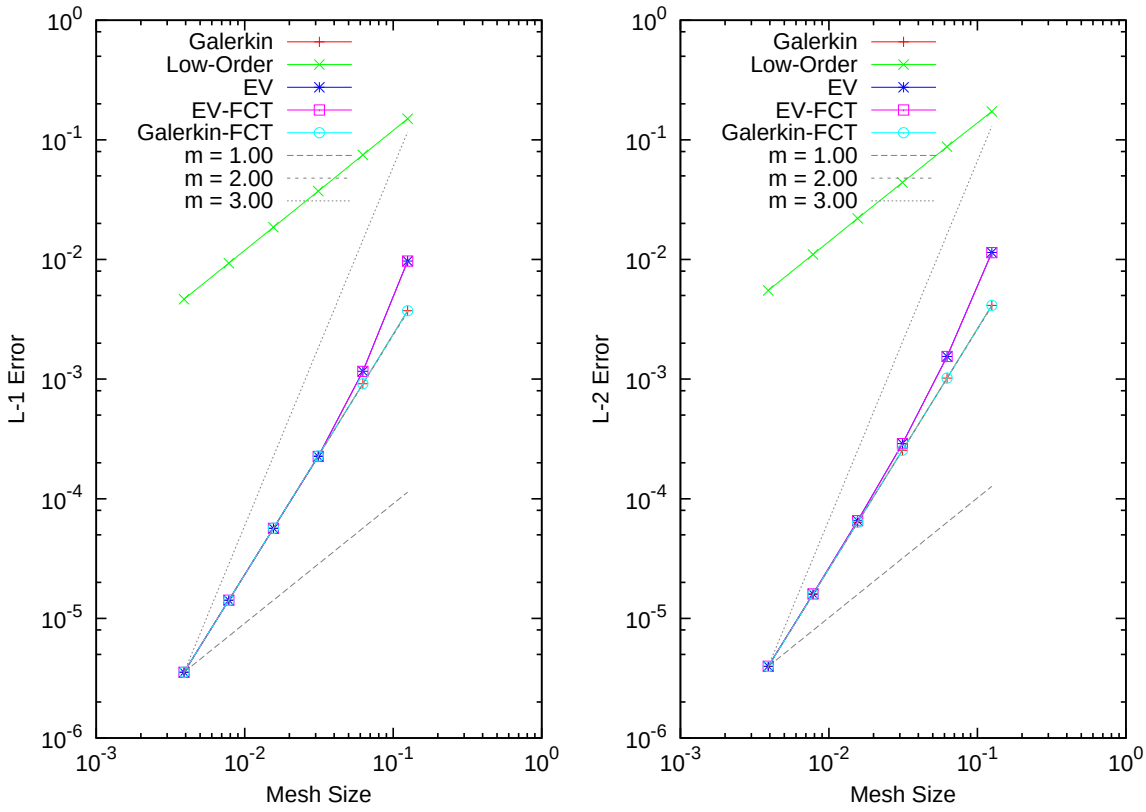


Figure 3.3: Comparison of Errors for Convergence Test Problem 1

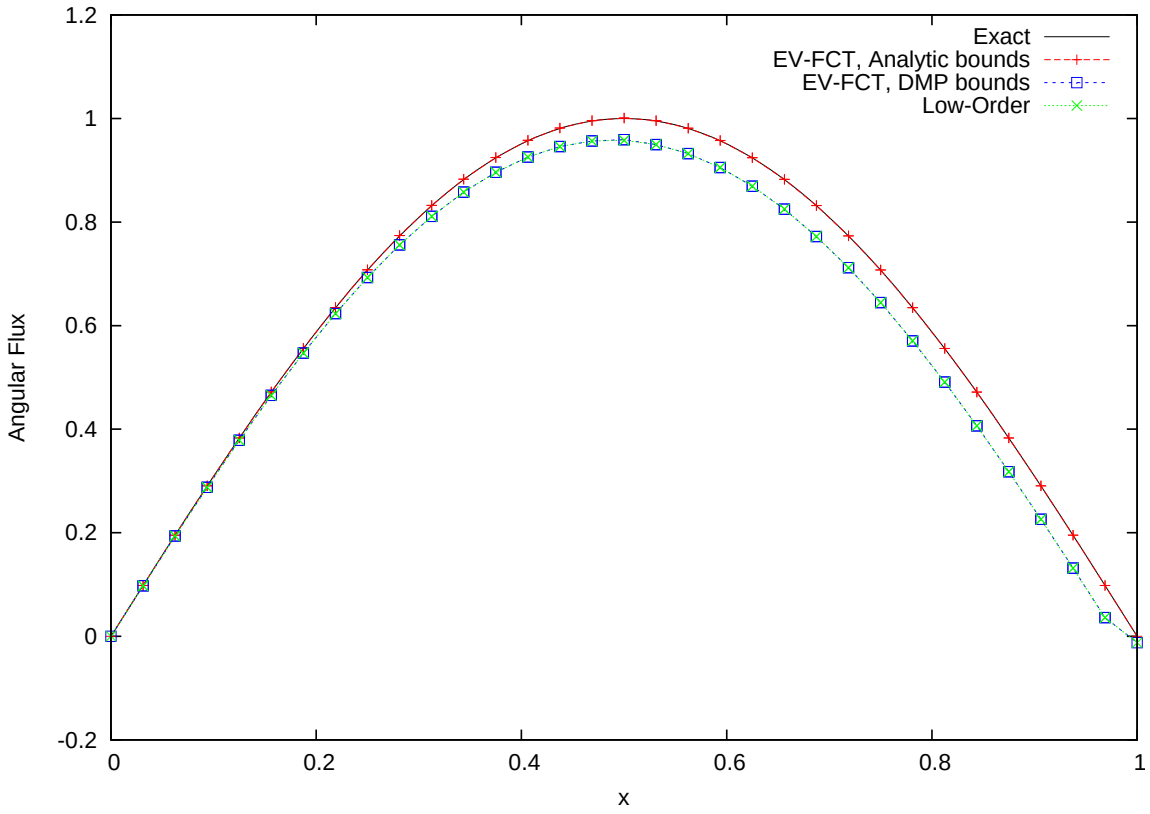


Figure 3.4: Comparison of FCT Solutions with Different Solution Bounds with 32 Cells

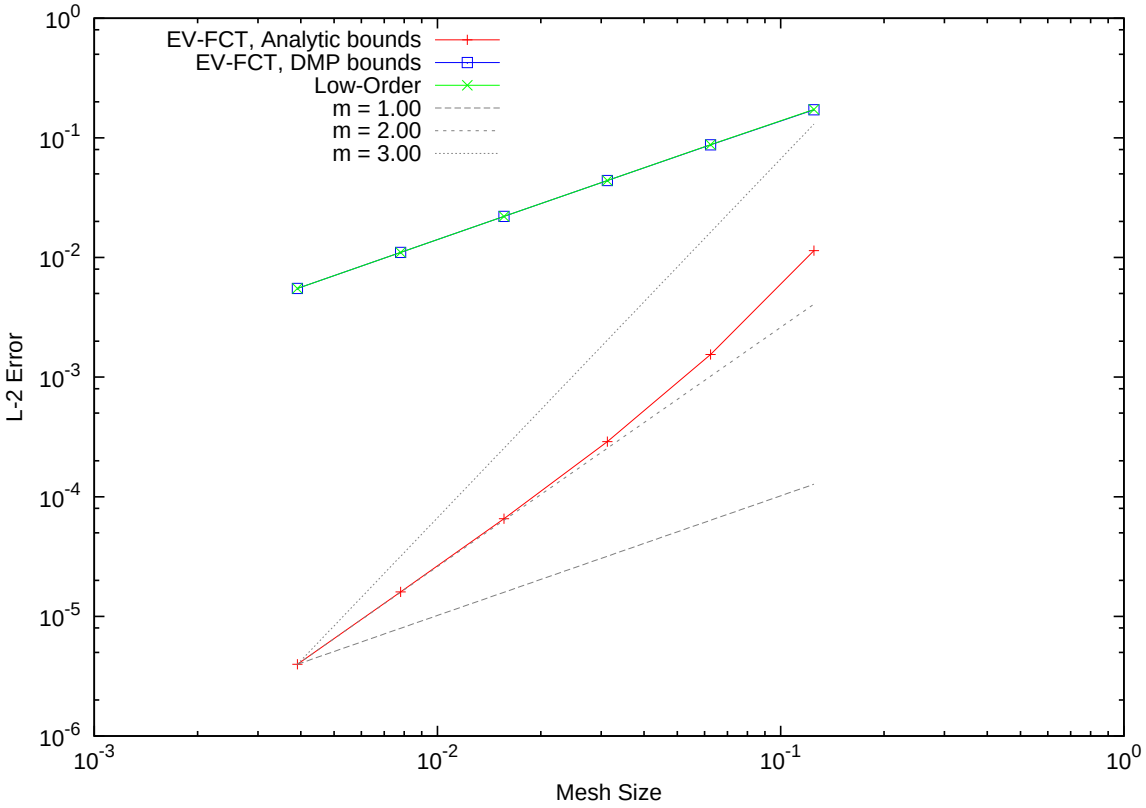


Figure 3.5: Comparison of Error of FCT Solutions with Different Solution Bounds

3.2.2.3 Convergence Study 2

This test problem is for a homogeneous absorber medium with an incident flux, which has a simple exponential decay solution. The problem summary is given in Table 3.3.

Table 3.3: Convergence Test Problem 2 Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0, 1)$
Boundary Conditions	$u(x) = 1, \quad x \in \partial\mathcal{D}^-,$ $\partial\mathcal{D}^- = \{x \in \partial\mathcal{D} : \mathbf{n}(x) \cdot \mathbf{\Omega} < 0\}$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(x) = 10$
Source	$q(x) = 0$
Speed	$v = 1$
Exact Solution	$u(x) = e^{-10x}$

This problem was run in steady-state to avoid temporal error so that spatial convergence rates could accurately be measured. The coarsest mesh size in this study uses 8 cells, and each successive mesh size is halved, with the finest mesh in the study using 256 cells. The run parameters for this test problem are given in Table 3.4.

Figure 3.6 shows a comparison of the solutions with 32 cells, and Figure 3.7 shows the corresponding viscosity profiles. The FCT methods use the analytic solution bounds given by Equation (A.10).

Figure 3.8 shows a comparison of errors with different methods. The DMP low-order method achieves first-order spatial convergence as expected, and the Galerkin and Galerkin-FCT methods achieve second-order accuracy. The entropy viscosity

Table 3.4: Convergence Test Problem 2 Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 8, 16, \mathbf{32}, 64, 128, 256$
Time Discretization	Steady-State
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	Analytic
FCT Initial Guess	$u^{(0)} = u^H$

and entropy-viscosity-FCT (EV-FCT) methods start out with more error than the Galerkin methods in the coarse meshes, but converge with order greater than 2 until the finer meshes, where the convergence of the EV methods asymptotically approach that of the Galerkin methods.

Figure 3.9 shows the convergence of the L^2 norm entropy residual $\mathcal{R}$ and entropy jumps $\mathcal{J}$, which are computed as

$$\|\mathcal{R}\|_{L^2(\mathcal{D})} = \sqrt{\int_{\mathcal{D}} \mathcal{R}(\mathbf{x})^2 \, d\mathbf{x}}\,, \tag{3.15}$$

$$\|\mathcal{J}\|_{L^2(\mathcal{D})} = \sqrt{\int_{\mathcal{D}} \mathcal{J}(\mathbf{x})^2 \, d\mathbf{x}} = \sqrt{\sum_K \mathcal{J}_K^2 |\mathcal{D}_K|}\,, \tag{3.16}$$

where $\mathcal{J}(\mathbf{x})$ is the piecewise constant function such that $\mathcal{J}(\mathbf{x}) = \mathcal{J}_K$ for $\mathbf{x} \in \mathcal{D}_K$. For this study, the coarsest mesh consists of 2 cells, and there are a total of 12 refinement levels, so the final refinement level corresponds to 2^12 cells. The entropy residual L^2 norm converges first-order as expected. The entropy jumps L^2 norm actually increases for the coarsest meshes, until meshes finer than roughly 32 cells, where it

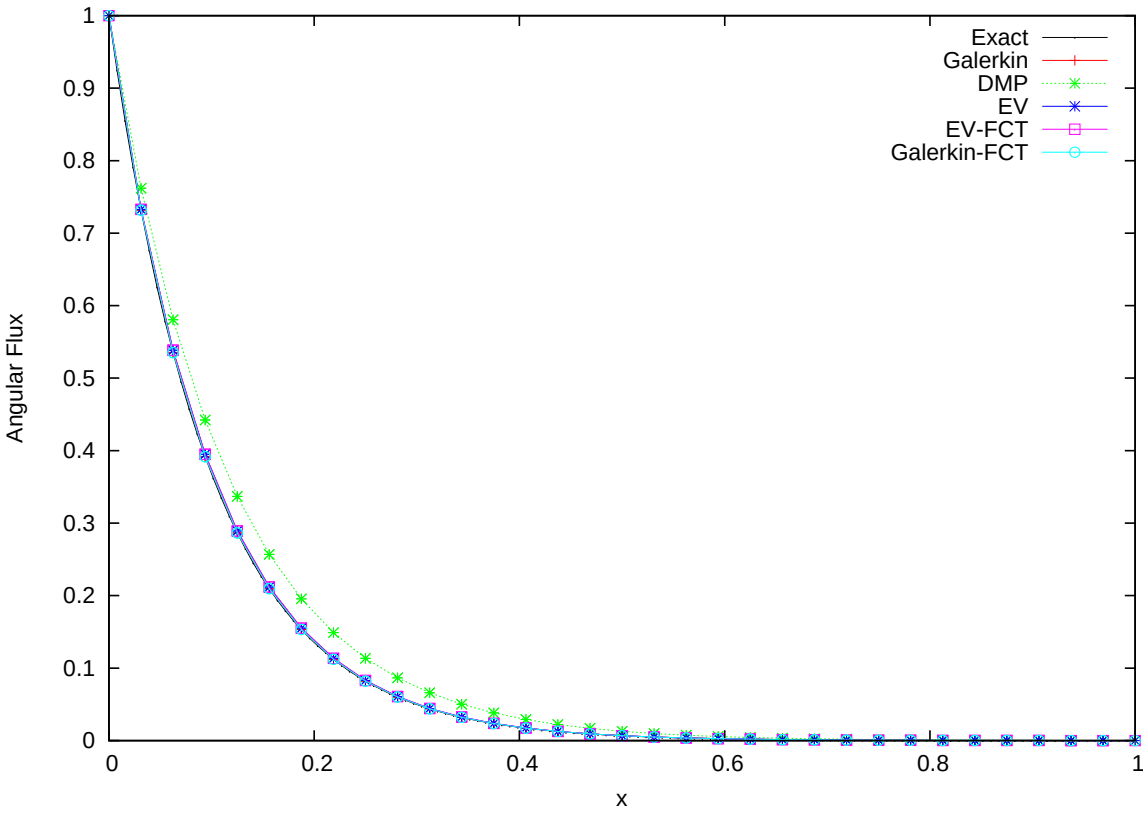


Figure 3.6: Comparison of Solutions for Convergence Test Problem 2 with 32 Cells

transitions to first-order convergence.

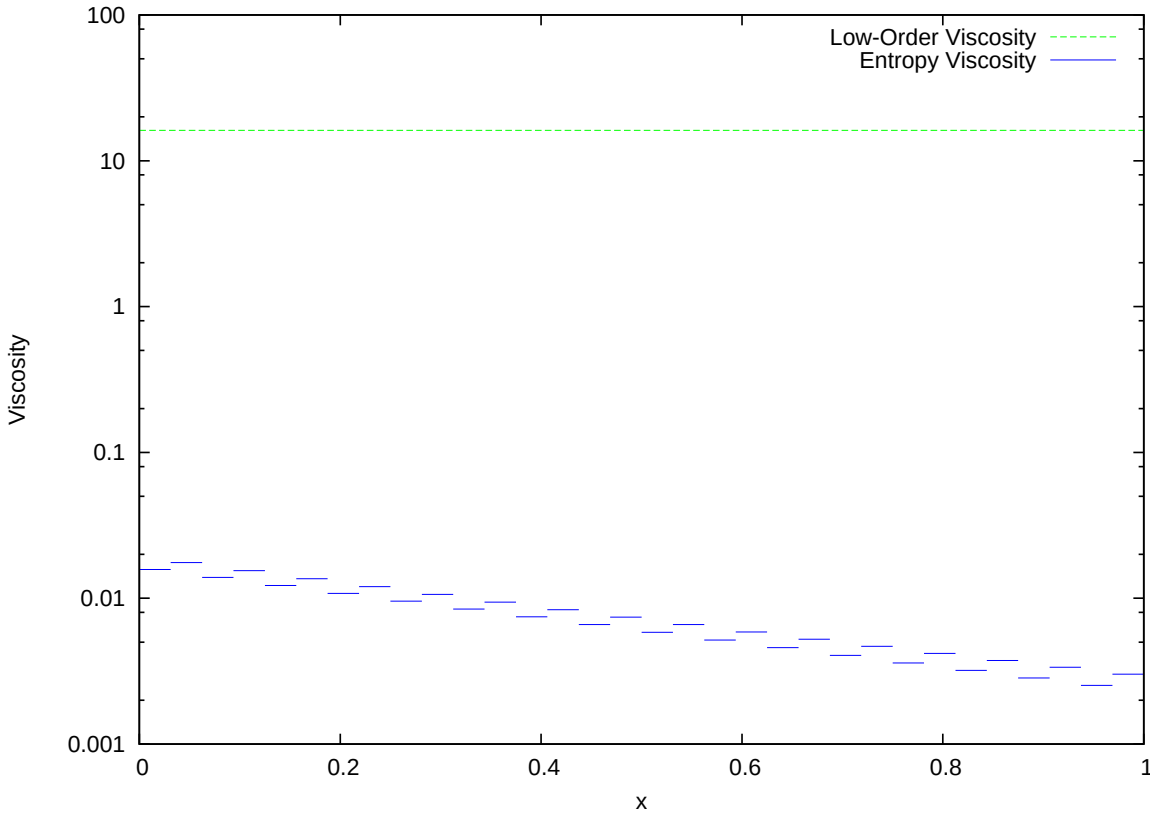


Figure 3.7: Viscosity Profiles for Convergence Test Problem 2 with 32 Cells

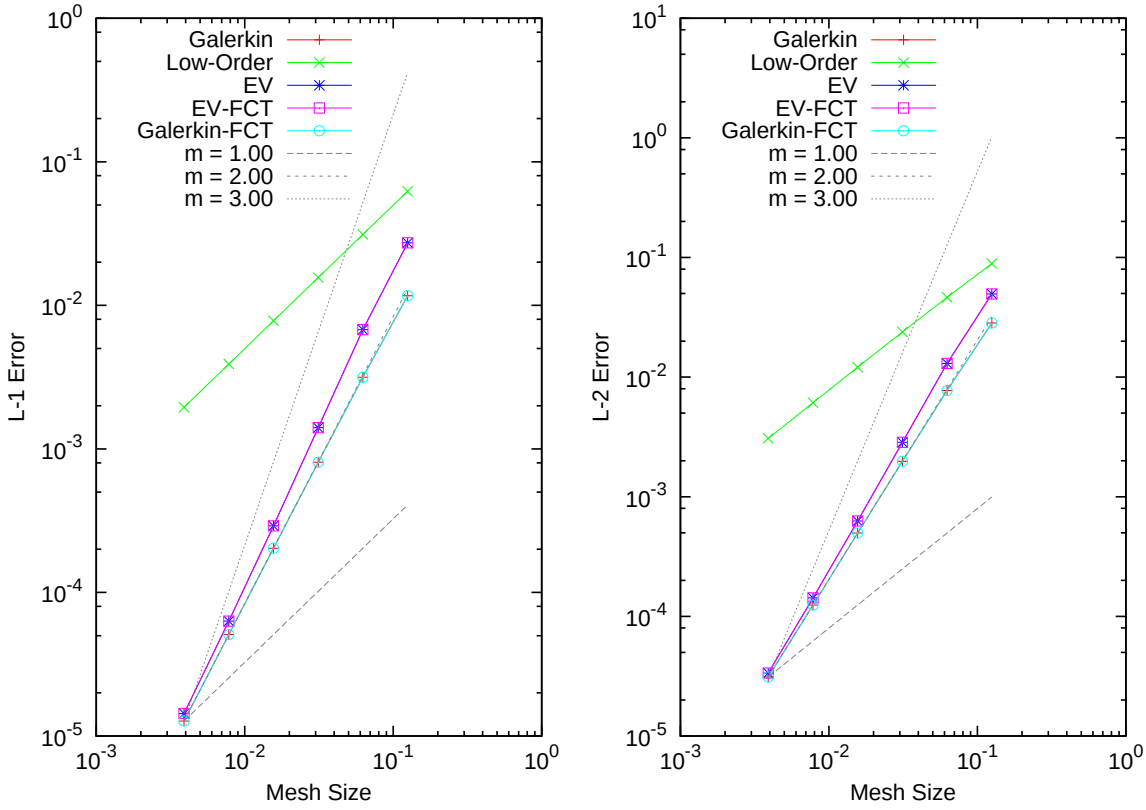


Figure 3.8: Comparison of Errors for Convergence Test Problem 2

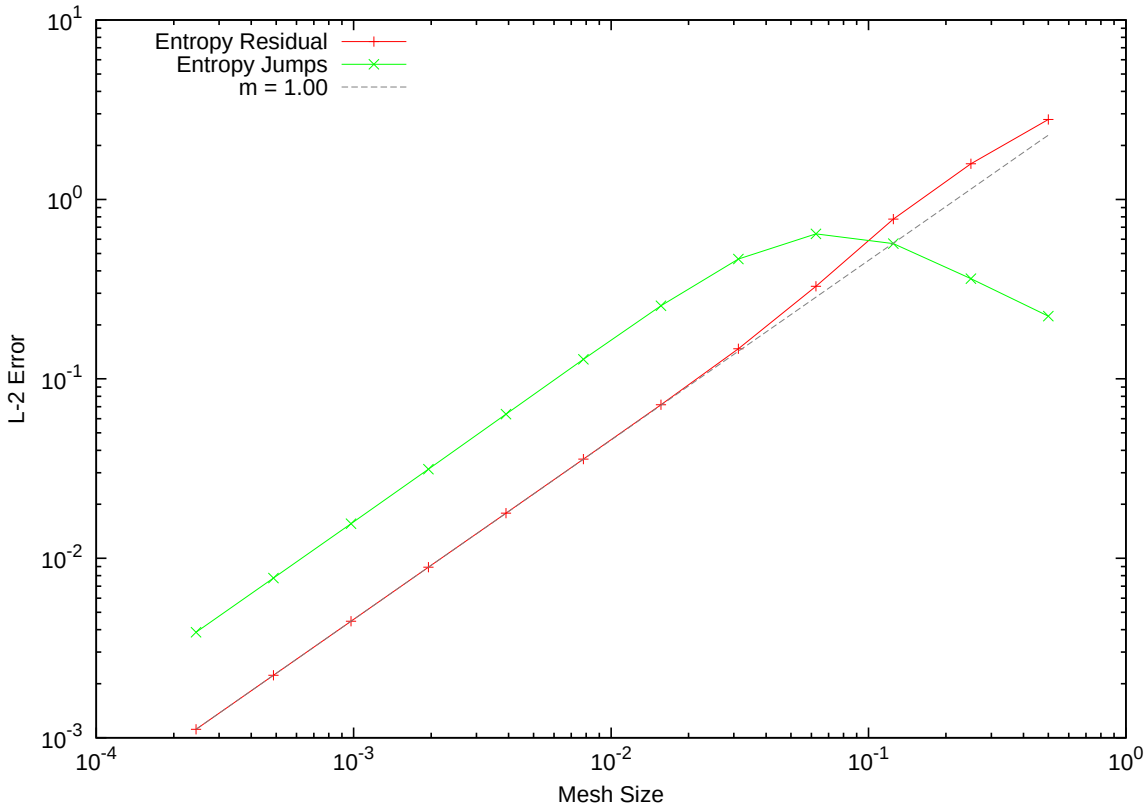


Figure 3.9: Convergence of Entropy Residual and Entropy Jumps for Convergence Test Problem 2

3.2.3 Multi-Region Test Problem

This section describes a class of test problems in which a d-dimensional unit hypercube $(0,1)^d$ domain is divided into N_r regions of uniform source q and reaction coefficient σ, Each of the test problems that fit this description are fully described by:

- the incoming flux u^{inc},
- the transport direction $\mathbf{\Omega}$,
- the transport speed v,
- the positions of the interfaces between the regions, $[r_1, \ldots, r_{N_r-1}]$ (the positions r_0 and r_{N_r} refer to the hypercube boundaries 0 and 1, respectively),
- the source strengths in each region, $q_0, \ldots, q_{N_r-1}$: $q(\mathbf{x}) = q_i$,
- the cross sections in each region, $\sigma_0, \ldots, \sigma_{N_r-1}$,

In sections that follow, exact solutions may reference the form of the exact solution described in this section instead of rewriting the exact solution for each case.

The exact solution can be described as having two components:

1. the component u_b associated with the upstream reference solution value, which may be the incident boundary value, and
2. the component u_q associated with the source.

The exact solution has the following form:

$$u(\mathbf{x}, t) = u_b + u_q \,, \tag{3.17a}$$

$$u_b = \tilde{u}_0(\mathbf{x} - v\boldsymbol{\Omega}t)e^{-\tau}\,, \quad \tau = \sum_{i=0}^{N_r-1} \sigma_i s_i\,, \tag{3.17b}$$

$$\tilde{u}_0(\mathbf{x}) = \begin{cases} u^{\text{inc}}, & \mathbf{x} \notin \mathcal{D} \\ u_0(\mathbf{x}), & \mathbf{x} \in \mathcal{D} \end{cases}\,, \tag{3.17c}$$

$$u_q = \sum_{i=0}^{N_r-1} u_{q,i} e^{-\tau_i}, \quad \tau_i = \sum_{j=i+1}^{N_r-1} \sigma_j s_j\,, \tag{3.17d}$$

$$u_{q,i} = \begin{cases} \frac{q_i}{\sigma_i}\left(1 - e^{-\sigma_i s_i}\right), & \sigma_i \neq 0 \\ q_i s_i, & \sigma_i = 0 \end{cases}\,, \tag{3.17e}$$

where s_i is the distance traveled in region i.

3.2.4 Void-to-Absorber Test Problem

This problem simulates the interface between a void region and a strong, pure absorber region, with an incoming flux boundary condition applied. Table 3.5 summarizes the problem parameters.

Table 3.6 shows the run parameters used to obtain the results in this section, Figures 3.10 and 3.11 show 2-D results for explicit Euler and SSPRK33 time discretizations, respectively, and Figure 3.13 shows 3-D results.

From Figure 3.11, one can see that the Galerkin scheme (which has no artificial dissipation) generates significant spurious oscillations perpendicular to the transport direction, even below the absorber region. The oscillations are particularly severe along the lower edge of the absorber region, where particles/photons are traveling parallel to the absorber; this edge has a sharper gradient in the solution than the left edge of the absorber region due to the lack of attenuation in this direction, which is present for the left edge. Figure 3.11, which uses explicit Euler instead of SSPRK33 does not show the Galerkin plot because the oscillations grew without bound, leading

Table 3.5: Normal Void-to-Absorber Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0,1)^d$
Initial Conditions	$u_0(\mathbf{x}) = 0$
Boundary Conditions	$u(\mathbf{x},t) = 1, \quad \mathbf{x} \in \partial\mathcal{D}^-, \quad t > 0,$ $\partial\mathcal{D}^- = \{\mathbf{x} \in \partial\mathcal{D} : \mathbf{n}(\mathbf{x}) \cdot \mathbf{\Omega} < 0\}$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(\mathbf{x}) = \begin{cases} 10, & \mathbf{x} \in (\frac{1}{2},1)^d \\ 0, & \text{otherwise} \end{cases}$
Source	$q(\mathbf{x},t) = 0$
Speed	$v = 1$
Exact Solution	$u(\mathbf{x},t) = \begin{cases} u_{\text{ss}}(\mathbf{x}), & x - t < 0 \\ 0, & \text{otherwise} \end{cases}$ $u_{\text{ss}}(\mathbf{x}) = \begin{cases} e^{-10(x-\frac{1}{2})}, & x \geq \frac{1}{2}, y \geq \frac{1}{2}, z \geq \frac{1}{2} \\ 1, & \text{otherwise} \end{cases}$

to infinite solution values. The entropy viscosity scheme is also vulnerable to spurious oscillations, although to a lesser extent than the Galerkin scheme.

Note that all numerical schemes except the Galerkin scheme involve some dissipation; this can be seen at the outgoing (right) boundary of the void region, where there is a solution gradient despite the lack of absorption. This is because the simulation was run to $t = 1$, and the transport speed is $v = 1$, so the wave front should be located at the right boundary of the domain since the domain width is equal to 1; the diffusivity at the right boundary is due to artificial diffusion along the wave front. For steady-state computations, where there is no transient and thus no wave front, one would not see this diffusivity at the right boundary.

Figure 3.12 shows the low-order and entropy viscosity profiles using SSPRK33, both on linear scales but separately scaled. One can see that the entropy viscosity is highest along the incident edge of the absorber region, and in particular, the corner of the absorber region in the center of the domain.

One can visually compare the width of the diffusive region to infer the diffusivity of each numerical scheme. For example, one can see that the low-order solution is a bit more diffusive than the high-order schemes and FCT schemes. Both the Galerkin-FCT and EV-FCT solutions show a lack of oscillations and less diffusivity than the low-order solution.

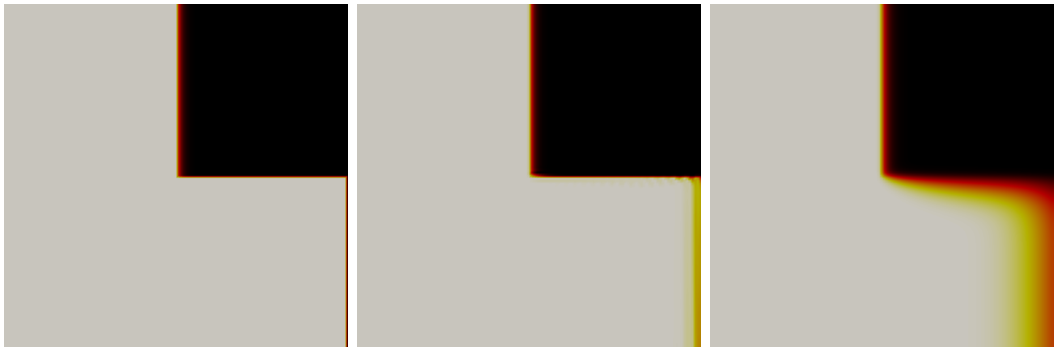
The 3-D results are included here to show a proof of principle that the FCT algorithm used is not restricted to 1-D or 2-D.

Table 3.6: Normal Void-to-Absorber Test Problem Run Parameters

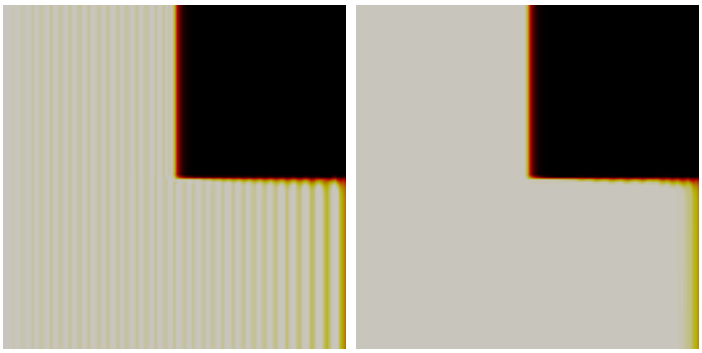
Parameter	*Value*
Number of Cells	$N_{cell} = 16384$
Time Discretization	Explicit Euler, SSPRK33, Steady-State
End Time	$t = 1$
CFL Number	$\nu = 0.5$
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	DMP

The steady-state FCT solution was found to have significant convergence difficulties, and for example, the steady-state solution with 4096 cells did not converge. The main difficulty of implicit and steady-state FCT is that in general the solution bounds, such as those given by Equation (A.9), are implicit:

$$W_i^- \leq U_i \leq W_i^+ \,, \tag{3.18a}$$



(a) Exact (b) Galerkin FCT (c) Low-Order



(d) Entropy Viscosity (e) Entropy Viscosity FCT

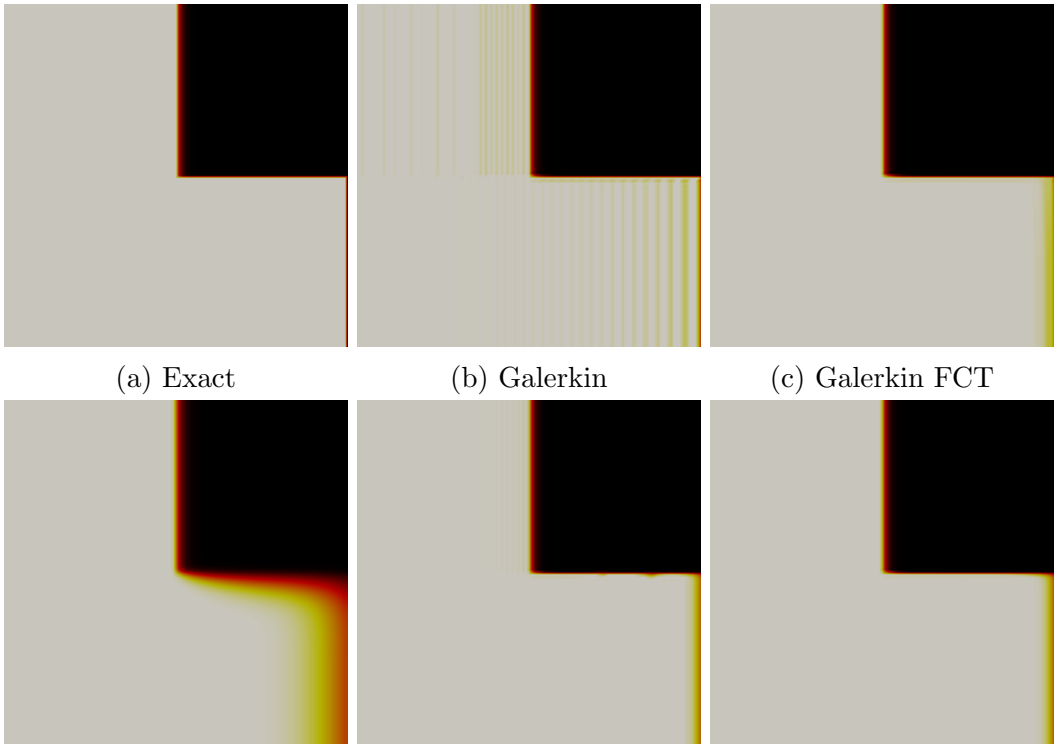
Figure 3.10: Comparison of Solutions for 2-D Normal Void-to-Absorber Test Problem Using Explicit Euler Time Discretization and Low-Order DMP Solution Bounds

$$W_i^- \equiv \begin{cases} U_{\min,i} e^{-\Delta x \sigma_{\max,i}} + \frac{q_{\min,i}}{\sigma_{\max,i}} (1 - e^{-\Delta x \sigma_{\max,i}}) \,, & \sigma_{\max,i} \neq 0 \\ U_{\min,i} + \Delta x q_{\min,i} \,, & \sigma_{\max,i} = 0 \end{cases} , \tag{3.18b}$$

$$W_i^+ \equiv \begin{cases} U_{\max,i} e^{-\Delta x \sigma_{\min,i}} + \frac{q_{\max,i}}{\sigma_{\min,i}} (1 - e^{-\Delta x \sigma_{\min,i}}) \,, & \sigma_{\min,i} \neq 0 \\ U_{\max,i} + \Delta x q_{\max,i} \,, & \sigma_{\min,i} = 0 \end{cases} . \tag{3.18c}$$

Note that for this test problem, the source q is zero; it is included in these expressions for generality.

To illustrate the issue of using implicit solution bounds, the problem was run again but with *explicitly* computed solution bounds, which use the known exact



(a) Exact (b) Galerkin (c) Galerkin FCT

(d) Low-Order (e) Entropy Viscosity (f) Entropy Viscosity FCT

Figure 3.11: Comparison of Solutions for 2-D Normal Void-to-Absorber Test Problem Using SSPRK33 Time Discretization and Low-Order DMP Solution Bounds

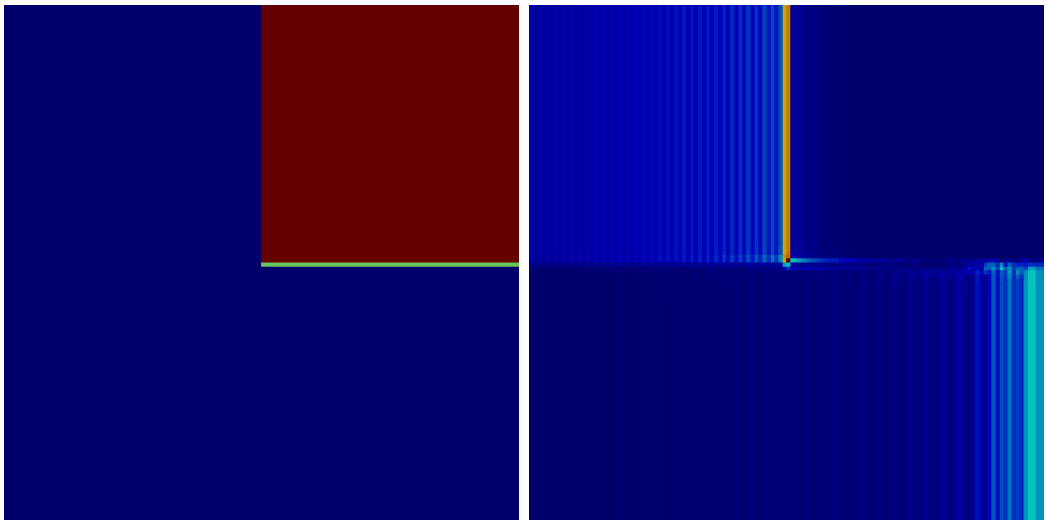
solution for this problem instead of the numerical solution values:

$$W_i^- \leq U_i \leq W_i^+ \,, \tag{3.19a}$$

$$W_i^- \equiv \begin{cases} u_{\min,i}^{\text{exact}} e^{-\Delta x \sigma_{\max,i}} + \frac{q_{\min,i}}{\sigma_{\max,i}} (1 - e^{-\Delta x \sigma_{\max,i}}) \,, & \sigma_{\max,i} \neq 0 \\ u_{\min,i}^{\text{exact}} + \Delta x q_{\min,i} \,, & \sigma_{\max,i} = 0 \end{cases} , \tag{3.19b}$$

$$W_i^+ \equiv \begin{cases} u_{\max,i}^{\text{exact}} e^{-\Delta x \sigma_{\min,i}} + \frac{q_{\max,i}}{\sigma_{\min,i}} (1 - e^{-\Delta x \sigma_{\min,i}}) \,, & \sigma_{\min,i} \neq 0 \\ u_{\max,i}^{\text{exact}} + \Delta x q_{\max,i} \,, & \sigma_{\min,i} = 0 \end{cases} . \tag{3.19c}$$

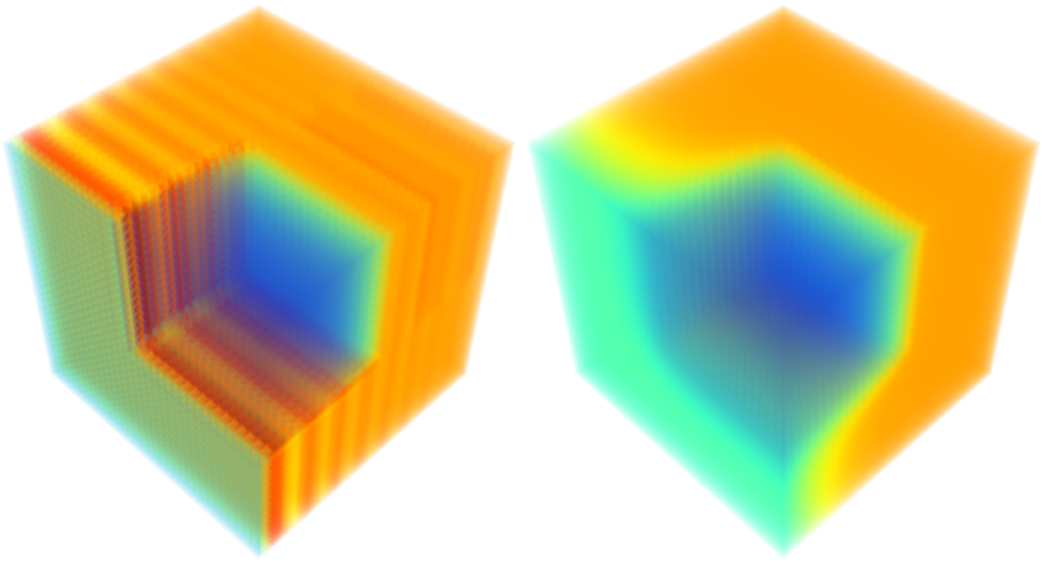
$$u_{\min,i}^{\text{exact}} \equiv \min_{j \in \mathcal{I}(S_i)} u^{\text{exact}}(\mathbf{x}_j) \,, \quad u_{\max,i}^{\text{exact}} \equiv \max_{j \in \mathcal{I}(S_i)} u^{\text{exact}}(\mathbf{x}_j) \,. \tag{3.19d}$$



(a) Low-order viscosity (b) Entropy viscosity

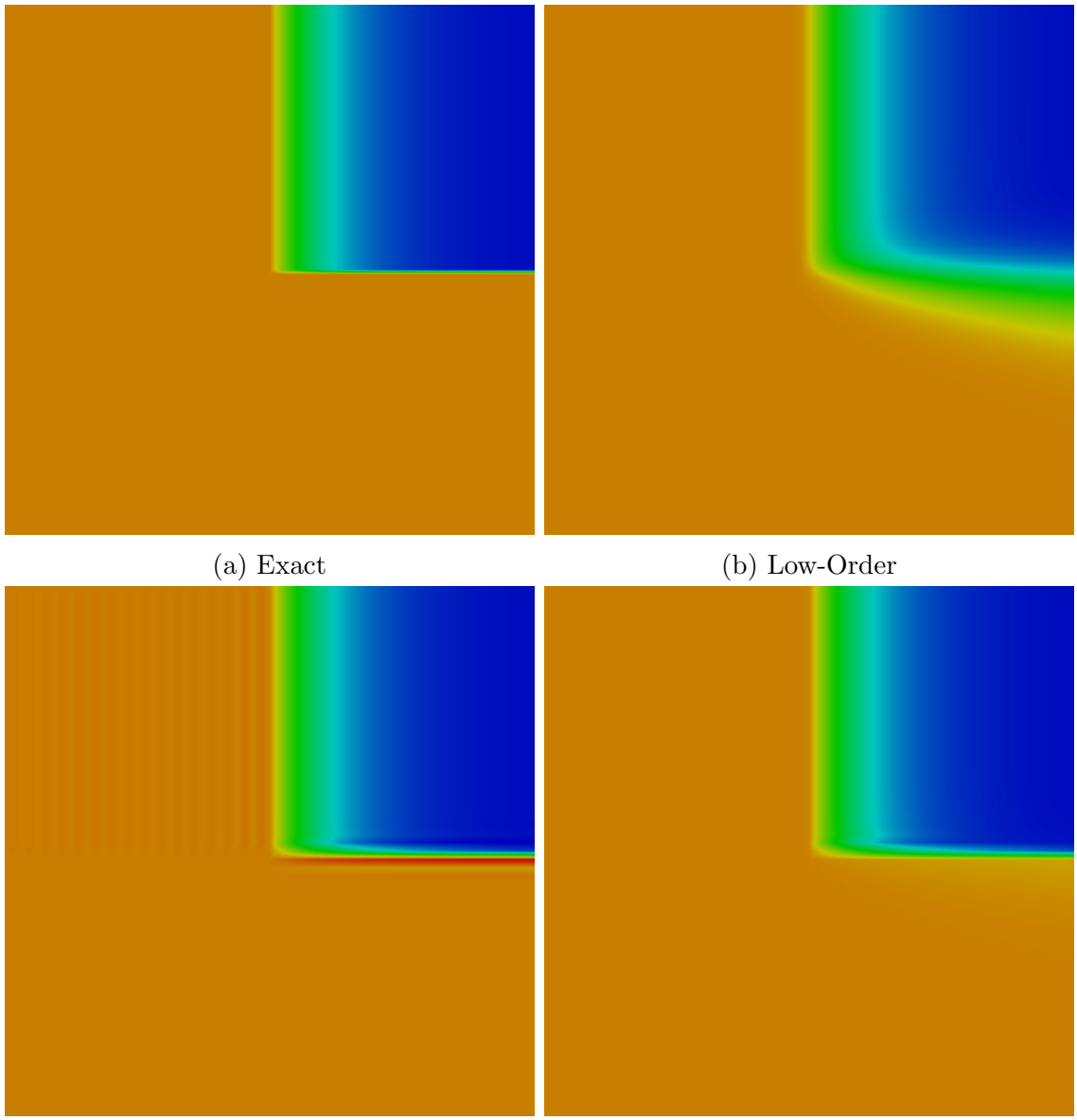
Figure 3.12: Viscosity Profiles for the 2-D Void-to-Absorber Test Problem Using SSPRK33 Time Discretization

When these bounds were used, the steady-state FCT solution did converge, with 50 iterations. Figure 3.14 compares the steady-state solutions computed with 4096 cells, with the FCT solution using the exact solution bounds given above.



(a) Galerkin (b) Galerkin with FCT

Figure 3.13: Comparison of Solutions for the 3-D Normal Void-to-Absorber Test Problem Using SSPRK33 Time Discretization and Low-Order DMP Solution Bounds



(a) Exact

(b) Low-Order

(c) Entropy Viscosity

(d) Entropy Viscosity FCT

Figure 3.14: Comparison of Steady-State Solutions for 2-D Normal Void-to-Absorber Test Problem with 4096 Cells with Exact Solution Bounds

3.2.5 Skew Void-to-Absorber Test Problem

This problem is a more general case of the test problem described in Section 3.2.4 in which the transport direction is not normal to the left boundary; it is a skew direction, for which $\Omega_i \geq 0, \forall i$, so in 2-D, particles enter through not just the left boundary, but the bottom boundary as well. The simulation is again run to $t = 1$, but since it is a skew direction, the wave front is located not along the right boundary but in an "L" shape in the strong absorber region. Table 3.7 summarizes the test parameters, where the definition of s is given below.

Table 3.7: Skew Void-to-Absorber Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0,1)^d$
Initial Conditions	$u_0(\mathbf{x}) = 0$
Boundary Conditions	$u(\mathbf{x},t) = 1, \quad \mathbf{x} \in \partial\mathcal{D}^-, \quad t > 0,$ $\partial\mathcal{D}^- = \{\mathbf{x} \in \partial\mathcal{D} : \mathbf{n}(\mathbf{x}) \cdot \mathbf{\Omega} < 0\}$
Direction	$\mathbf{\Omega} = \left[\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{3}}, \frac{1}{\sqrt{6}}\right]$
Cross Section	$\sigma(\mathbf{x}) = \begin{cases} 10, & \mathbf{x} \in (\frac{1}{2},1)^d \\ 0, & \text{otherwise} \end{cases}$
Source	$q(\mathbf{x},t) = 0$
Speed	$v = 1$
Exact Solution	$u(\mathbf{x},t) = \begin{cases} u_{\text{ss}}(\mathbf{x}), & \mathbf{x} - \mathbf{\Omega}t \notin \mathcal{D} \\ 0, & \text{otherwise} \end{cases}$ $u_{\text{ss}}(\mathbf{x}) = \begin{cases} e^{-10s}, & x \geq \frac{1}{2}, y \geq \frac{1}{2}, z \geq \frac{1}{2} \\ 1, & \text{otherwise} \end{cases}$, with s given by Equation (3.20).

The condition $\mathbf{x} - \mathbf{\Omega}t \notin \mathcal{D}$ is equivalent to the following condition:

$$\mathbf{x} - \mathbf{\Omega}t \notin \mathcal{D} \Rightarrow \exists i : x_i - \Omega_i t < 0,$$

where i denotes a coordinate direction index x, y, or z. The distance traveled in the absorber region, s, is computed by first determining which plane segment of the absorber region through which the line $\mathbf{x}-\mathbf{\Omega}t$ passes; the coordinate direction normal to this plane is denoted by i and the other two by j and k. This is determined as follows:

$$i : \frac{x_i - \frac{1}{2}}{\Omega_i} = \min_j \left(\frac{x_j - \frac{1}{2}}{\Omega_j} \right).$$

Then, s is computed as follows:

$$s = \sqrt{s_i^2 + s_j^2 + s_k^2}, \quad s_i = x_i - \frac{1}{2}, \quad s_j = \frac{\Omega_j}{\Omega_i} s_i, \quad s_k = \frac{\Omega_k}{\Omega_i} s_i. \tag{3.20}$$

The simulation was run to time $t = 1$ with a CFL of $\nu = 0.5$ and 16384 cells. Table 3.8 summarizes the run parameters for this test problem.

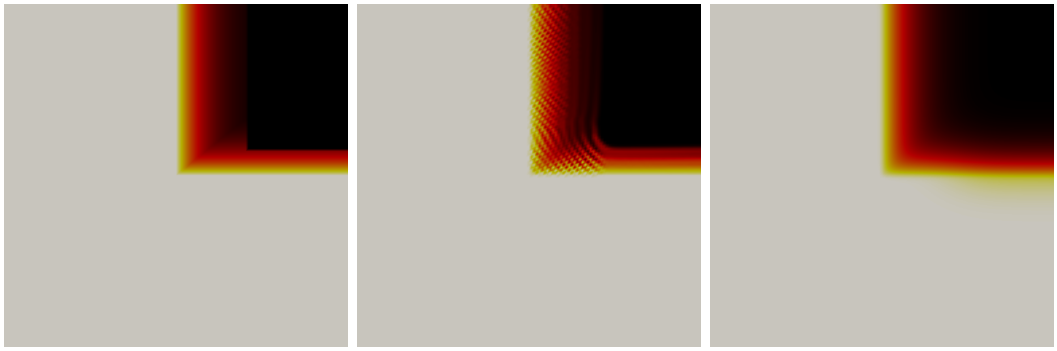
Table 3.8: Skew Void-to-Absorber Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 16384$
Time Discretization	Explicit Euler, SSPRK33
End Time	$t = 1$
CFL Number	$\nu = 0.5$
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	DMP

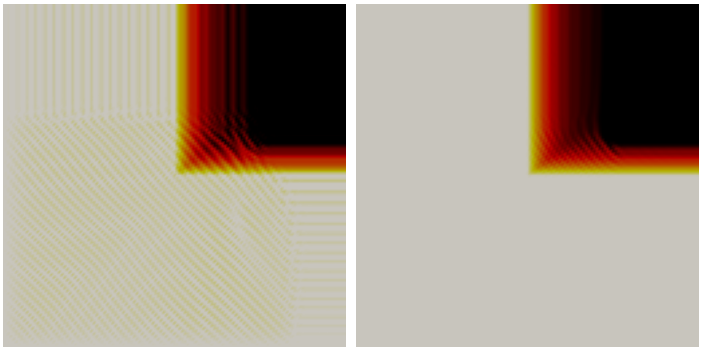
Figures 3.15 and 3.16 show 2-D results for this problem for explicit Euler and SSPRK33 time discretizations, respectively.

This test problem reveals some multidimensional effects of spurious oscillations, as one can see in the Galerkin and EV solutions. Comparing Figures 3.15 and 3.10, one can see that in the skew case, oscillations are now not just perpendicular to the x axis but also to the y axis, due to the gradients on each edge of the absorber region. One can also see some faint oscillations propagating from the corner of the absorber region. It is also interesting to note a circular effect centered at the lower left corner of the domain; this is presumed to be due to interactions between the horizontal and vertical oscillations. Also note that in this problem, the wave front is not at the right edge of the domain as it was in non-skew test problem; instead, the wave front is in the absorber region, which one can clearly see in the exact solution. The wave front is an "L" shape since with the skew, particles/photons now also enter the domain from the lower boundary of the domain. The conclusion of the results is the same as before: FCT eliminates the spurious oscillations apparent in the Galerkin and EV solutions and is sharper than the low-order solution. The advantage of using FCT over the low-order scheme is less evident in these results simply because the wave front is in the absorber region, where the solution is already very small; thus the gradient is more difficult to see.

Figure 3.17 shows the low-order and entropy viscosity profiles for SSPRK33. As in the normally-incident void-to-absorber test problem, the highest entropy viscosity occurs at the corner of the absorber region, and is otherwise large on the boundaries of the absorber region.



(a) Exact (b) Galerkin FCT (c) Low-order



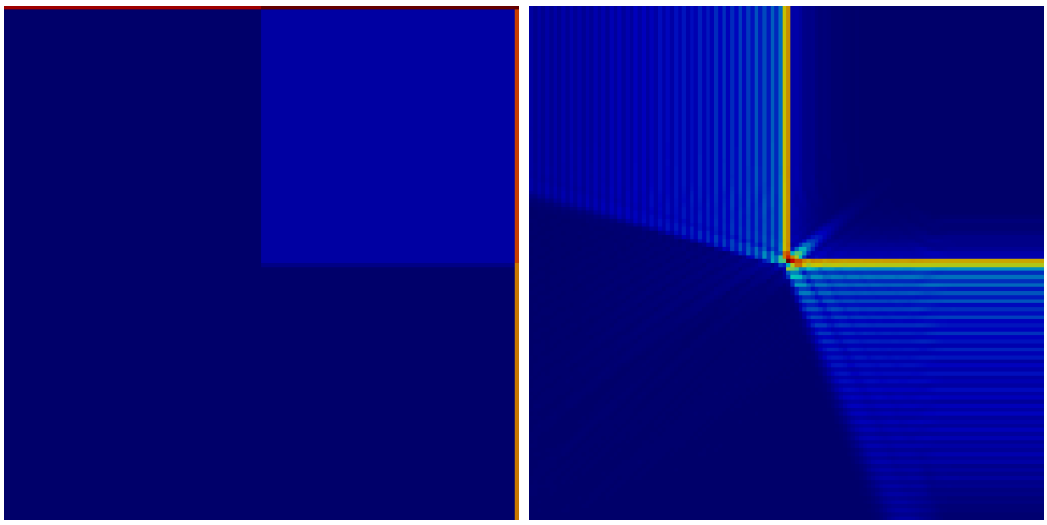
(d) Entropy Viscosity (e) Entropy Viscosity FCT

Figure 3.15: Comparison of Solutions for the Skew Void-to-Absorber Test Problem Using Explicit Euler Time Discretization and DMP Solution Bounds with 16384 Cells

(a) Exact (b) Galerkin (c) Galerkin FCT

(d) Low-order (e) Entropy Viscosity (f) Entropy Viscosity FCT

Figure 3.16: Comparison of Solutions for the Skew Void-to-Absorber Test Problem Using SSPRK33 Time Discretization and DMP Solution Bounds with 16384 Cells



(a) Low-order viscosity (b) Entropy viscosity

Figure 3.17: Viscosity Profiles for the the Skew Void-to-Absorber Test Problem Using SSPRK33 Time Discretization

3.2.6 Glance-in-Void Test Problem

This test problem simulates an incident flux at a shallow angle to the bottom edge of a 2-D void region. This test reveals the diffusivity of a transport scheme in the transverse direction; artificial diffusion will cause the "sides" of a radiation beam, not just the front, to diffuse. Table 3.9 summarizes the problem parameters.

Table 3.9: Glance-in-Void Test Problem Summary

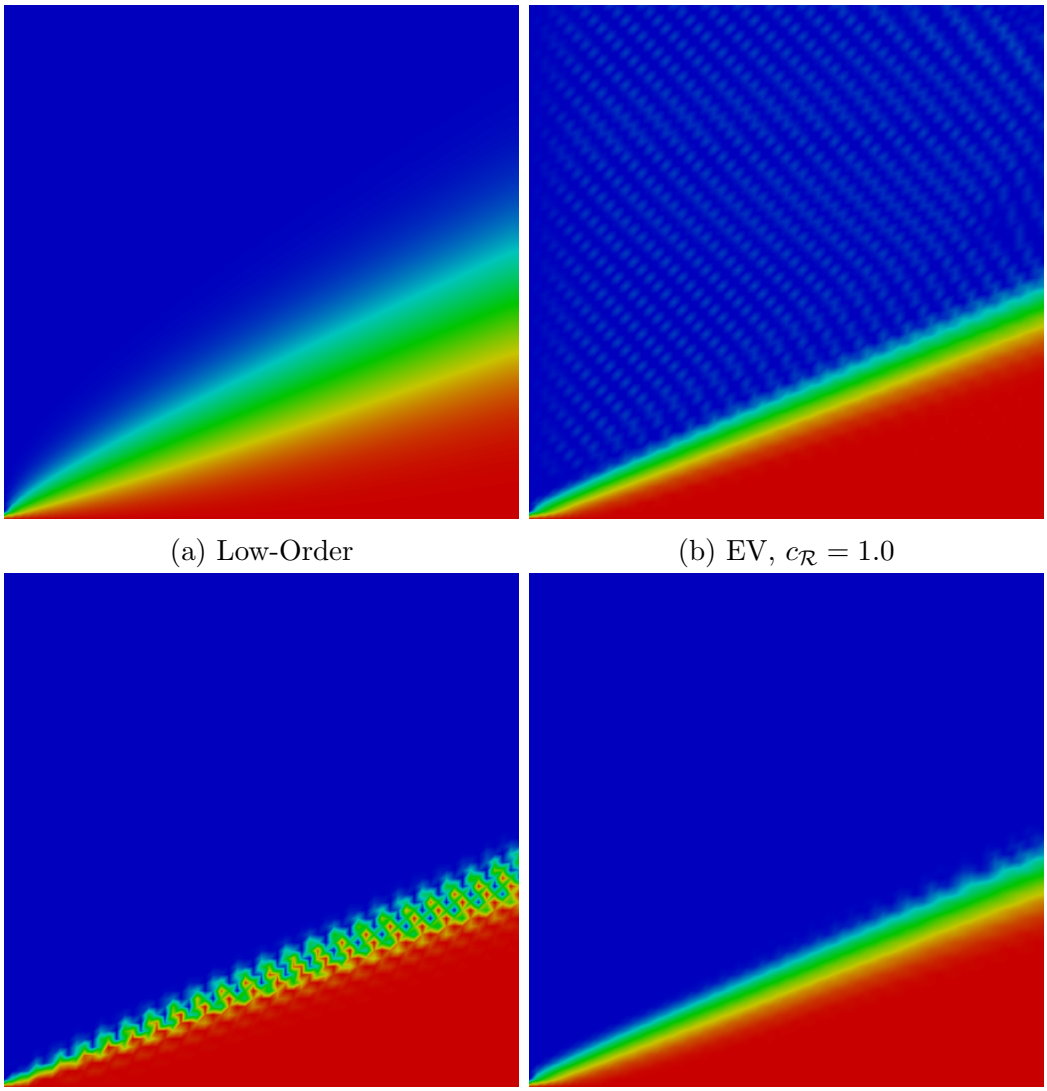
Parameter	*Value*
Domain	$\mathcal{D} = (0, 10)^2$
Initial Conditions	$u_0(\mathbf{x}) = 0$
Boundary Conditions	$u(\mathbf{x}, t) = \begin{cases} \frac{\pi}{3} & y = 0,\ t > 0 \\ 0 & x = 0,\ t > 0 \end{cases}$,
Direction	$\mathbf{\Omega} = (0.868890300722, 0.350021174582)$, normalized such that $\|\mathbf{\Omega}\| = 1$
Cross Section	$\sigma(\mathbf{x}) = 0$
Source	$q(\mathbf{x}, t) = 0$
Speed	$v = 1$

This test problem was run on a 64×64-cell mesh. The run parameters are summarized in Table 3.10. Figure 3.18 compares the solutions obtained for explicit Euler. This shows that the Galerkin-FCT solution contains many FCT artifacts, but these artifacts largely disappear when using EV-FCT instead. Figure **??** shows a comparison of EV and EV-FCT solutions for different values of the entropy residual coefficient $c_{\mathcal{R}}$; this shows that FCT artifacts gradually disappear with increasing $c_{\mathcal{R}}$. Finally, Figure **??** shows a comparison of EV-FCT and Galerkin-FCT solutions using explicit Euler vs. SSPRK33, which shows that usage of SSPRK33 essentially eliminates the FCT artifacts present in the explicit Euler solutions. However, one

can still see that the Galerkin-FCT solution has some artifacts not present in the EV-FCT solution, even when using SSPRK33.

Table 3.10: Normal Void-to-Absorber Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 4096$
Time Discretization	Explicit Euler, SSPRK33
End Time	$t = 2$
CFL Number	$\nu = 0.5$
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1, 0.5, 1.0$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1, 0.5, 1.0$
FCT Solution Bounds	DMP



(a) Low-Order

(b) EV, $c_{\mathcal{R}} = 1.0$

(c) Galerkin-FCT

(d) EV-FCT, $c_{\mathcal{R}} = 1.0$

Figure 3.18: Comparison of Solutions for the Glance-in-Void Test Problem Using Explicit Euler Time Discretization and DMP Solution Bounds with 4096 Cells

3.2.7 Obstruction Test Problem

This test problem simulates an absorber obstruction in a vacuum. An incoming flux beam at an angle of 45° to the x-axis, travels through a vacuum and collides with an absorber region. This test problem shows how the beam diffuses transversely to the travel direction and how the flux interacts at the upper left and lower right corners of the absorber block. Table 3.11 summarizes the problem parameters.

Table 3.11: Obstruction Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0,1)^2$
Initial Conditions	$u_0(\mathbf{x}) = 0$
Boundary Conditions	$u(\mathbf{x},t) = \begin{cases} 1 & y = 0 \quad t > 0 \\ 1 & x = 0 \quad t > 0 \end{cases}$
Direction	$\mathbf{\Omega} = \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)$
Cross Section	$\sigma(\mathbf{x}) = \begin{cases} 10 & \mathbf{x} \in (\frac{1}{3}, \frac{2}{3})^2 \\ 0 & \text{otherwise} \end{cases}$
Source	$q(\mathbf{x},t) = 0$
Speed	$v = 1$

This test problem was run with a number of time discretizations, including explicit (forward) Euler (FE), implicit (backward) Euler (BE), and steady-state (SS). Run parameters are summarized in Table 3.12.

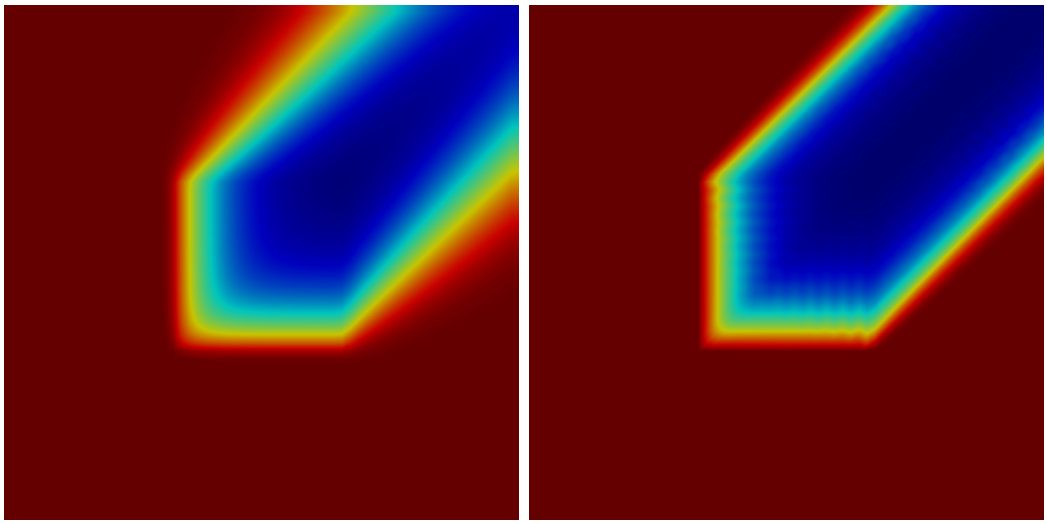
Figure 3.19 shows the explicit Euler results for the low-order scheme and the EV-FCT scheme. The low-order solution is relatively diffusive, and one can observe that the beam broadens as it travels farther away from the absorber region. In the EV-FCT solution, this broadening is not observable in the image, and the thickness of this diffusive band is smaller. However, one can still observe some slight

Table 3.12: Obstruction Test Problem Run Parameters

Parameter	*Value*
Number of Cells	FE: $N_{cell} = 4096$
	BE: $N_{cell} = 1024$
	SS: $N_{cell} = 1024$
Time Discretization	Explicit Euler, Implicit Euler, Steady-State
End Time	$t = 2$ (or to steady-state)
CFL Number	FE: $\nu = 0.1$
	BE: $\nu = 1.0$
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	DMP
FCT Initial Guess	$u^{(0)} = u^L$

"stair-stepping" behavior, where the solution does not necessarily lack monotonicity but some artificial plateaus have appeared. Figure 3.20 compares the low-order and entropy viscosity profiles for this problem, using explicit Euler. The low-order viscosity is shown on a linear scale, and the entropy viscosity is shown on a log scale. The entropy viscosity is very large around the incident edges of the absorber region, particularly the corners, and also large along the edges of the unattenuated beam.

Figure 3.21 shows the EV-FCT results obtained using implicit Euler. If comparing the EV-FCT solution to that obtained with explicit Euler, note that the number of cells for the FE solution is 4 times the number of cells used for BE and that the CFL for BE was 10 times as large as for FE. This is why one can see the beam edges begin to drift outward for the implicit Euler case - a steady-state was not reached with $t = 2$ yet when the CFL $\nu = 1$ is used, due to temporal diffusion. Figure 3.22 shows the EV-FCT results obtained using steady-state discretization.



(a) Low-order scheme (b) EV-FCT scheme

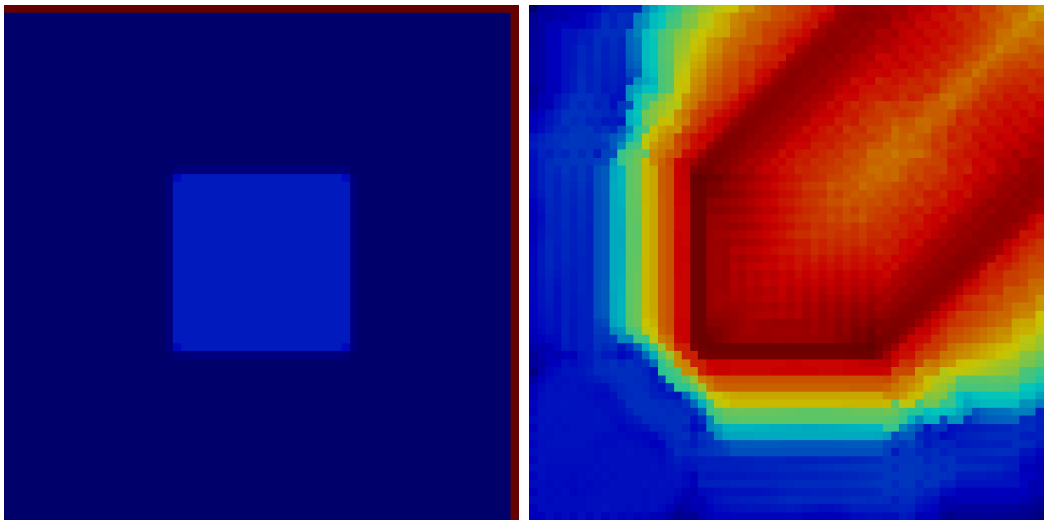
Figure 3.19: Comparison of Solutions for the Obstruction Test Problem Using Explicit Euler Time Discretization with DMP Solution Bounds

The fixed-point iteration schemes described in Sections 2.9.1 and 2.9.2 are used to converge the nonlinearities in the implicit and steady-state entropy viscosity (EV) and FCT schemes. The nonlinear solution of EV and FCT solutions is shown to have significant convergence issues. These convergence issues have most strongly been linked to the dimensionality of the problem, mesh size, and CFL number. The convergence criteria used in the nonlinear solver for the nonlinear system $\mathbf{B}(\mathbf{U})\mathbf{U} = \mathbf{s}$ (shown for each nonlinear scheme in Section 2.9.1) is

$$e^{(\ell)} = \|\mathbf{B}^{(\ell)}\mathbf{U}^{(\ell)} - \mathbf{s}^{(\ell)}\|_{\ell^2} < \epsilon = 10^{-10}\,, \tag{3.21}$$

where $\|\mathbf{a}\|_{\ell^2}$ denotes the discrete L-2 norm of $\mathbf{a}$.

Table 3.13 shows a study of the number of nonlinear iterations required for BE with 256 cells for a range of CFL numbers. In computing the EV-FCT solution for a time step, one first computes the EV solution iteratively and then computes the FCT



(a) Low-order Viscosity (b) EV-FCT scheme

Figure 3.20: Comparison of Solutions for the Obstruction Test Problem Using Explicit Euler Time Discretization with DMP Solution Bounds

solution iteratively; these two solves in the EV-FCT solution are represented by the *EV* and *FCT* columns. The sub-columns *Total* and *Avg.* correspond to the total number of iterations in the transient and the average number of iterations per time step, respectively. Entries of – denote that convergence failed during the transient for either EV or FCT, and the entry "**FAIL**" denotes which of the nonlinear iterations (EV or FCT) caused the failure. The results in the table indicate that both EV and FCT iterations increase with increasing CFL number but that the increase is more drastic with FCT. At a CFL of 20.0, the FCT scheme fails to converge. As a general rule, the solution bounds are larger with larger time steps, and there is more opportunity for antidiffusive flux limiting coefficients to vary.

Table 3.14 shows a study for backward Euler (BE) time discretization with a constant CFL of $\nu = 1$, but a varying mesh size. This shows that the average number of iterations per time step decreases with increasing mesh refinement, although this

Table 3.13: Nonlinear Iterations vs. CFL Number for the Obstruction Test Problem Using Implicit Euler Time Discretization with 256 Cells

CFL	*EV*		*FCT*	
	Total	*Avg.*	*Total*	*Avg.*
0.1	3999	8.14	3585	7.30
0.5	896	9.05	1499	15.14
1.0	501	10.02	970	19.40
5.0	157	15.70	1130	113.00
10.0	79	15.80	753	150.60
20.0	–	–	**FAIL**	

effect is not drastic.

Table 3.14: Nonlinear Iterations vs. Number of Cells for the Obstruction Test Problem Using Implicit Euler Time Discretization with CFL = 1

N_{cell}	*EV*		*FCT*	
	Total	*Avg.*	*Total*	*Avg.*
64	261	10.44	699	27.96
256	501	10.02	1053	21.06
1024	827	8.35	2040	20.61

Table 3.15 shows a study for steady-state where the mesh size is varied. These results show a different trend indicated by Table 3.14. For EV, the number of iterations required *increases* with mesh refinement. For FCT, the number of iterations required shows little or no relationship to the mesh size. For very fine mesh refinements, the EV solution does not even converge.

Table 3.15: Nonlinear Iterations vs. Number of Cells for the Steady-State Obstruction Test Problem

N_{cell}	EV	FCT
64	32	9284
256	59	440
1024	1072	3148
4096	–	–

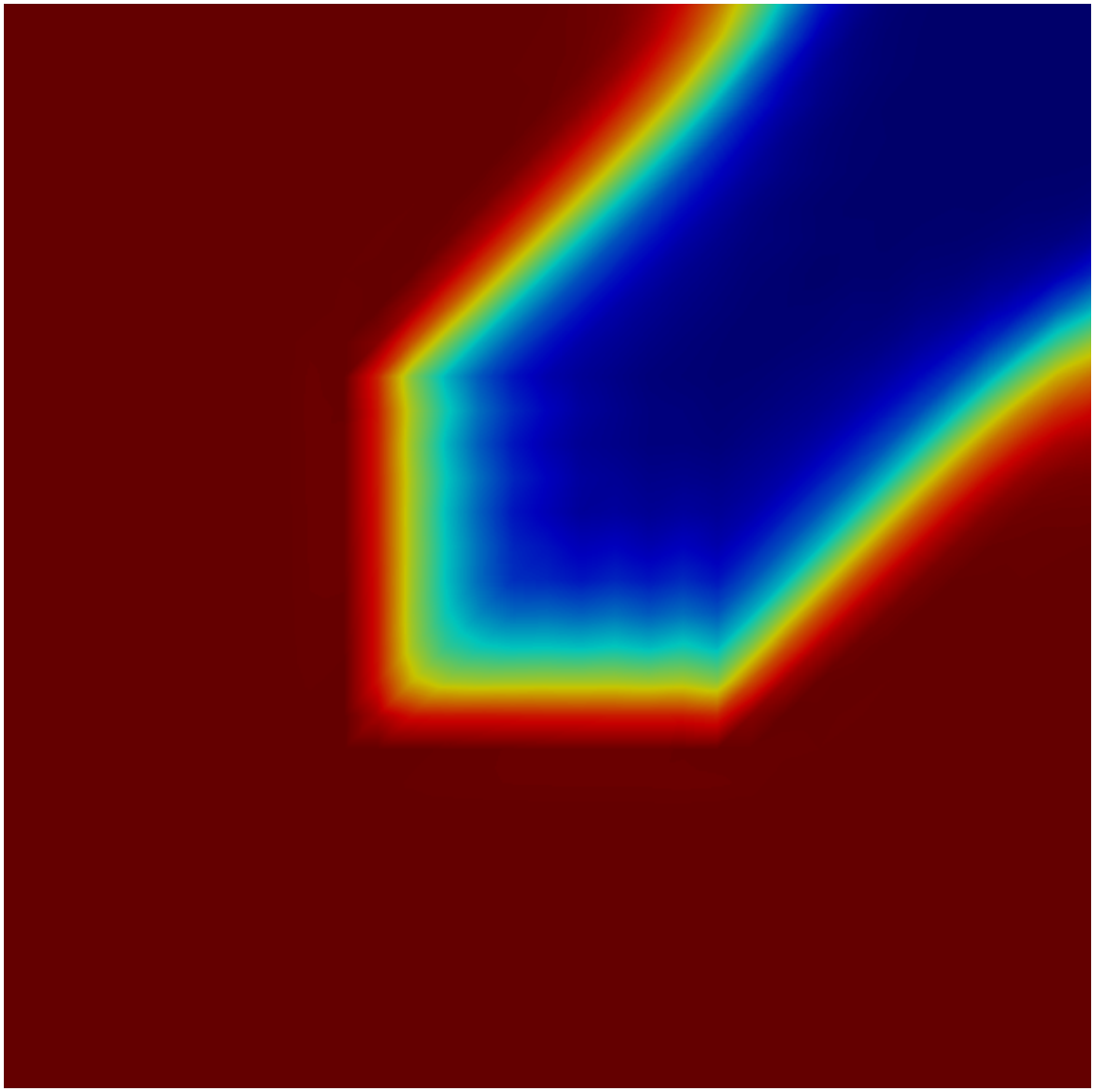


Figure 3.21: EV-FCT Solution for the Obstruction Test Problem Using Implicit Euler Time Discretization with DMP Solution Bounds

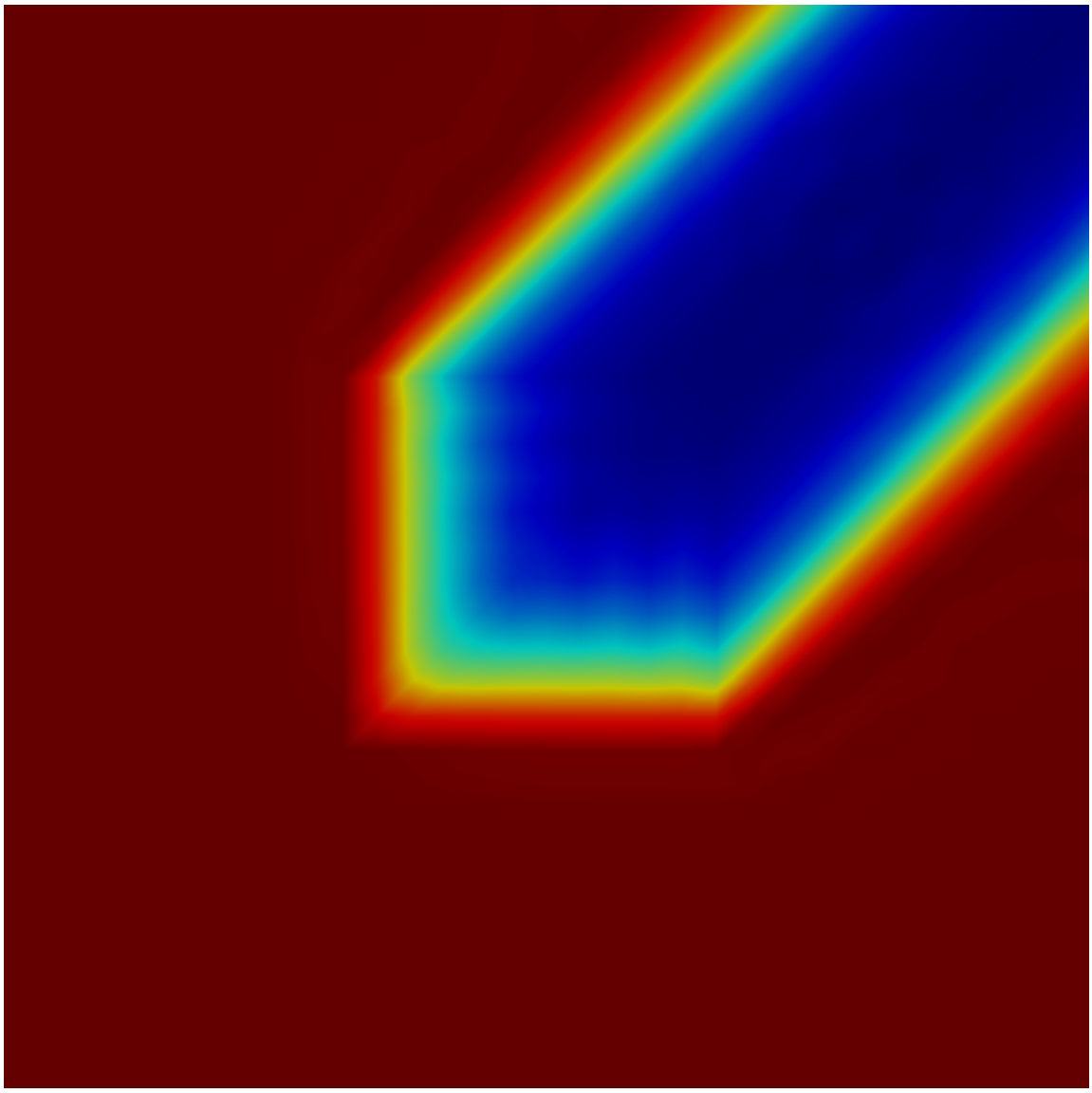


Figure 3.22: EV-FCT Solution for the Obstruction Test Problem Using Steady-State Time Discretization with DMP Solution Bounds

3.2.8 Source Void-to-Absorber Test Problem

In this section, results are presented for the source-void-to-absorber test problem. This is a 1-D test problem with zero incoming flux incident on the left boundary, a constant source in a void in the left half of the domain, and an absorber with no source in the right half of the domain. The test problem description is given by Table 3.16.

Table 3.16: Source-Void-to-Absorber Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0, 1)$
Initial Conditions	$u_0(x) = 0$
Boundary Conditions	$u(0, t) = 0\,, \quad t > 0$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(x) = \begin{cases} 0, & x < \frac{1}{2} \\ 10, & \text{otherwise} \end{cases}$
Source	$q(\mathbf{x}, t) = \begin{cases} 1, & x < \frac{1}{2} \\ 0, & \text{otherwise} \end{cases}$
Speed	$v = 1$
Exact Solution	$u(x, t) = \begin{cases} u_{\text{ss}}(x), & x - t < 0 \\ 0, & \text{otherwise} \end{cases}$
	$u_{\text{ss}}(x) = \begin{cases} e^{-10(x-\frac{1}{2})}, & x \geq \frac{1}{2} \\ 1, & \text{otherwise} \end{cases}$

Figure 3.23 shows the results for this problem using SSPRK time discretization, a CFL of 0.5, and 32 cells. Entropy residual and jump coefficients $c_{\mathcal{R}}$ and $c_{\mathcal{J}}$ are both 1. Table 3.17 summarizes the run parameters to generate the results in this section. Figure 3.24 shows results for a finer mesh (256 cells) that illustrates the shortcomings of Galerkin-FCT vs. EV-FCT: Galerkin-FCT does not necessarily converge to the entropy solution.

Table 3.17: Source-Void-to-Absorber Test Problem Run Parameters

Parameter	*Value*
Number of Cells	(varies by run)
Time Discretization	SSPRK33, Steady-State, Implicit Euler
End Time	$t = 1$
CFL Number	$\nu = 0.5$
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$, unless otherwise specified
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 1$
FCT Solution Bounds	DMP
FCT Initial Guess	$u^{(0)} = u^L$

The steady-state results for this test problem revealed some significant FCT issues regarding the antidiffusion from Dirichlet nodes. When Dirichlet boundary conditions are strongly imposed, solution bounds do not apply, and it becomes unclear how to limit antidiffusion fluxes from these nodes. Consider symmetric limiters, i.e., those such that $L_{i,j} = L_{j,i}$, such as Zalesak's limiter, for which

$$L_{i,j} = \begin{cases} \min(L_i^+, L_j^-) & P_{i,j} > 0 \\ \min(L_i^-, L_j^+) & P_{i,j} < 0 \end{cases} . \tag{3.22}$$

Suppose i corresponds to a degree of freedom for which Dirichlet boundary conditions are strongly imposed. The uncertainty is the correct way to decide L_i^+ and L_i^- since there are no valid bounds from which to compute these values. Figures 3.25 and 3.26 show the solutions obtained using strongly imposed Dirichlet boundary conditions with these values set to $L_i^+ = L_i^- = 1$ and $L_i^+ = L_i^- = 0$, respectively. When $L_i^+ = L_i^- = 1$, the correction flux from the Dirichlet DoF i, which is positive, has only

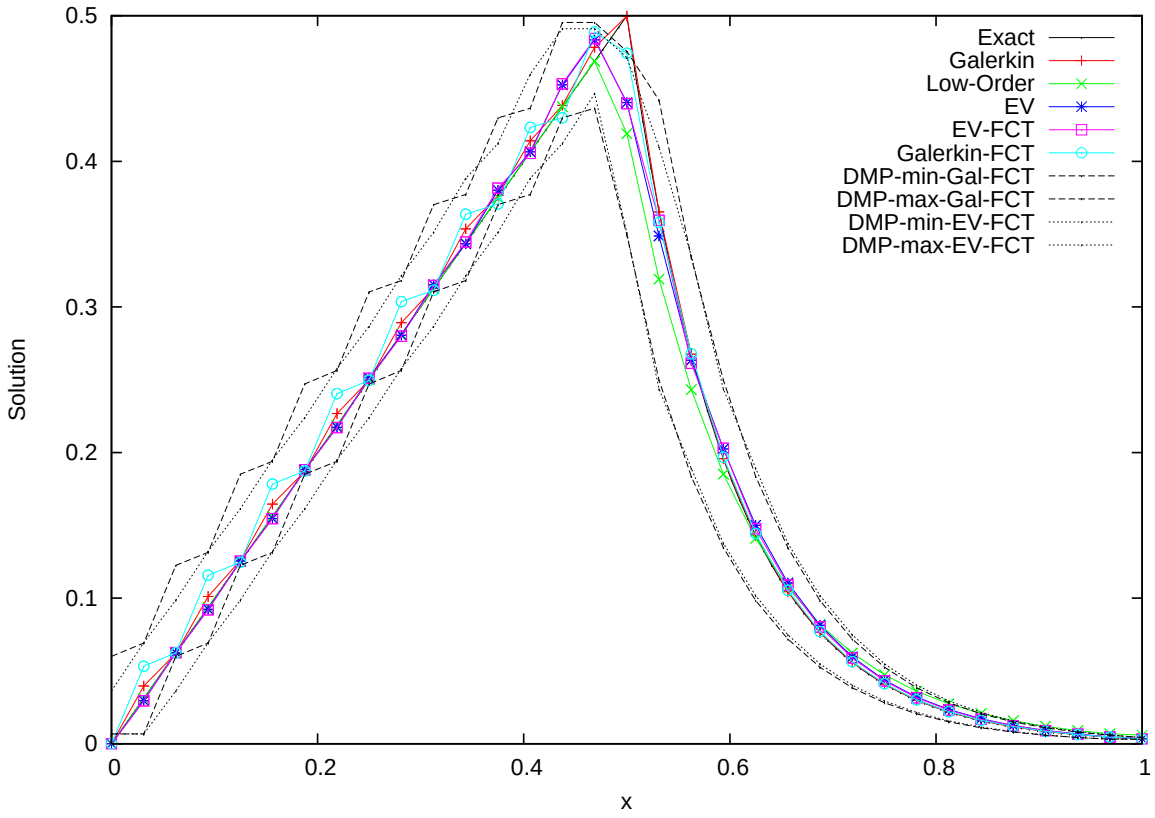


Figure 3.23: Comparison of Solutions for the Source-Void-to-Absorber Problem Using SSPRK33 with DMP Solution Bounds with 32 Cells

the upper bound for j to consider. The upper bound for j, which is inflated above the analytical solution due to the source, does not restrict this antidiffusion flux, and thus it is accepted fully to the unphysical value. Due to the implicitness of the solution bounds, the lower solution bound for j is computed from this unphysical value and excludes the possibility of antidiffusion back to the analytical solution. This process continues with all of the other degrees of freedom. When instead, $L_i^+ = L_i^- = 0$, the solution does not lie above the analytical solution in the source region, but significant peak clipping appears at the interface between the source and absorber regions. It should be noted that there are combinations of limiting coefficient values, each in the range $(0, 1)$, that produce a more accurate solution to this problem (without the

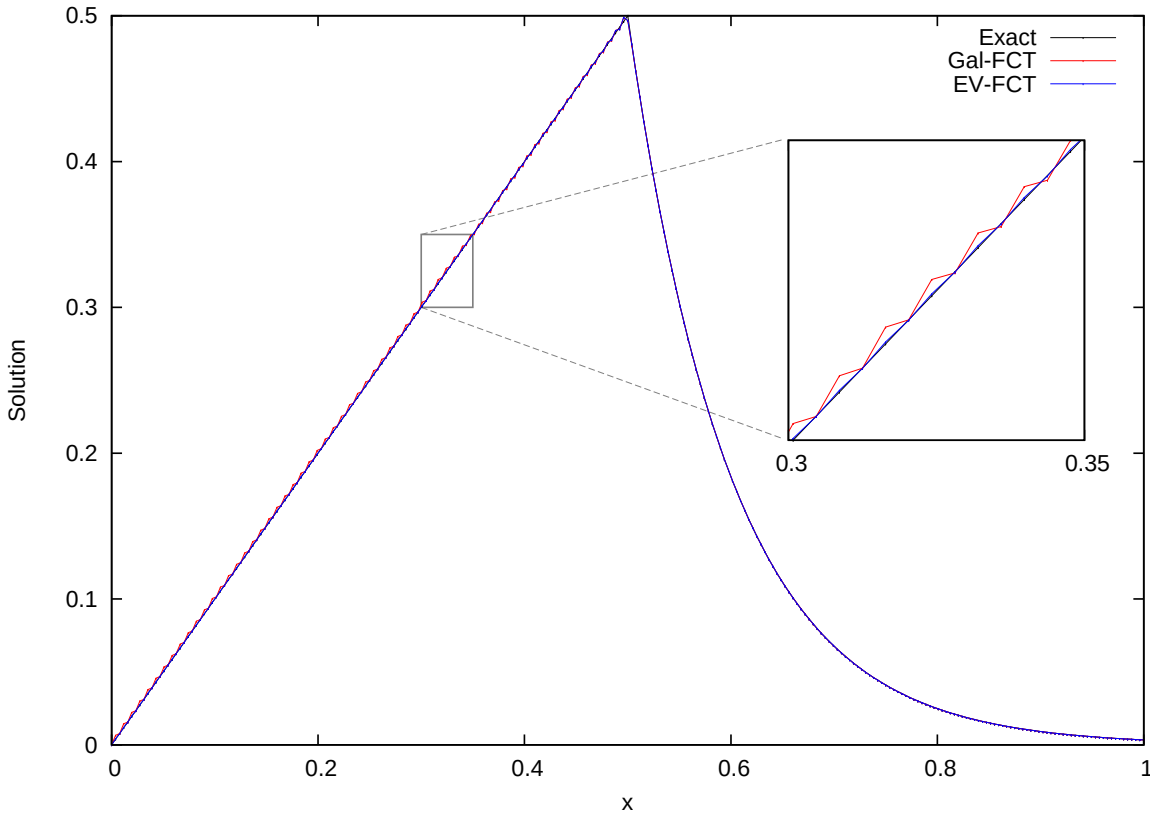


Figure 3.24: Comparison of Solutions for the Source-Void-to-Absorber Problem Using SSPRK33 with DMP Solution Bounds with 256 Cells

peak clipping and resulting inaccuracy in the absorber region); the problem is that Zalesak's limiter (and in general, any practical limiter) is not optimal in the sense that it maximizes the magnitude of antidiffusive flux. One could in principle solve an optimization problem to select limiting coefficients that maximize the antidiffusive flux, but this is very expensive and thus not recommended for general use.

For weakly imposed Dirichlet boundary conditions, solution bounds still apply, so limiting coefficients may be computed without special consideration. However, one must now consider the possibility of inaccurate boundary values. Figures 3.27 shows the steady-state solutions obtained using weakly imposed Dirichlet boundary conditions. In this case, the antidiffusion flux from the boundary gets limited (but

not fully) due to the lower solution bound of the Dirichlet node. Because some antidiffusion was accepted here, the peak reaches a higher value than with the $L_i^+ = L_i^- = 0$ case. Finally, Figure 3.28 shows the steady-state solution obtained with weakly imposed Dirichlet boundary conditions and a boundary penalty (see Section 2.2.1.1). The FCT solution looks very similar to the case without any penalty, but the effect on the low-order and entropy viscosity solutions is clear.

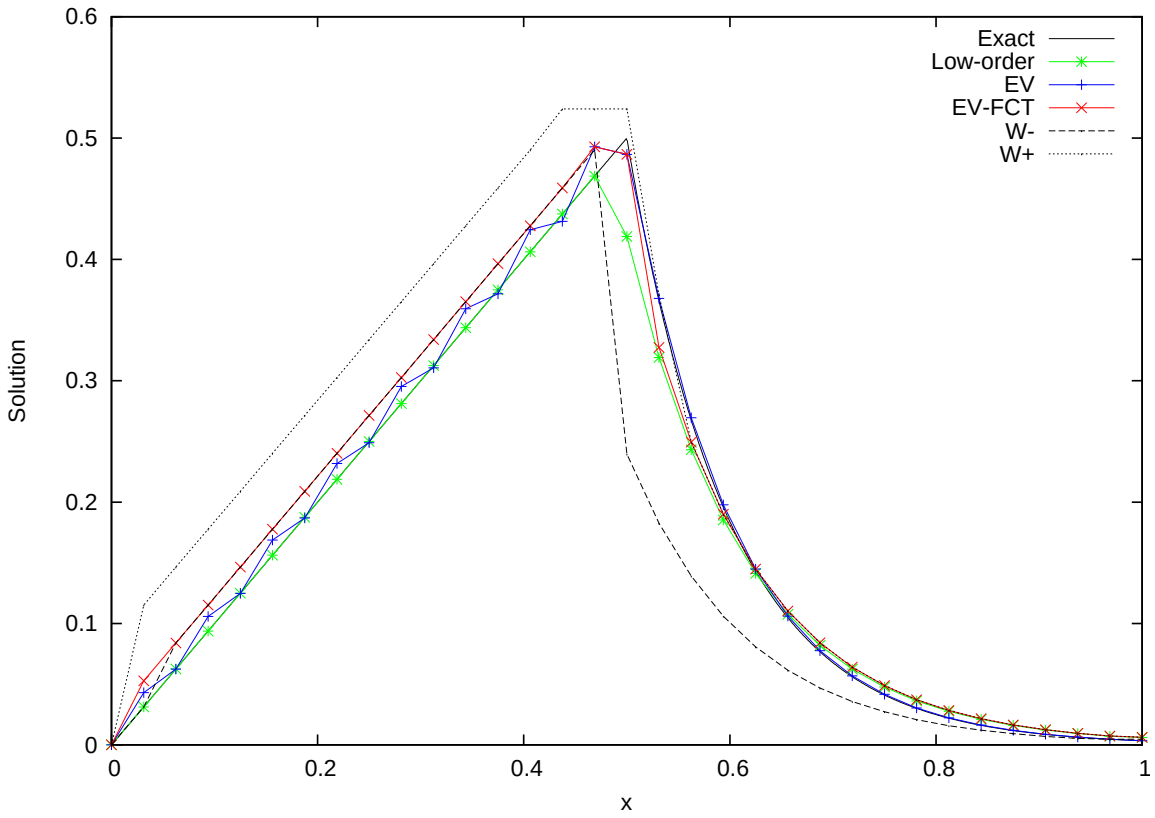


Figure 3.25: Steady-State Solutions for the Source-Void-to-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 1$ and DMP Solution Bounds

Table 3.18 shows the results of a study of the number of EV and FCT iterations for BE time discretization, required in a transient with a constant CFL of 1 and

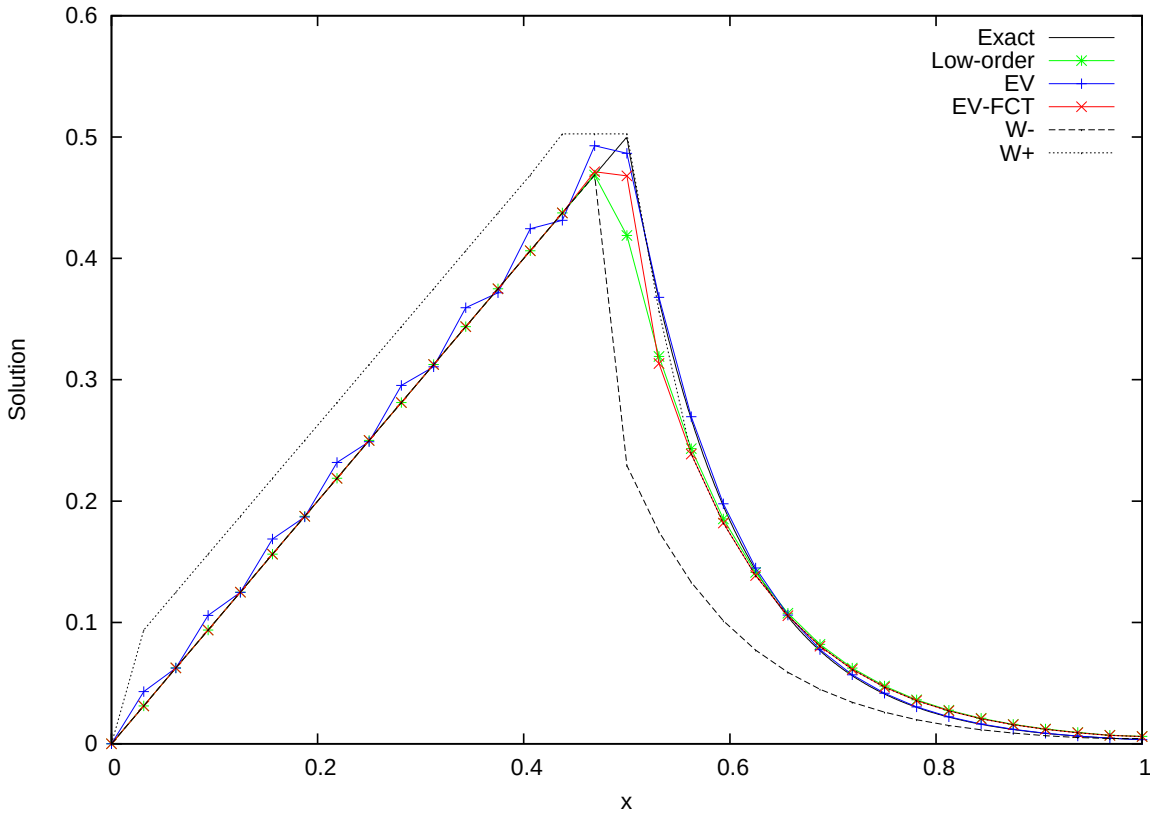


Figure 3.26: Steady-State Solutions for the Source-Void-to-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 0$ and DMP Solution Bounds

varying mesh sizes. The results in the table show a decrease in the number of EV iterations per time step, and a relatively constant number of FCT iterations per time step.

Table 3.19 shows the results of a study of nonlinear iterations vs. CFL number for implicit Euler time discretization and 128 cells. The general trend shows that entropy viscosity iterations per time step gradually increase with increasing CFL, while FCT iterations per time step increases much more quickly. Even more problematic is that the EV-FCT solution error jumps very quickly from CFLs $\nu = 5$ to $\nu = 10$.

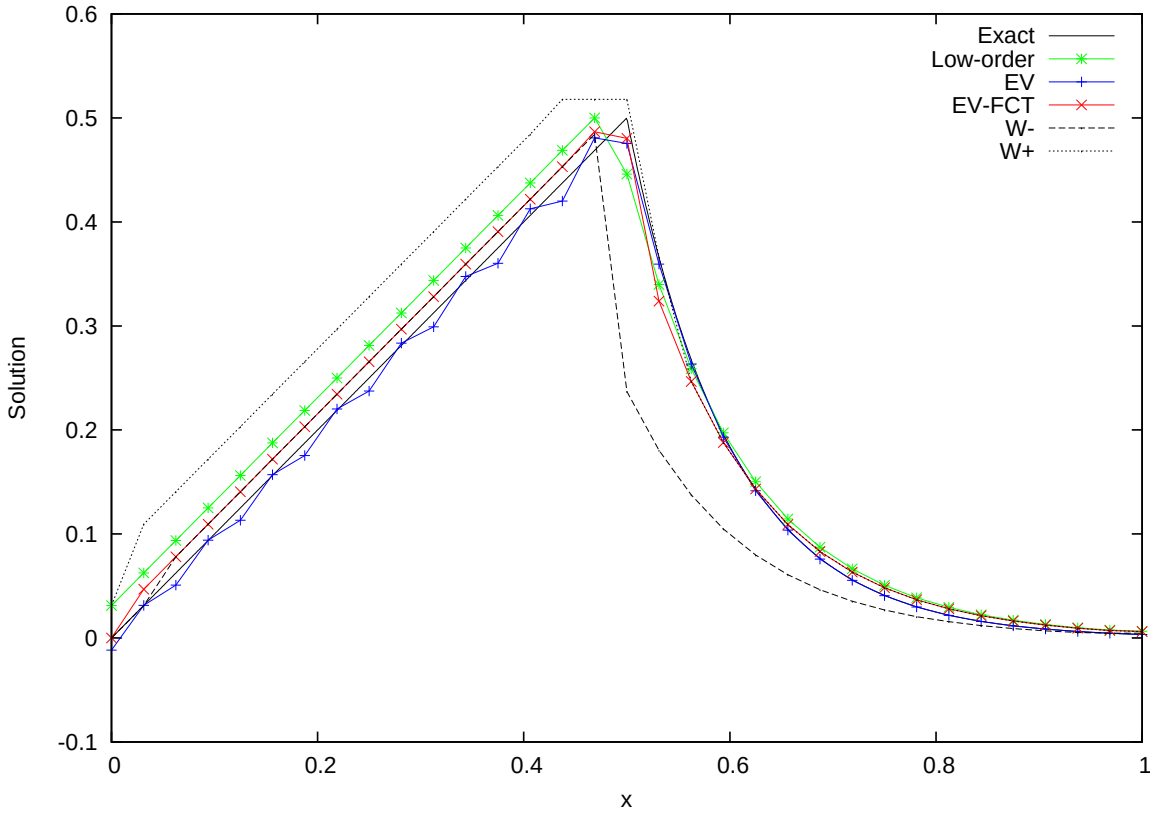


Figure 3.27: Steady-State Solutions for the Source-Void-to-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions and DMP Solution Bounds

Table 3.18: Nonlinear Iterations vs. Number of Cells for the Source-Void-to-Absorber Test Problem Using Implicit Euler Time Discretization with CFL = 1

N_{cell}	*EV*		*FCT*	
	Total	*Avg.*	*Total*	*Avg.*
8	661	24.48	244	9.04
16	807	19.21	655	15.60
32	844	11.25	1194	15.92
64	1204	8.72	2024	14.67
128	1752	6.59	3675	13.82
256	2713	5.20	6673	12.78
512	4284	4.14	12098	11.69

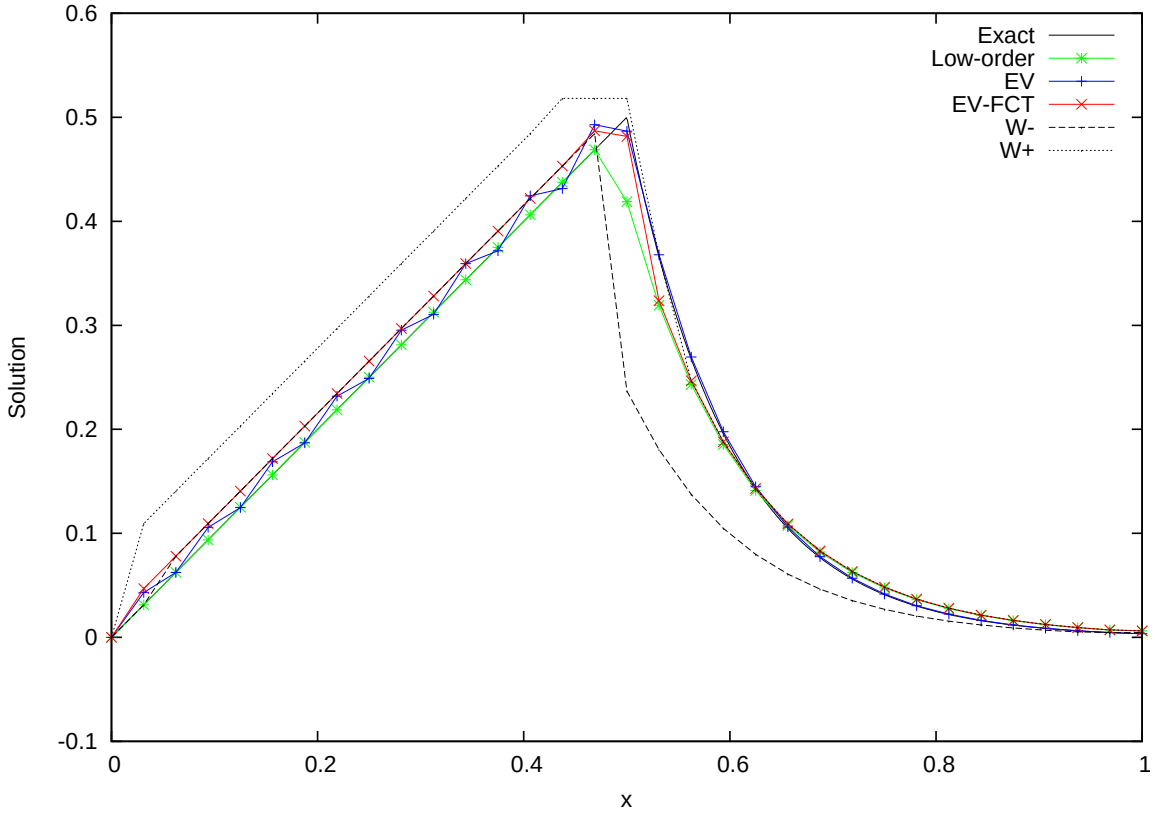


Figure 3.28: Steady-State Solutions for the Source-Void-to-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions and Boundary Penalty and DMP Solution Bounds

Table 3.19: Nonlinear Iterations vs. CFL Number for the Source-Void-to-Absorber Test Problem Using Implicit Euler Time Discretization with 128 Cells

		EV		*FCT*		
CFL	N_{step}	*Total*	*Avg.*	*Total*	*Avg.*	L^2 *err.*
0.1	2661	15006	5.64	14036	5.27	3.013×10^{-3}
0.5	533	3445	6.46	5000	9.38	3.033×10^{-3}
1.0	266	1752	6.59	3675	13.82	3.023×10^{-3}
5.0	54	471	8.72	12208	226.07	2.979×10^{-3}
10.0	27	232	8.59	6126	226.89	3.325×10^{-3}
20.0	14	133	9.50	3713	265.21	3.727×10^{-3}
50.0	6	62	10.33	2077	346.17	7.191×10^{-3}

3.2.9 Source-in-Absorber Test Problem

This test problem is a 1-D problem with a source in the left half of an absorber and a zero incident flux imposed on the left boundary. The solution in the left half of the domain shows the solution reaching its saturation value $\frac{q}{\sigma}$, and the solution in the right half is an exponential decay. Table 3.20 summarizes the test parameters.

Table 3.20: Source-in-Absorber Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0, 1)$
Initial Conditions	$u_0(x) = 0$
Boundary Conditions	$u(0, t) = u_{inc} = 0$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(\mathbf{x}) = \begin{cases} \sigma_0, & x \in [x_0, x_1] \\ \sigma_1, & x \in (x_1, x_2] \end{cases}$, $\begin{bmatrix} \sigma_0 \\ \sigma_1 \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \end{bmatrix}$ $\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix}$
Source	$q(\mathbf{x}, t) = \begin{cases} q_0, & x \in [x_0, x_1] \\ q_1, & x \in (x_1, x_2] \end{cases}$, $\begin{bmatrix} q_0 \\ q_1 \end{bmatrix} = \begin{bmatrix} 10 \\ 0 \end{bmatrix}$
Speed	$v = 1$
Exact Solution	(Equation (3.17))

Table 3.21 shows the run parameters used for the results in this section. Figures 3.29, 3.30, 3.31, and 3.32 compares the solutions computed for each scheme using different boundary conditions, respectively the following: strongly imposing with $L_i^- = L_i^+ = 1$ for Dirichlet nodes, strongly imposing with $L_i^- = L_i^+ = 0$ for Dirichlet nodes, weakly imposing, and weakly imposing with boundary penalty ($\alpha_i = 1000$). One can see clearly from a comparison of the FCT solutions with strongly imposed BC that cancellation of the antidiffusive fluxes from the boundary has a significant

impact on the solution; the approach to the first saturation value roughly follows the low-order solution because the first antidiffusive flux was canceled. Allowing nonzero antidiffusive fluxes from the Dirichlet node shows superior results in this case - the FCT solution follows the high-order solution on the first approach to saturation, which is within the FCT solution bounds. These results illustrate a typical dilemma with this issue; accepting some antidiffusion from a Dirichlet node does not satisfy conservation statements, but often in practice, cancellation of antidiffusive fluxes from these nodes leads to an inaccurate solution in the vicinity. Weak imposition of BC here gives very inaccurate boundary values; however, with a penalty applied, the solutions are much more accurate, and in addition, the solution is conservative, unlike when strongly imposing the BC with $L_i^- = L_i^+ = 1$ for Dirichlet nodes.

Table 3.21: Source-in-Absorber Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 32$
Time Discretization	Steady-State
Boundary Conditions	(varies by run)
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	Analytic

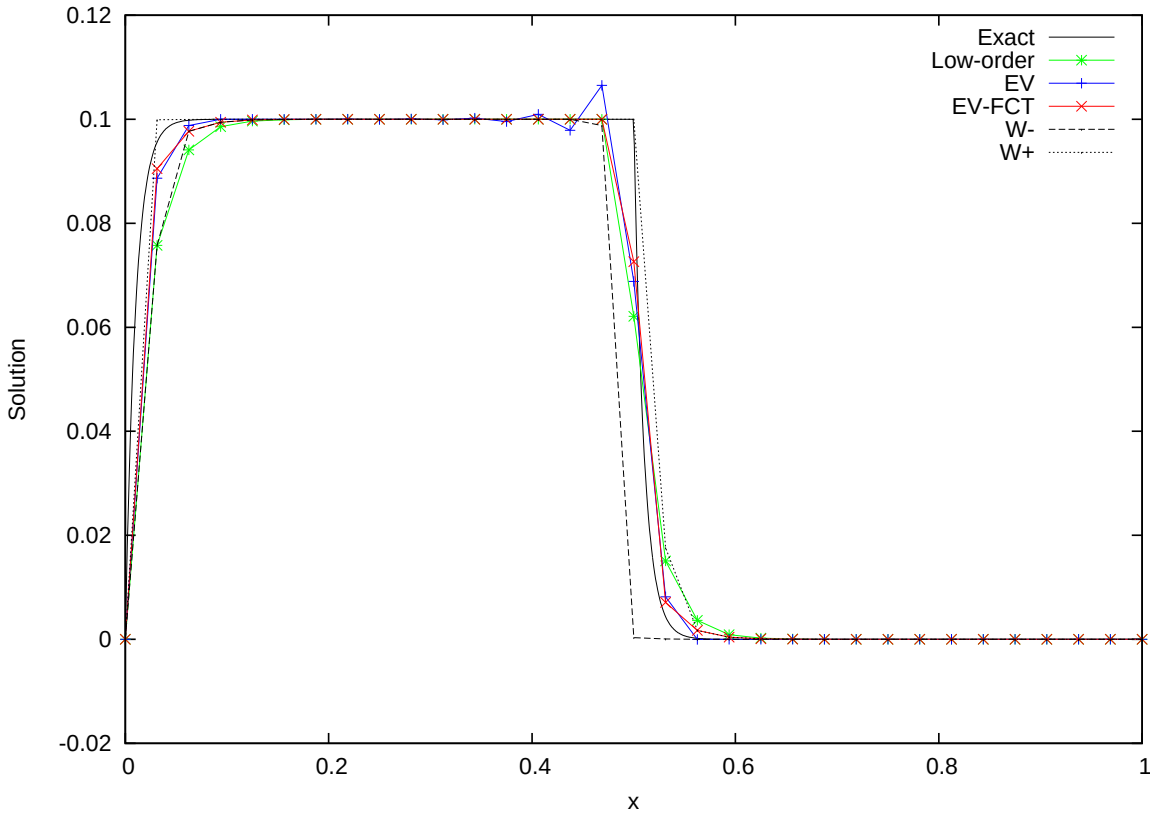


Figure 3.29: Steady-State Solutions for the Source-in-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 1$

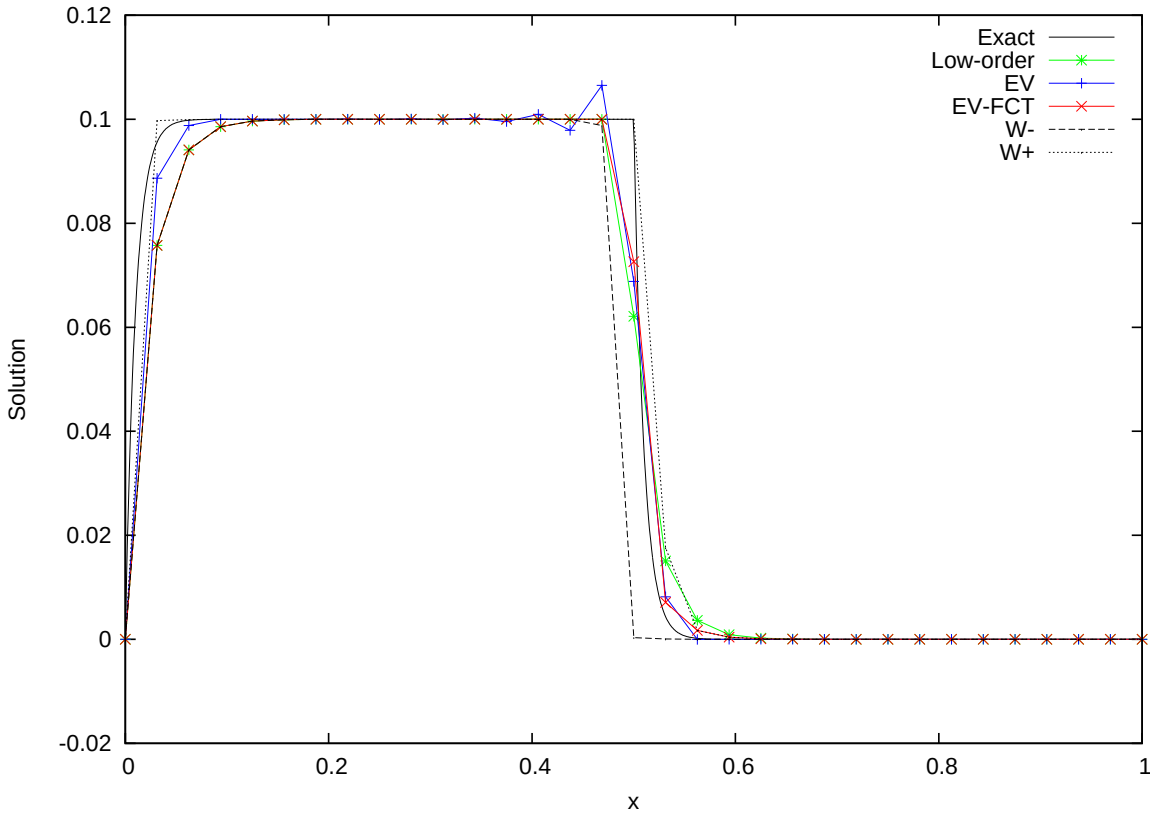


Figure 3.30: Steady-State Solutions for the Source-in-Absorber Problem with Strongly Imposed Dirichlet Boundary Conditions with $L_i^- = L_i^+ = 0$

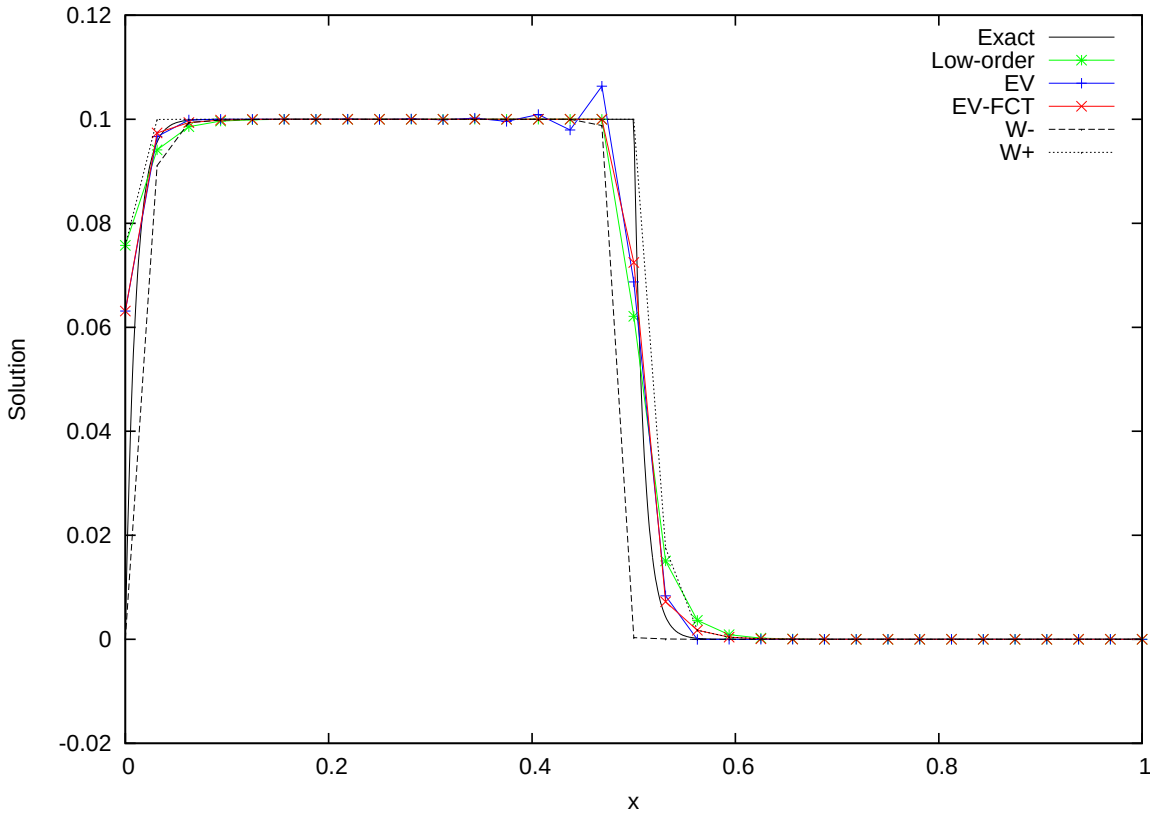


Figure 3.31: Steady-State Solutions for the Source-in-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions

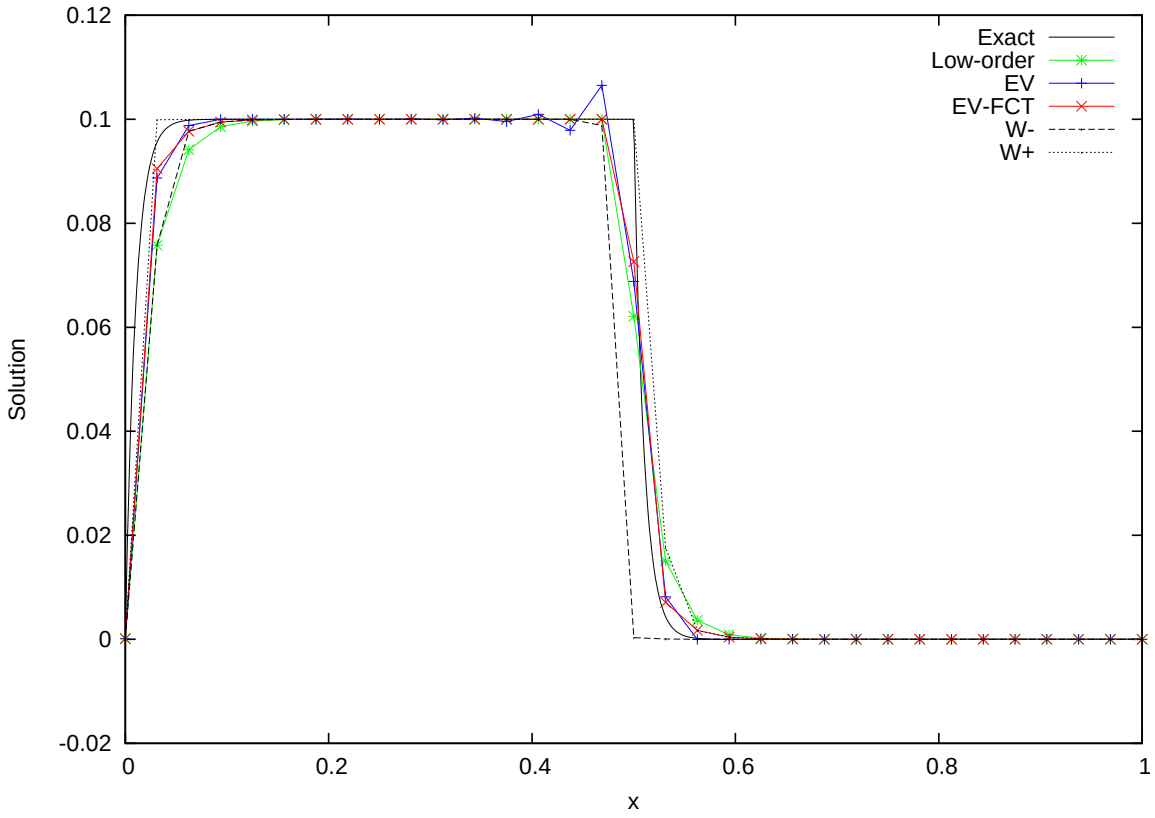


Figure 3.32: Steady-State Solutions for the Source-in-Absorber Problem with Weakly Imposed Dirichlet Boundary Conditions and Boundary Penalty

3.2.10 Interface Test Problem

This test problem is a 1-D problem with two regions with nonzero, uniform reaction coefficients and sources. The left region has a saturation value of $\frac{q}{\sigma} = 1$, and the right region has a saturation value of $\frac{q}{\sigma} = 0.5$. Table 3.22 summarizes the test parameters.

Table 3.22: Interface Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0, 1)$
Initial Conditions	$u_0(x) = 0$
Boundary Conditions	$u(0, t) = u_{inc} = 0$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(\mathbf{x}) = \begin{cases} \sigma_0, & x \in [x_0, x_1] \\ \sigma_1, & x \in (x_1, x_2] \end{cases}$, $\begin{bmatrix} \sigma_0 \\ \sigma_1 \end{bmatrix} = \begin{bmatrix} 10 \\ 40 \end{bmatrix}$ $\begin{bmatrix} x_0 \\ x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.5 \\ 1 \end{bmatrix}$
Source	$q(\mathbf{x}, t) = \begin{cases} q_0, & x \in [x_0, x_1] \\ q_1, & x \in (x_1, x_2] \end{cases}$, $\begin{bmatrix} q_0 \\ q_1 \end{bmatrix} = \begin{bmatrix} 10 \\ 20 \end{bmatrix}$
Speed	$v = 1$
Exact Solution	(Equation (3.17))

Two studies are performed with this test problem. The first considers a modification of the analytic solution bounds, and the second considers upwind solution bounds and the multi-pass limiting approach described in Section 2.7.4.2. Both studies are performed in steady-state with 32 cells. Table 3.23 summarizes the run parameters used for these studies.

The steady-state analytic solution given by Equation (A.9) applied to this test

Table 3.23: Interface Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 32$
Time Discretization	Steady-State
Boundary Conditions	Weak Dirichlet with Boundary Penalty
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	Analytic, Modified, Upwind

problem are

$$W_i^- \leq U_i \leq W_i^+ \,, \tag{3.23a}$$

$$W_i^- \equiv U_{\min,i} e^{-\Delta x \sigma_{\max,i}} + \frac{q_{\min,i}}{\sigma_{\max,i}} (1 - e^{-\Delta x \sigma_{\max,i}}) \,, \tag{3.23b}$$

$$W_i^+ \equiv U_{\max,i} e^{-\Delta x \sigma_{\min,i}} + \frac{q_{\max,i}}{\sigma_{\min,i}} (1 - e^{-\Delta x \sigma_{\min,i}}) \,. \tag{3.23c}$$

The first study considers instead, the tighter solution bounds

$$W_i^- \equiv U_{\min,i} e^{-\Delta x \sigma_{\max,i}} + \left(\frac{q}{\sigma}\right)_{\min,i} (1 - e^{-\Delta x \sigma_{\max,i}}) \,, \tag{3.24a}$$

$$W_i^+ \equiv U_{\max,i} e^{-\Delta x \sigma_{\min,i}} + \left(\frac{q}{\sigma}\right)_{\max,i} (1 - e^{-\Delta x \sigma_{\min,i}}) \,. \tag{3.24b}$$

These tighter solution bounds have not been proven analytically; however, they may still be a decent approximation in practice. Figure 3.33 shows a comparison of these solution bounds and the FCT solutions computed using the bounds. For the original bounds, there are sharp peaks at the interface of the two regions where the value $q_{\min,i}/\sigma_{\max,i}$ becomes overly conservative for the lower bound and similarly for the upper bound. Results show little or no difference between the FCT solutions even

though the modified bounds lack the peaks at the interface. In this case at least, it turns out that either there is no antidiffusion left to go into the interface node, or the solution bounds of neighboring nodes prohibit any additional antidiffusion into the interface node. Note that only the reason why the solution bounds differ at nodes other than the interface node is because of the difference in the iterative paths taken to obtain the FCT solutions.

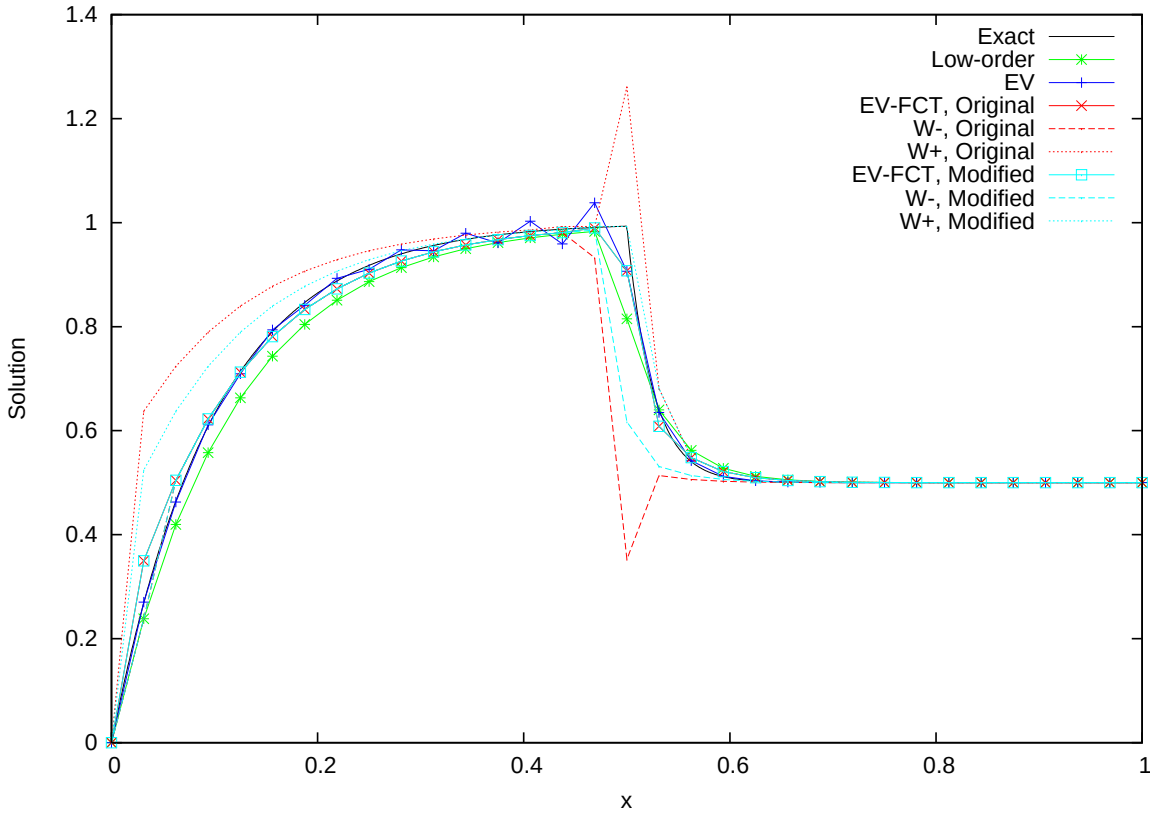


Figure 3.33: Comparison of Steady-State FCT Solutions for the Interface Test Problem Obtained Using Original and Modified Analytic Solution Bounds with 32 Cells

For the second study, the upwind solution bounds given by Equation (A.5) are

considered, where here the node indices are assumed to be ordered left to right:

$$W_i^- \equiv U_{i-1} e^{-\Delta x \sigma_{\max,(i-1,i)}} + \frac{q_{\min,(i-1,i)}}{\sigma_{\max,(i-1,i)}} (1 - e^{-\Delta x \sigma_{\max,(i-1,i)}}) \,, \tag{3.25a}$$

$$W_i^+ \equiv U_{i-1} e^{-\Delta x \sigma_{\min,(i-1,i)}} + \frac{q_{\max,(i-1,i)}}{\sigma_{\min,(i-1,i)}} (1 - e^{-\Delta x \sigma_{\min,(i-1,i)}}) \,, \tag{3.25b}$$

$$\sigma_{\min,(i-1,i)} \equiv \min_{\mathbf{x} \in (\mathbf{x}_{i-1}, \mathbf{x}_i)} \sigma(\mathbf{x}) \,, \quad \sigma_{\max,(i-1,i)} \equiv \max_{\mathbf{x} \in (\mathbf{x}_{i-1}, \mathbf{x}_i)} \sigma(\mathbf{x}) \,, \tag{3.25c}$$

$$q_{\min,(i-1,i)} \equiv \min_{\mathbf{x} \in (\mathbf{x}_{i-1}, \mathbf{x}_i)} q(\mathbf{x}) \,, \quad q_{\max,(i-1,i)} \equiv \max_{\mathbf{x} \in (\mathbf{x}_{i-1}, \mathbf{x}_i)} q(\mathbf{x}) \,, \tag{3.25d}$$

The second study also considers the multi-pass limitation process described in Section 2.7.4.2. Figure 3.34 shows the results for the non-upwind solution bounds given by Equation (3.23), with and without multi-pass limiting, and Figure 3.35 shows the results for the upwind solution bounds, with and without multi-pass limiting. The upwind solution bounds are shown to be much tighter, but the FCT solution with single-pass limiting is still relatively inaccurate due to the implicit nature of the solution bounds in steady-state FCT. Multi-pass limiting gives far superior results in both cases. Note that the multi-pass limiting stopping criteria was that the total antidiffusion accepted in a pass is less than 1 percent of the original available antidiffusion. Given that multi-pass limiting produces superior results, one just needs to consider the additional cost of this procedure against the benefit. It may be that a more practical multi-pass limiting procedure can be achieved by using a less strict tolerance. Table 3.24 gives the number of FCT iterations required for each considered set of solution bounds for this test problem, as well as the number of multi-pass limiting passes. The use of upwind bounds significantly decreases the number of required iterations; this is because the tighter solution bounds give smaller ranges for the antidiffusive fluxes. The use of multi-pass limiting appears to sometimes increase

the number of iterations and sometimes decrease the number of iterations.

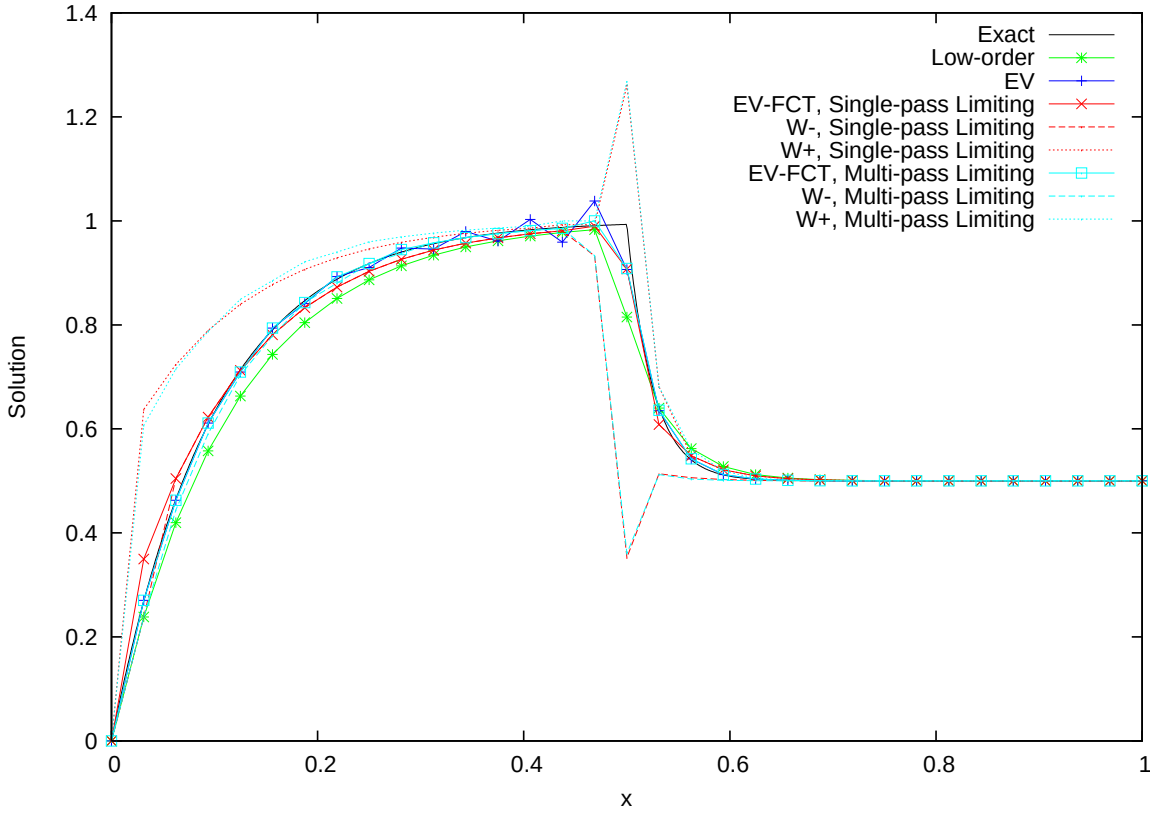


Figure 3.34: Comparison of Steady-State FCT Solutions for the Interface Test Problem Obtained with Non-Upwind Analytic Solution Bounds with Single-Pass and Multi-Pass Limiting

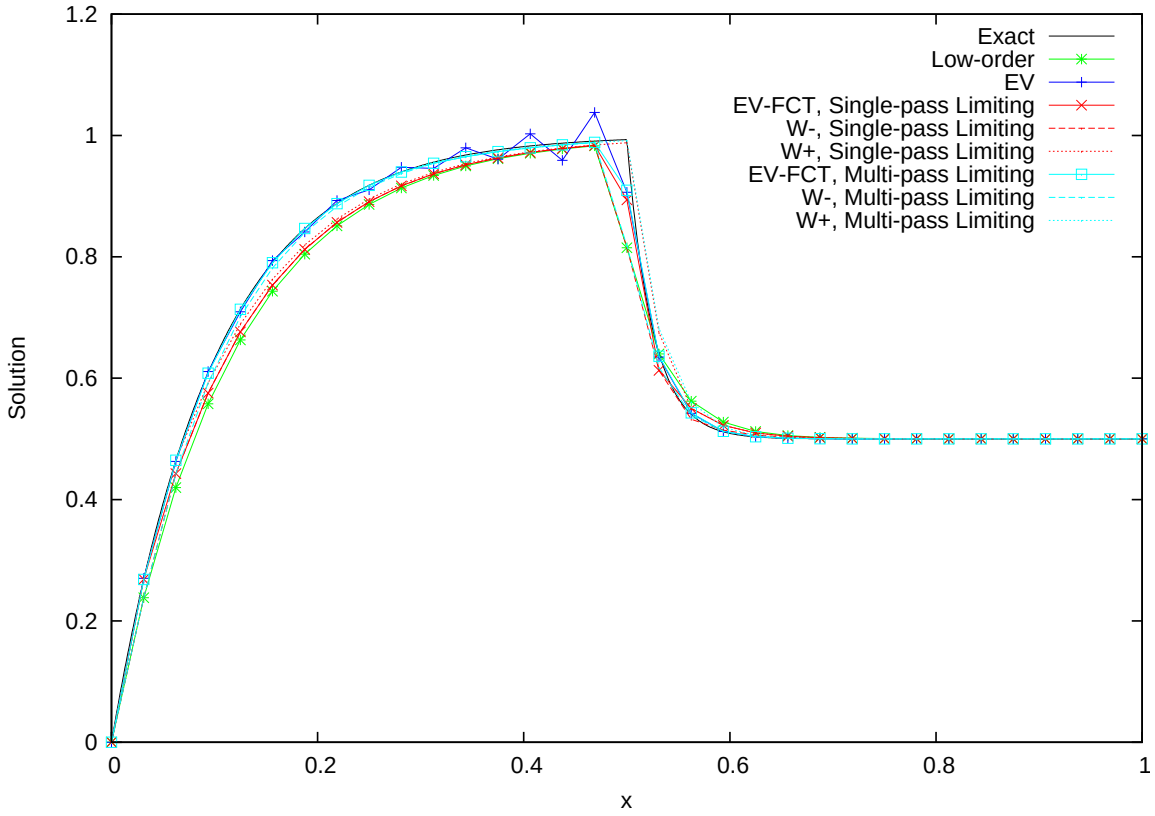


Figure 3.35: Comparison of Steady-State FCT Solutions for the Interface Test Problem Obtained with Upwind Analytic Solution Bounds with Single-Pass and Multi-Pass Limiting

Table 3.24: FCT Iterations Required for Different Solution Bounds for the Interface Test Problem

Bounds	*Limiting*	*Iterations*	*Limiter Passes Per Iteration*
Original	Single-pass	23	1
Original	Multi-pass	58	≈ 10
Upwind	Single-pass	15	1
Upwind	Multi-pass	7	≈ 19
Modified	Single-pass	23	1

3.2.11 Three-Region Test Problem

This is a 1-D problem that consists of a domain with 3 regions of differing saturation values $\frac{q}{\sigma}$. This is used to test both reaction terms and source terms. Table 3.25 summarizes the test parameters.

Table 3.25: Three-Region Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0, 1)$
Initial Conditions	$u_0(x) = 0$
Boundary Conditions	$u(x,t) = u_{inc} = 1$
Direction	$\mathbf{\Omega} = \mathbf{e}_x$
Cross Section	$\sigma(\mathbf{x}) = \begin{cases} \sigma_0, & x \in [x_0, x_1] \\ \sigma_1, & x \in (x_1, x_2] \\ \sigma_2, & x \in (x_2, x_3] \end{cases}$, $\begin{bmatrix} \sigma_0 \\ \sigma_1 \\ \sigma_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 40 \\ 20 \end{bmatrix}$ $\begin{bmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 0 \\ 0.3 \\ 0.6 \\ 1 \end{bmatrix}$
Source	$q(\mathbf{x}, t) = \begin{cases} q_0, & x \in [x_0, x_1] \\ q_1, & x \in (x_1, x_2] \\ q_2, & x \in (x_2, x_3] \end{cases}$, $\begin{bmatrix} q_0 \\ q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 5 \\ 20 \end{bmatrix}$
Speed	$v = 1$
Exact Solution	(Equation (3.17))

Table 3.26 shows the run parameters used. The boundary conditions are chosen to be strong Dirichlet with $L_i^+ = L_i^- = 1$ for the Dirichlet node. A comparison of FCT solutions for different imposed solution bounds are given by Figures 3.36 through 3.39, which respectively, correspond to the DMP solution bounds, the analytic bounds given by Equation (3.23), the modification to those analytic bounds, given by Equation (3.24), and the upwind bounds given by Equation (3.25). These

results show that for this test problem, the low-order DMP solution bounds give a better result than the analytic bounds: the analytic bounds have the overly conservative q/σ bound in them, as discussed in Section 3.2.10, and as a result, over the course of the transient, the upper bound is increased due to the presence of the source, eventually fully allowing the oscillation occurring at the interface of the first and second region. The modification to the analytic bounds suggested in Section 3.2.10 avoids this issue, and produces results superior to those obtained using the low-order DMP bounds. Using upwind bounds for this problem proves overly restrictive for this test problem; the resulting FCT solution is only marginally improved over the low-order solution.

Table 3.26: Three-Region Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 32$
Time Discretization	SSPRK33
End Time	$t = 1$
CFL Number	$\nu = 0.5$
Boundary Conditions	Strong Dirichlet with $L_i^- = L_i^+ = 1$
Entropy Function	$\eta(u) = \frac{1}{2}u^2$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$
FCT Solution Bounds	(varies by run)

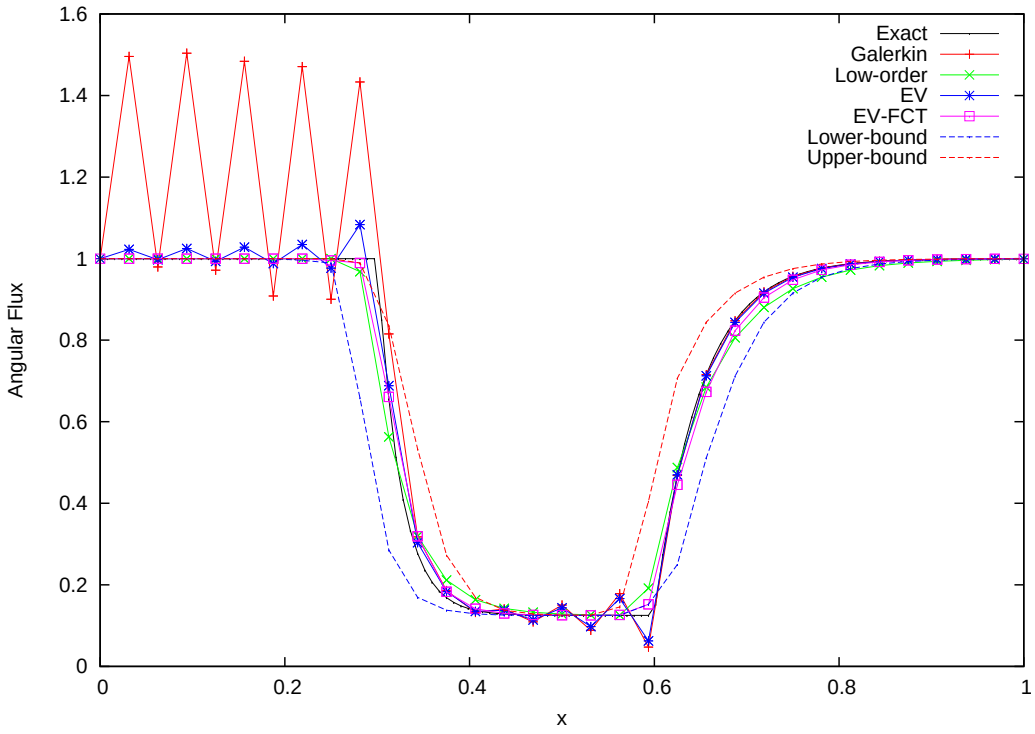


Figure 3.36: Comparison of Solutions for the 3-Region Test Problem Using SSPRK33 and DMP Solution Bounds

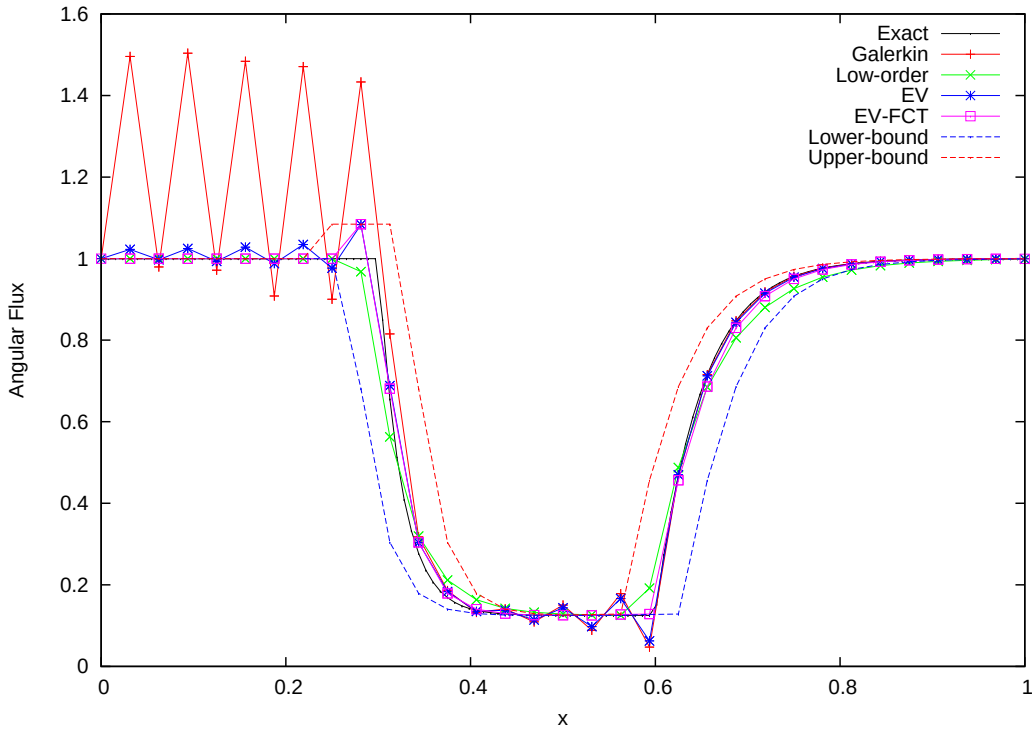


Figure 3.37: Comparison of Solutions for the 3-Region Test Problem Using SSPRK33 and Analytic Solution Bounds

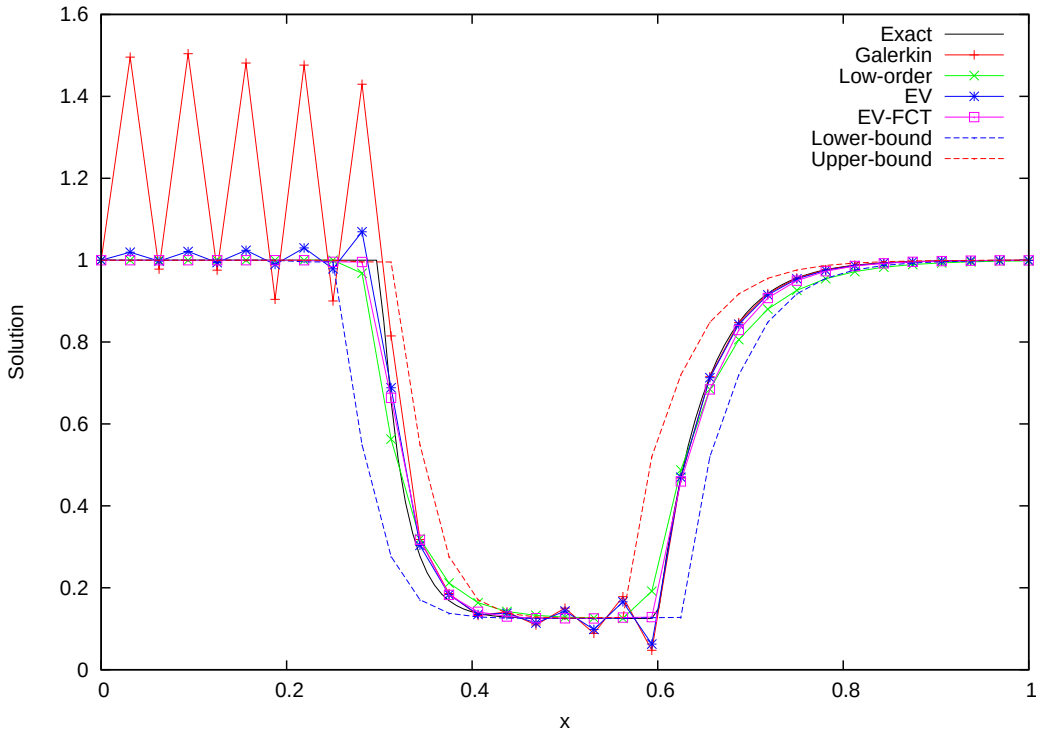


Figure 3.38: Comparison of Solutions for the 3-Region Test Problem Using SSPRK33 and Modified Analytic Solution Bounds

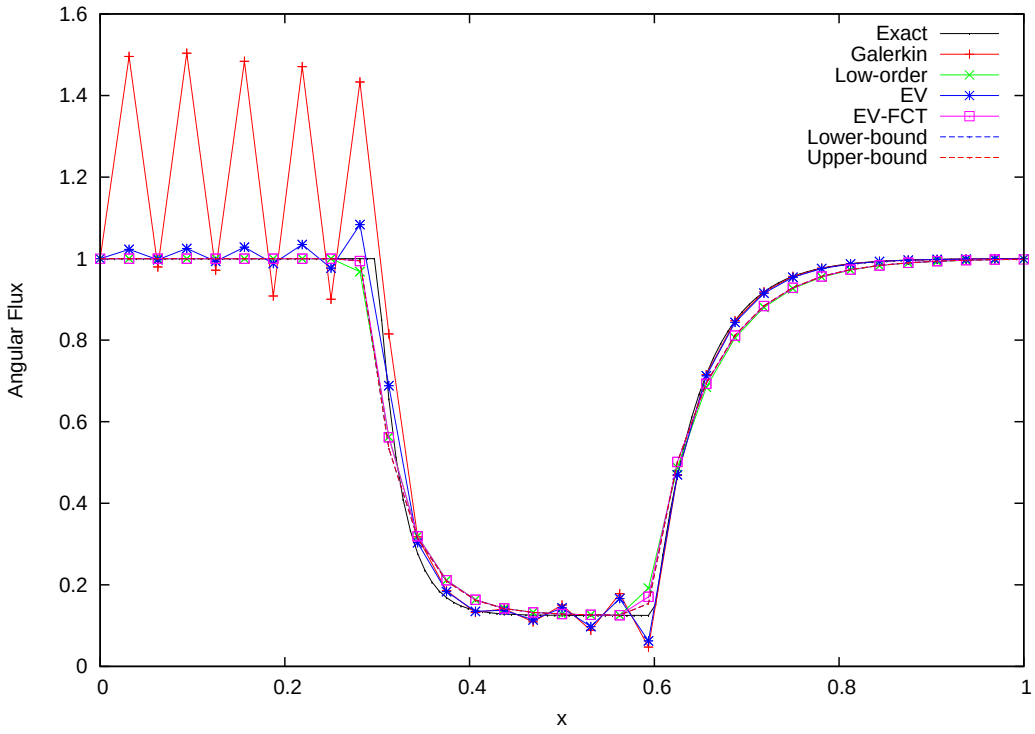


Figure 3.39: Comparison of Solutions for the 3-Region Test Problem Using SSPRK33 and Upwind Analytic Solution Bounds

3.2.12 Summary

This section provides a summary of the scalar transport results. As discussed in Section 3.2.1, there are a number of parameters that influence the success of the scalar FCT algorithm, so this section attempts to analyze what the optimal FCT configuration might be.

Firstly, it is strongly recommended to use the entropy viscosity method as the high-order scheme instead of the standard Galerkin method that lacks any artificial dissipation. It should be noted that entropy-based artificial viscosity is the main component of a successful CFEM scalar conservation law solution; this is the component which ensures convergence to the entropy solution. The FCT algorithm is best viewed as a conservative clean-up procedure for mitigating spurious oscillations and preventing negativities. As the mesh is refined, the entropy viscosity solution converges to the entropy solution, which lacks any spurious oscillations or negativities, but for coarser meshes, some of these artifacts are still present, hence the need for FCT. Thus it is preferred to use EV-FCT over Galerkin-FCT. One can find cases/configurations in which the Galerkin-FCT solution is more accurate EV-FCT solution, but this is not the general case. Also, one might find that an oscillatory solution has a smaller L^1 and/or L^2 error than a non-oscillatory solution, but that the non-oscillatory is more favorable qualitatively. Galerkin-FCT is more vulnerable to the well-known FCT phenomenon known as "terracing", "stair-stepping", or "plateauing"; this effect is associated with the limitation of large oscillations. The EV method mitigates or eliminates oscillations that the Galerkin method encounters, and thus the FCT algorithm for EV encounters smaller magnitude oscillations in the high-order solution, and the stair-stepping effect is decreased from that of Galerkin FCT.

For explicit time discretizations, it is strongly recommended to use a higher-order SSPRK scheme as opposed to explicit Euler. Explicit Euler is particularly vulnerable to the formation of spurious oscillations of large magnitude and thus FCT solutions are more vulnerable to stair-stepping. High-order SSPRK schemes such as SSPRK33 are expressed as a sequence of explicit Euler steps so that the same methodology can be used in each step, and usage of these high-order time discretizations reduces spurious oscillations even without artificial viscosity or FCT.

Implicit time discretizations and steady-state often suffer from convergence difficulties. The nonlinear scheme used for the EV solution and the FCT solution is a type of fixed-point iteration, which is arguably the simplest and slowest-converging nonlinear iteration scheme, but the lack of convergence in some cases is due to the fact that the imposed solution bounds are implicit. When the FCT solution iterate changes, the solution bounds change as well. In some cases an FCT solution iterate produces solution bounds that lead to a reversal of antidiffusive fluxes made in the previous iteration, and then this can repeat indefinitely, preventing convergence. One can try using a relaxation factor in solution updates, but this is not a solution to the underlying issue, and thus it may or may not be a successful approach for a particular problem. For implicit time discretizations, convergence difficulties are most prominent when high CFL numbers are used; in this case, the solution bounds are wider, and the limiting coefficient values have more opportunity to vary. For both implicit time discretizations and steady-state, convergence difficulties are most prominent for multi-dimensional problems; this is because the solution bounds have a greater number of degrees of freedom coupled for a given node - the neighborhoods around each node are larger. With these issues considered, implicit FCT currently requires more research to become a reliable method.

Recall that there were a number of different approaches for imposing incoming

flux boundary conditions. Strong Dirichlet BC for example have the issue in FCT that the conservation property is not satisfied unless the antidiffusive fluxes from Dirichlet nodes are completely canceled. Accurate solutions can still be obtained by ignoring this requirement, and often the cancellation of Dirichlet antidiffusive fluxes leads to less accurate solutions. Weak Dirichlet BC are found to be very inaccurate, especially when the values in the vicinity are deemed very important; thus they are not recommended for general use. However, using weak Dirichlet BC with a boundary penalty was found to be effective, without sacrificing the conservation property or the opportunity to accept antidiffusion from incoming boundary nodes. It is thus recommended for general use to use weak Dirichlet BC with a boundary penalty.

There were a number of different solution bounds considered. The low-order DMP bounds were initially considered because for fully explicit time discretization, they automatically satisfied the fail-safe conditions on the antidiffusion bounds, as discussed in Section 2.7.3.2; however, they were found to prevent second-order spatial accuracy in many cases because these bounds were proven to be inaccurate when a reaction term is present. It was found that the fail-safe condition could be enforced, no matter what solution bounds are imposed, by using Equation (2.151), and the bounds derived using the method of characteristics (MoC) were found to overcome the issues of the low-order DMP bounds and thus obtain second-order accuracy. For general use, it is recommended to use the MoC bounds in favor of the low-order DMP bounds. For the MoC bounds, there are two additional dimensions. Dimension one is the possibility of only evaluating the bounds along the upwind line segment instead of the spherical neighborhood, and dimension two is the modification presented by Equation (3.24). Usage of upwind solution bounds gave mixed results. The upwind line solution bounds are very tight; for some cases, the only reason that the upper and lower bounds differ is due to the enforcement of the fail-safe condition, which ef-

fectively enforces that the lower bound is not greater than the low-order solution and that the upper bound is not lesser than the low-order solution. Tight solution bounds have advantages and disadvantages. The obvious advantage is that there is less room for unphysical oscillations, so these remaining artifacts, including the stair-stepping effect, are reduced. Also, tighter bounds give smaller ranges for limiting coefficient values, which leads to a decrease in nonlinear iterations for implicit and steady-state FCT. However, a disadvantage of tight solution bounds is that sub-optimal limiters (all known single-pass limiters) will not be able to accept as much antidiffusion, even though there may be a combination of limiting coefficients that produce superior results. Thus in some cases the FCT solution can look only marginally better than the low-order solution. A remedy to this fact is to use a more optimal limiter; to achieve this, one can employ the multi-pass limiting procedure introduced in Section 2.7.4.2. This usually produces the best results; however, the computational expense of the additional passes may need to be considered. In summary, for general use, the upwind MoC solution bounds are recommended, especially if multi-pass limiting is able to be applied. The second dimension of the solution bounds is to make the modification presented by Equation (3.24), which gives a less conservative estimate for the bounds of q/σ. This modification is not supported by an analytic proof, but for the test problems in which it is applied, the solution bounds appear much more sensible. The modification only makes a difference for nodes for which both the reaction coefficient and source vary in the surrounding neighborhood. Without the modification, there are large peaks in the solution bounds of such nodes, but these peaks disappear with the modification. In some cases studied, the modification of the solution bounds had little or no effect because limitation requirements of nodes adjacent to these interface nodes were the bottleneck; however, in other test problems, this modification makes a significant difference. The recommendation here is to

make the modification if the rigorous mathematical support of the solution bounds is deemed less important.

3.3 The Shallow Water Equations

3.3.1 Overview

This section presents the results for the shallow water equations. For transient simulations, the time step size used is given as a "CFL" number ν:

$$\nu \equiv \frac{\Delta t}{\Delta t_{\text{CFL}}}, \quad \Delta t_{\text{CFL}} \equiv \frac{\Delta x_{\min}}{\lambda_{\max}}, . \tag{3.26a}$$

$$\Delta x_{\min} \equiv \min_K \Delta x_K \,, \tag{3.26b}$$

$$\lambda_{\max} \equiv \max_{\mathbf{x}\in\mathcal{D}} \left(\|\mathbf{v}(\mathbf{x})\| + a(\mathbf{x})\right) \,, \tag{3.26c}$$

where the maximum over the domain is approximated by the maximum over all quadrature points in the domain.

Note that to guarantee the invariant domain property for the low-order scheme, one still needs to verify the time step size requirement given by Equation (2.92); the condition $\nu < 1$ is not sufficient.

3.3.2 1-D Dam Break

This test problem is the classic 1-D dam break problem for the shallow water equations [21]. This is an example of a Riemann problem for the SWE: the initial data consists of constant left and right states. The problem parameters are summarized in Table 3.27.

The simulations were run using explicit Euler and SSPRK33 to a final time of $t = 2$, both methods using a CFL of 0.1. For the entropy viscosity method, the entropy residual coefficient and entropy jump coefficient were set to $c_{\mathcal{R}} = 1$ and

Table 3.27: Bathtub Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0,1)^2$
Initial Conditions	$h_0(\mathbf{x}) = 1 + e^{-250((x-0.25)^2+(y-0.25)^2)}$
	$\mathbf{v}_0(\mathbf{x}) = \mathbf{0}$
Boundary Conditions	$\nabla h(\mathbf{x},t) = 0\,, \quad \mathbf{x} \in \partial\mathcal{D}\,, \quad t > 0,$
	$\mathbf{v}\,(\mathbf{x},t) \cdot \mathbf{n} = 0\,, \quad \mathbf{x} \in \partial\mathcal{D}\,, \quad t > 0,$
Bathymetry	$b(\mathbf{x}) = 0$
Gravity	$g = 1$

$c_{\mathcal{J}} = 1$, respectively. Run parameters are summarized in Table 3.28.

Figures 3.40 and 3.41 show the height and momentum solutions, respectively, obtained using explicit Euler and 32 cells, and Figures 3.42 and 3.43 show the solutions for 256 cells. In both cases, there are a significant number of oscillations present in the entropy viscosity solutions. The FCT solution lacks most of these oscillations, but there is a significant amount of stair-stepping behavior seen in the rarefaction wave. Figures 3.44 and 3.45 show the solutions obtained using SSPRK33 time discretization with 256 cells. In this case, the entropy viscosity solution is without oscillation; the FCT solution not only adds unnecessary artificial diffusion, but also degrades the quality of the solution with respect to the entropy viscosity solution.

Table 3.28: 1-D Dam Break Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 32, 256$
Time Discretization	Explicit Euler, SSPRK33
End Time	$t = 2$
CFL Number	$\nu = 0.1$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 0.1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 0.1$

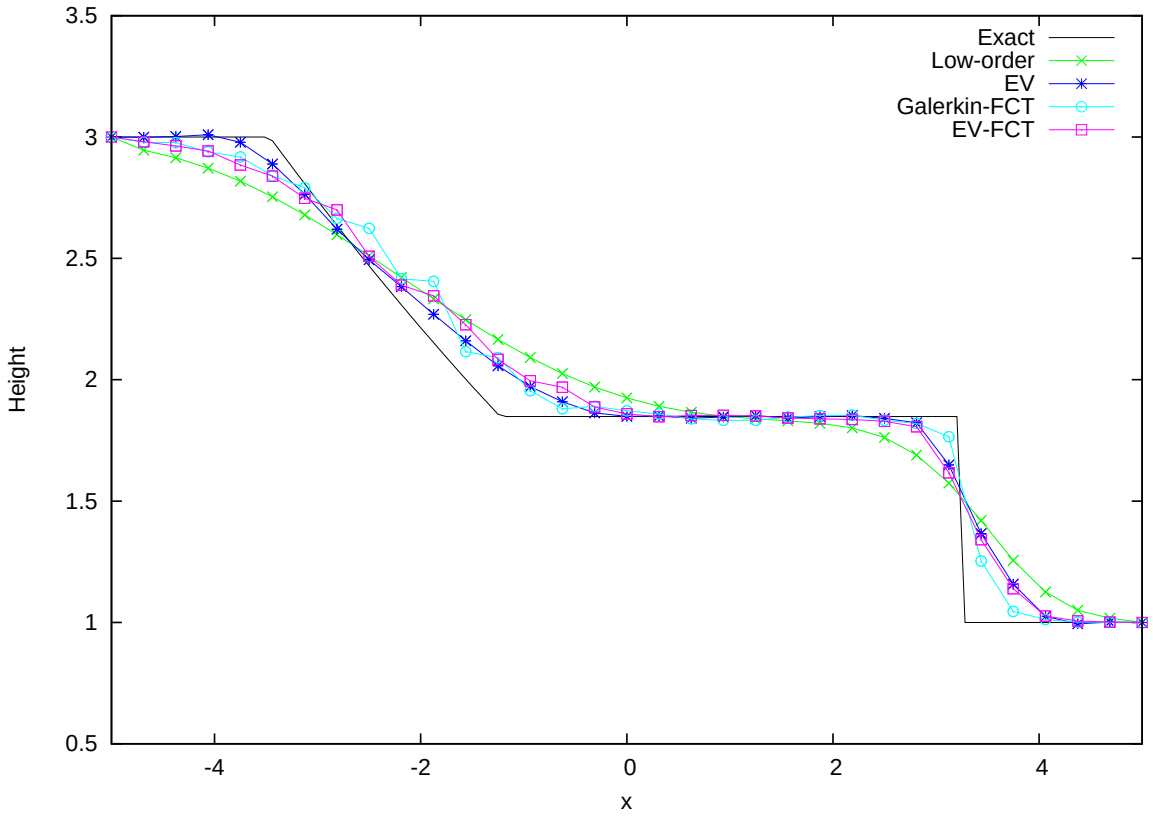


Figure 3.40: Comparison of Height Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 32 Cells

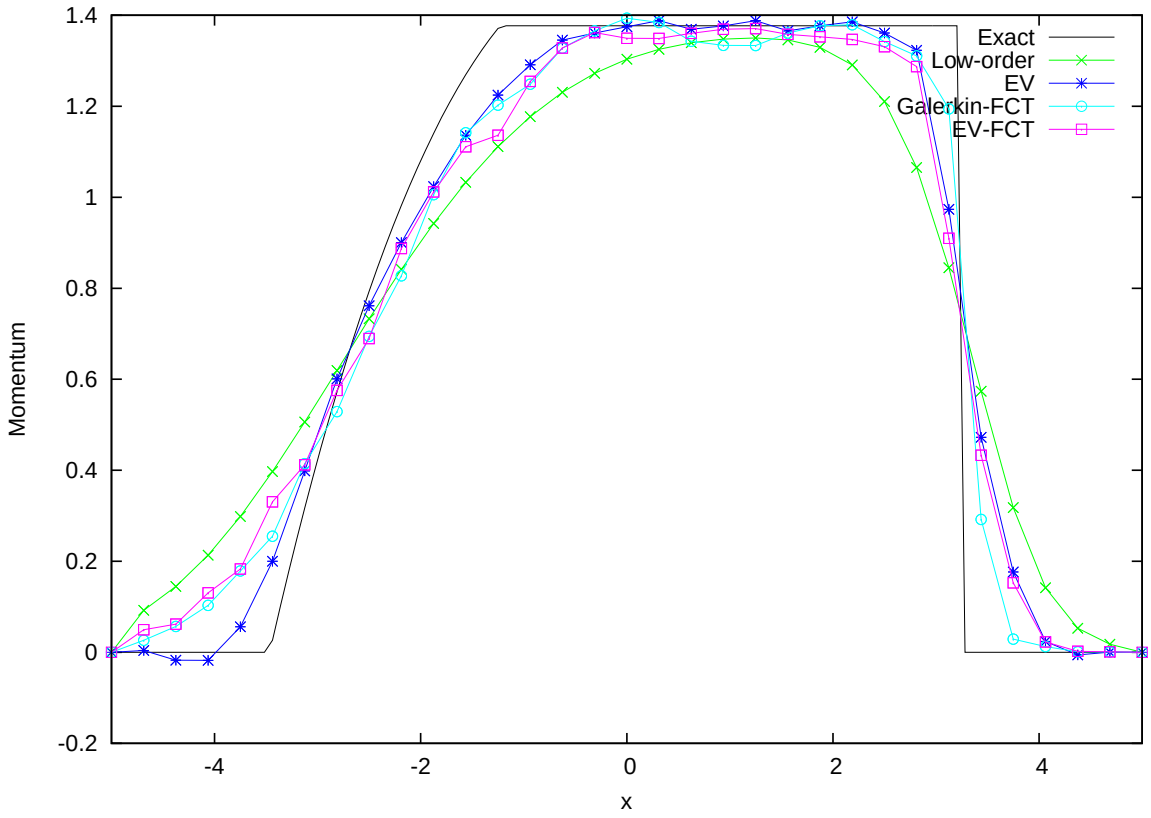


Figure 3.41: Comparison of Momentum Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 32 Cells

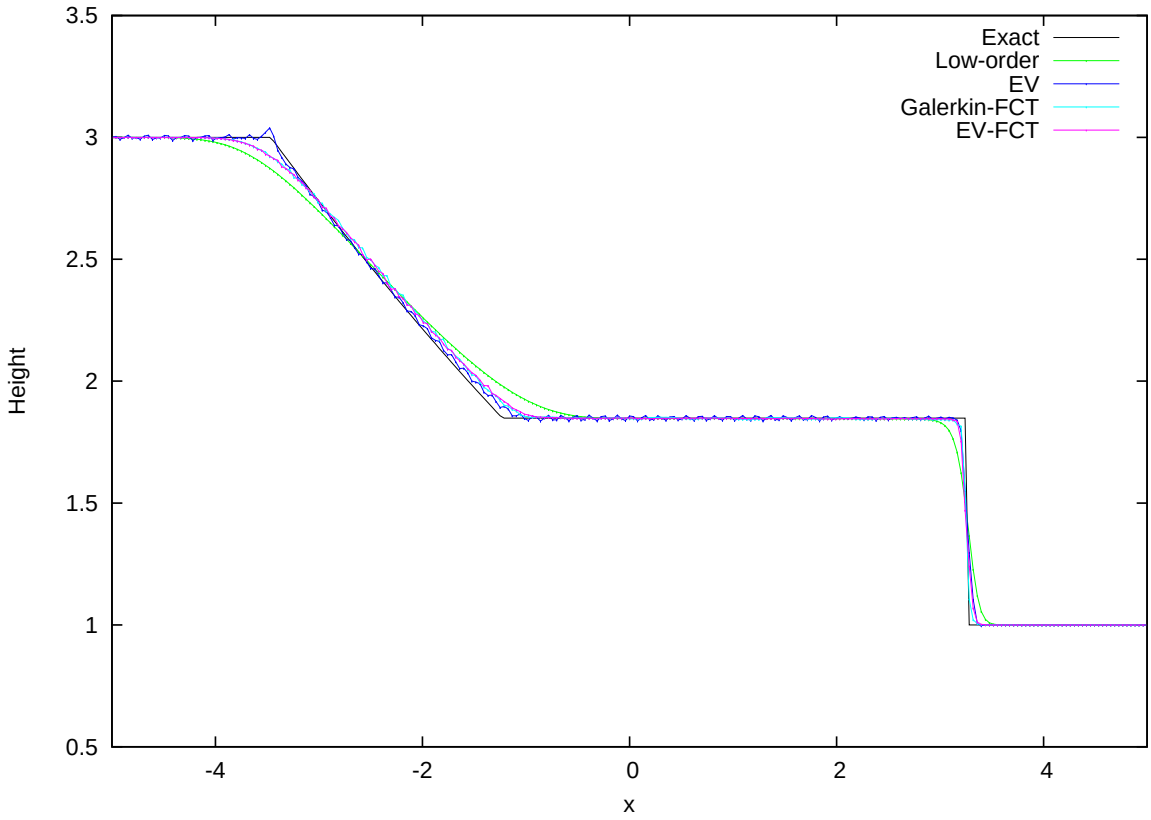


Figure 3.42: Comparison of Height Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 256 Cells

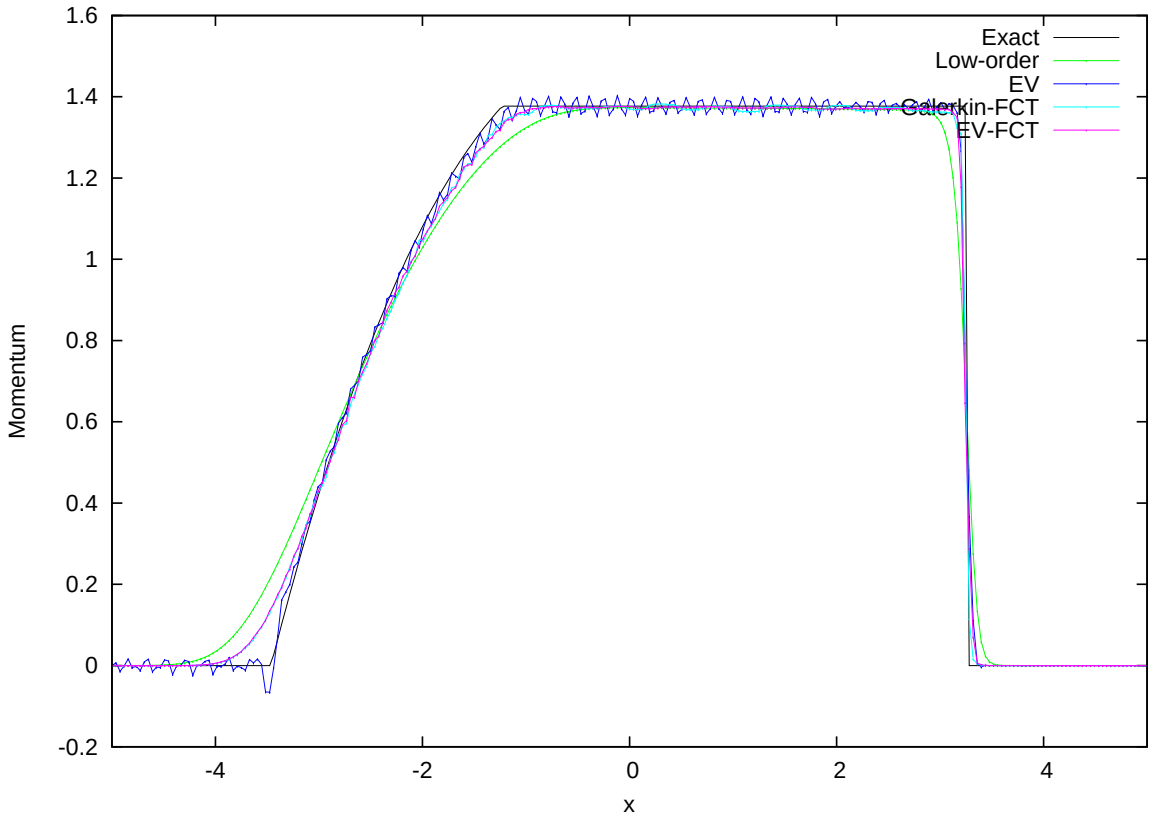


Figure 3.43: Comparison of Momentum Solutions for the 1-D Dam Break Test Problem Using Explicit Euler Time Discretization with 256 Cells

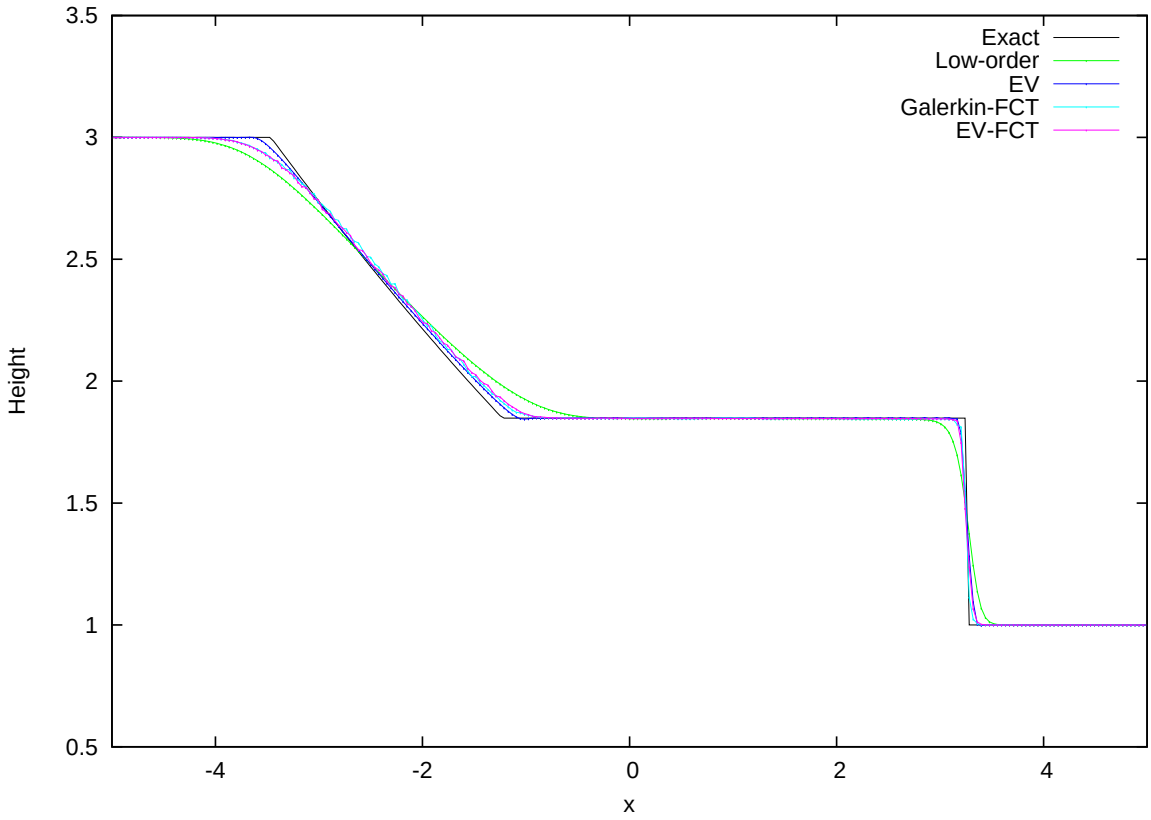


Figure 3.44: Comparison of Height Solutions for the 1-D Dam Break Test Problem Using SSPRK33 Time Discretization with 256 Cells

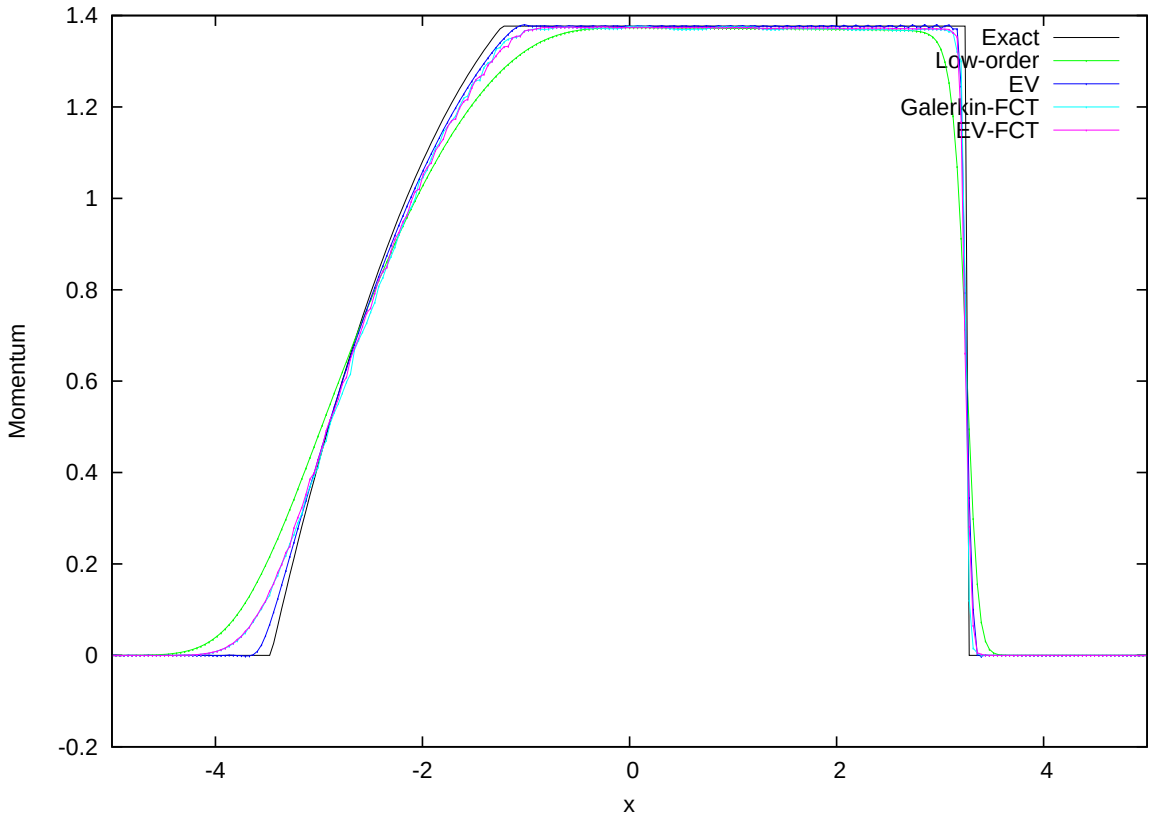
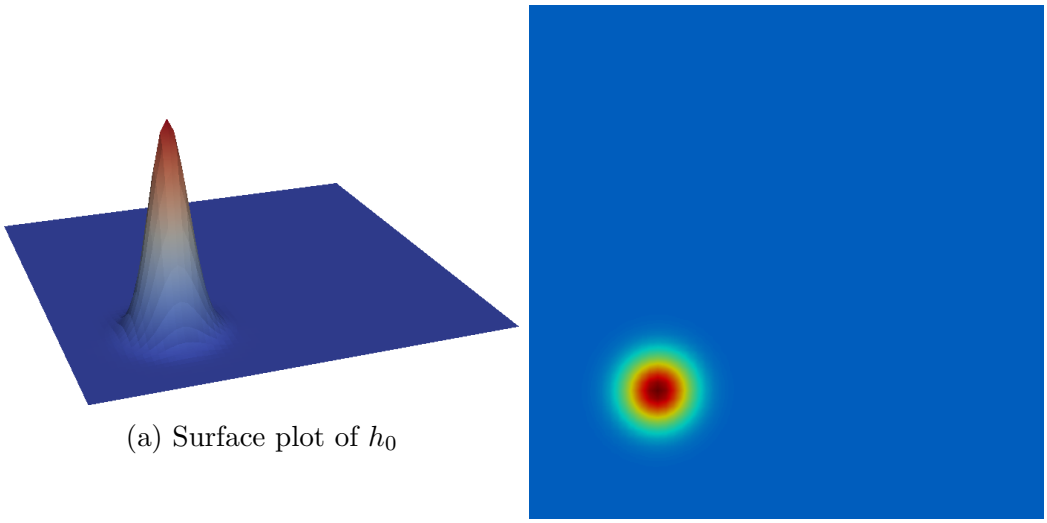


Figure 3.45: Comparison of Momentum Solutions for the 1-D Dam Break Test Problem Using SSPRK33 Time Discretization with 256 Cells

3.3.3 Bathtub Test Problem

In this test problem, an initially flat fluid surface is perturbed, and waves from this perturbation travel to the boundary and reflect back into the domain. This problem might, for example, simulate a droplet of water in a bathtub. Figure 3.46 shows this initial perturbation on the fluid surface, representing the droplet of water falling into the bathtub.



(a) Surface plot of h_0

(b) Color map of h_0

Figure 3.46: Initial Height Profile for the Bathtub Test Problem

The problem parameters are summarized in Table 3.29. The simulations were performed on a 64×64-cell mesh, with a constant time step size of $\Delta t = 0.002$, run until $t = 0.5$. For the entropy viscosity method, the entropy residual coefficient and entropy jump coefficient were set to $c_{\mathcal{R}} = 1$ and $c_{\mathcal{J}} = 1$, respectively.

Figure 3.47 shows a comparison of the solutions of the low-order invariant domain method vs. the entropy viscosity method. Solutions were obtained using explicit

Table 3.29: Bathtub Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = (0,1)^2$
Initial Conditions	$h_0(\mathbf{x}) = 1 + e^{-250((x-0.25)^2+(y-0.25)^2)}$
	$\mathbf{v}_0(\mathbf{x}) = \mathbf{0}$
Boundary Conditions	$\nabla h(\mathbf{x},t) = 0\,, \quad \mathbf{x} \in \partial\mathcal{D}\,, \quad t > 0,$
	$\mathbf{v}\,(\mathbf{x},t) \cdot \mathbf{n} = 0\,, \quad \mathbf{x} \in \partial\mathcal{D}\,, \quad t > 0,$
Bathymetry	$b(\mathbf{x}) = 0$
Gravity	$g = 1$

Euler time discretization and are compared at $t = 0.1$ and $t = 0.5$. Run parameters are summarized in Table 3.30.

Table 3.30: Bathtub Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 4096$
Time Discretization	Explicit Euler
End Time	$t = 0.1, 0.5$
Time Step Size	$\Delta t = 0.002$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 1$

The entropy viscosity solution is shown here to be much sharper than the low-order method. It is already clear at $t = 1$ how much diffusion the low-order scheme introduces; by $t = 5$, wave forms are hardly visible, as opposed to the results shown by the entropy viscosity method, where the reflected wave forms are clearly visible: four waves can be identified in Figure 3.47d:

1. the main, largest wave, going from roughly (0,0.75) to (0.75,0),

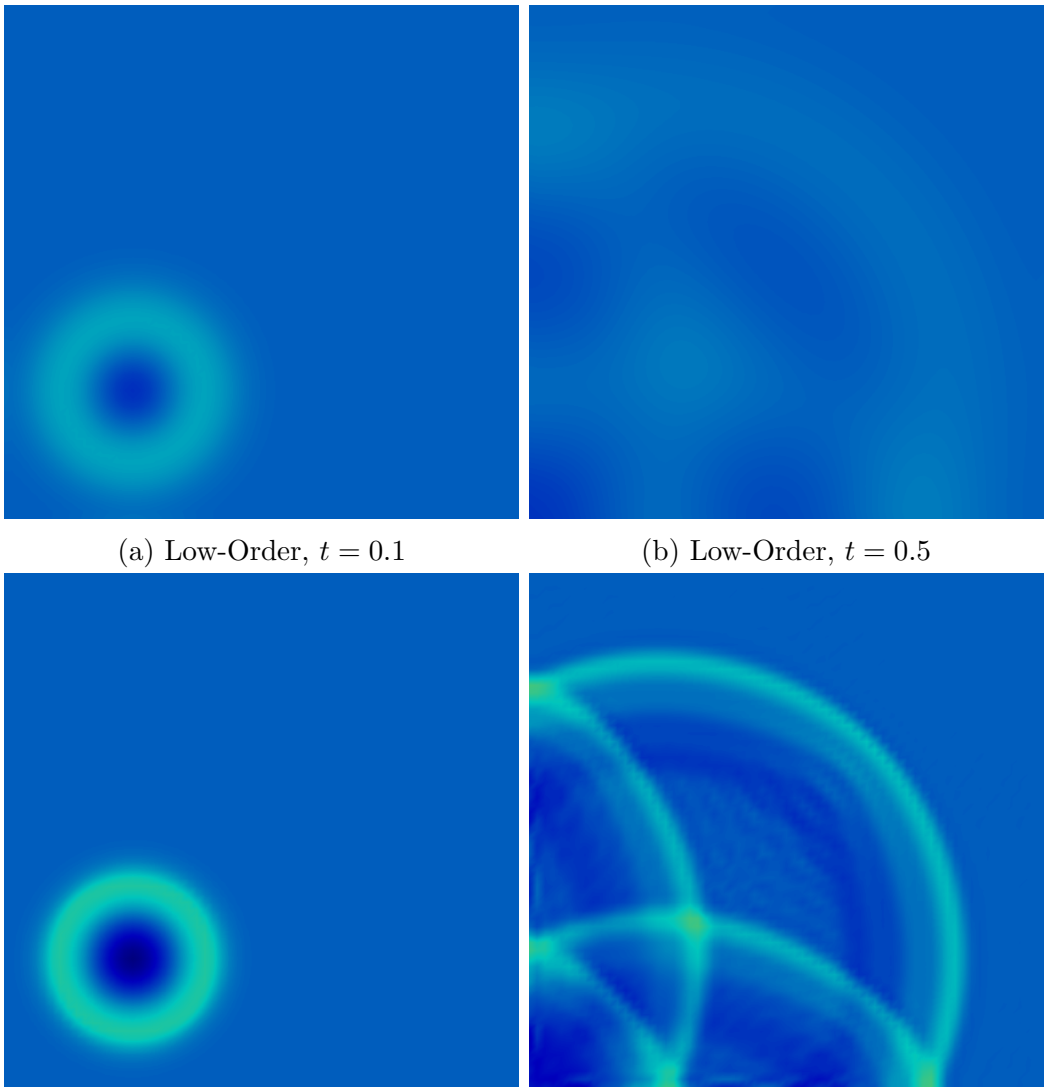
2. a wave going from roughly (0,0.25) to (0.75,0),

3. a wave going from roughly (0,0.75) to (0.25,0), and

4. the smallest wave, going from roughly (0,0.25) to (0.25,0).

If one reflects wave 2 about $y = 0$, wave 3 about $x = 0$, and wave 4 about both $y = 0$ and $x = 0$, then one constructs the circular wave form that would have formed from the droplet in an infinite domain. The Galerkin method was also applied to this problem, but when the initial circular wave from the droplet crashed into the boundary, severe oscillations developed and grew. This was also the case using SSPRK33, although the oscillations grew more slowly.

As of yet, no FCT method for the 2-D shallow water equations has been developed in this research because the characteristic transformation performed by the limiter described in this dissertation cannot simultaneously be applied in both x and y directions. In this case, the entropy viscosity method does not encounter noticeable oscillations, but more challenging test problems, such as a 2-D dam break, could reveal oscillations.



(a) Low-Order, $t = 0.1$

(b) Low-Order, $t = 0.5$

(c) EV, $t = 0.1$

(d) EV, $t = 0.5$

Figure 3.47: Comparison of Low-Order and High-Order Solutions for the Bathtub Test Problem Using Explicit Euler Time Discretization

3.3.4 2-D Dam Break

This test problem is a 2-D dam break problem for the shallow water equations. In this test problem, a walled, H-shaped region (from $x = 0$ to $x = 6$ and from $y = 0$ to $y = 6$) is initially dammed at $x = 4$, from $y = 2$ to $y = 4$, until at $t = 0$, the dam breaks. Water flows into what will be referred to as the "reservoir" region, to the right of $x = 4$. Figure 3.48 illustrates the initial height profile for this test problem and the H-shaped problem domain. The initial velocity is zero everywhere. The problem parameters are summarized in Table 3.31.

Table 3.31: 2-D Dam Break Test Problem Summary

Parameter	*Value*
Domain	$\mathcal{D} = ((0,2) \times (0,6)) \cup ((2,4) \times (2,4)) \cup ((4,6) \times (0,6))$
Initial Conditions	$h_0(\mathbf{x}) = \begin{cases} 0.05 & x < 4 \\ 0.01 & x >= 4 \end{cases}$
	$\mathbf{v}_0(\mathbf{x}) = \mathbf{0}$
Boundary Conditions	$\nabla h(\mathbf{x}, t) = 0\,, \quad \mathbf{x} \in \partial\mathcal{D}\,, \quad t > 0,$
	$\mathbf{v}\,(\mathbf{x}, t) \cdot \mathbf{n} = 0\,, \quad \mathbf{x} \in \partial\mathcal{D}\,, \quad t > 0,$
Bathymetry	$b(\mathbf{x}) = 0$
Gravity	$g = 0.05$

This simulation was run using the low-order invariant domain method and the entropy viscosity method to $t = 50$, using the SSPRK33 time discretization. The run parameters for this test problem are summarized in Table 3.32.

Figure 3.49 compares the height color maps for the low-order and high-order schemes, and Figure 3.50 shows the height surfaces, colored by the magnitude of the momentum. The entropy viscosity solution shows a much less diffusive solution - note for example, the "negative" wave front to the left of $x = 2$, the sharpness of the

Table 3.32: 2-D Dam Break Test Problem Run Parameters

Parameter	*Value*
Number of Cells	$N_{cell} = 7168$
Time Discretization	SSPRK33
End Time	$t = 50$
CFL Number	$\nu = 0.05$
Entropy Residual Coefficient	$c_{\mathcal{R}} = 1$
Entropy Jump Coefficient	$c_{\mathcal{J}} = 1$

wave front in the reservoir region to the right of $x = 4$, and the height of the reflected wave hitting the right boundary of the reservoir region. However, note the presence of some spurious oscillations at the interior corners along $x = 2$ in the surface plot of the entropy viscosity solution. These interior corners, and especially the interior corners along $x = 4$, where the initial discontinuity lies, are challenging regions and tend to give rise to severe oscillations if the time step size is too large. Note that these corner effects should be able to be mitigated by smoothing out the corner in the mesh; however, this was not performed here.

In these simulations, a CFL as small as $\nu = 0.1$ was shown to give an unstable solution, so a CFL of $\nu = 0.05$ was used. Figures 3.51 shows the low-order and entropy viscosity profiles, both on log scales (but separately scaled). From the entropy viscosity profile, one can see that the interior corner cells have the highest entropy viscosities, followed by the wave front in the reservoir region and its reflected wave.

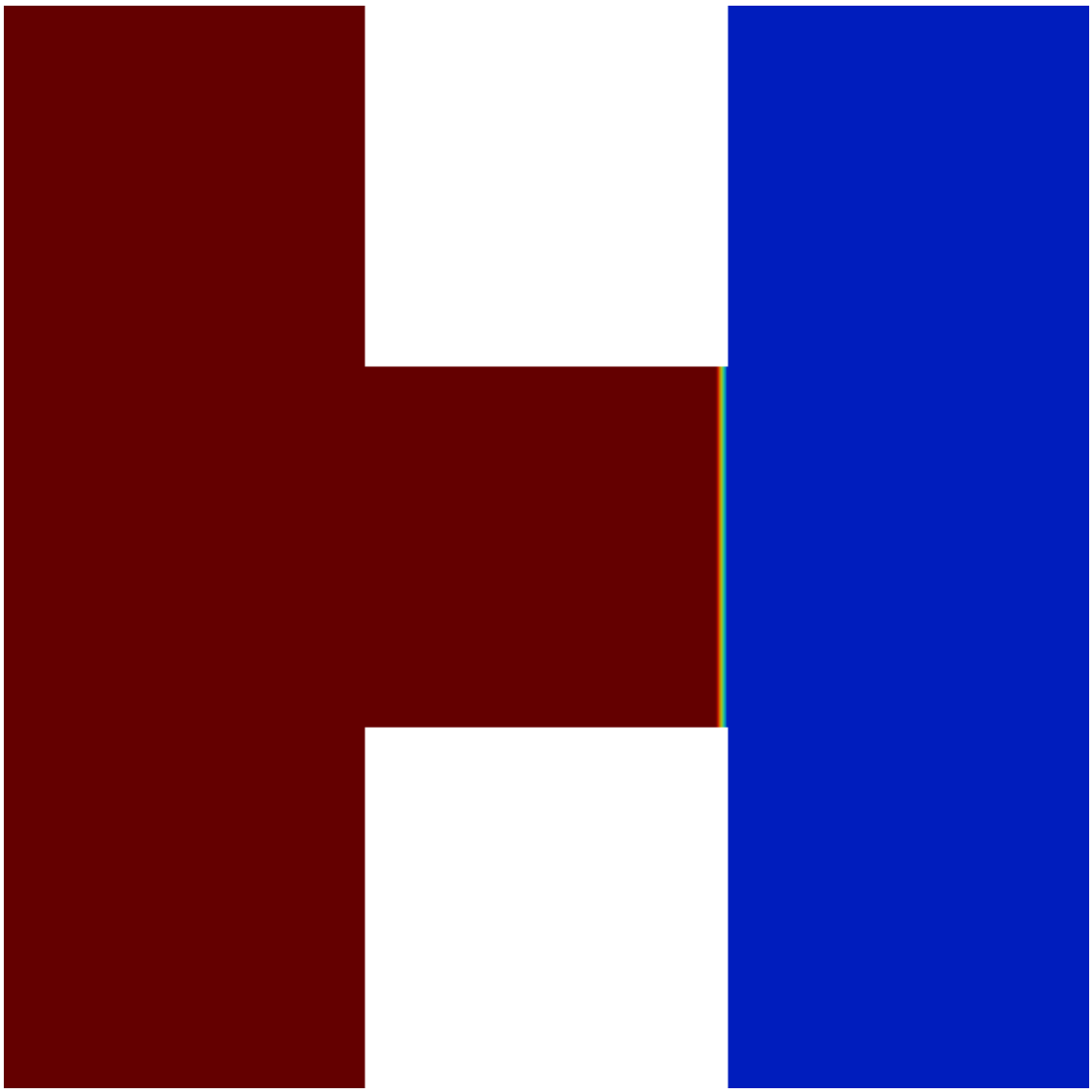
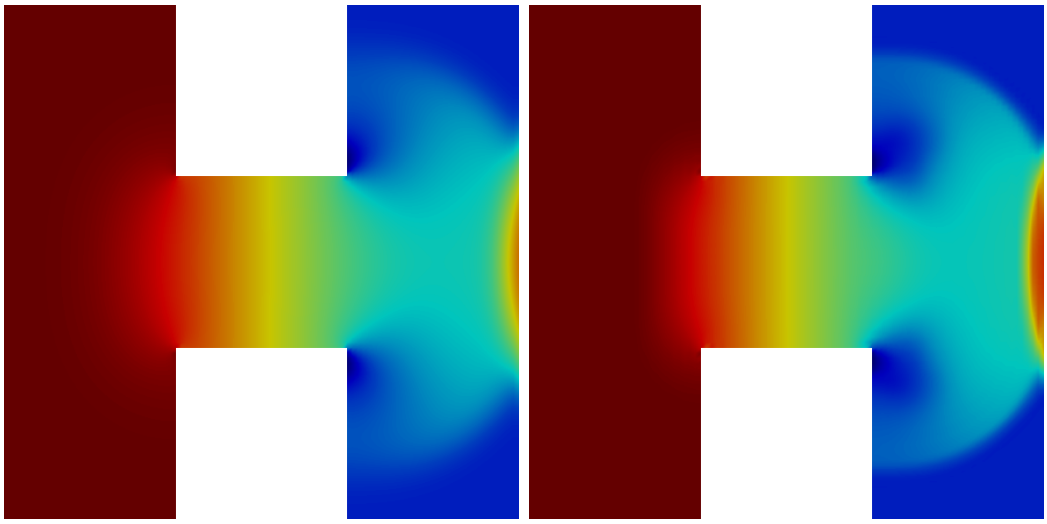
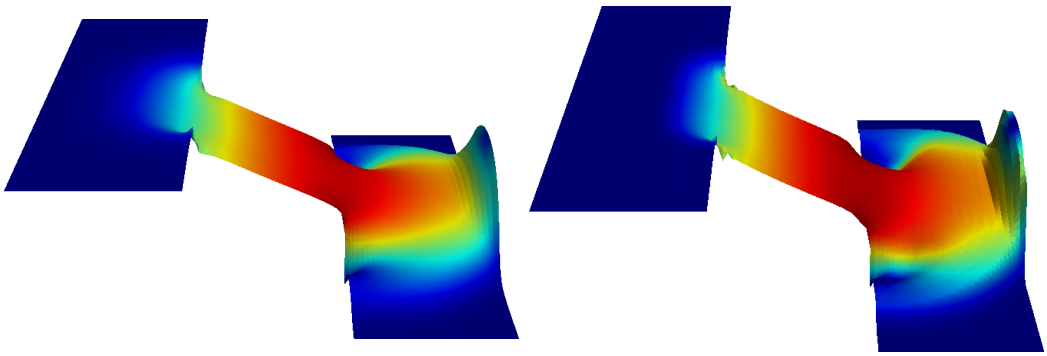


Figure 3.48: Initial Height Profile for the 2-D Dam Break Test Problem



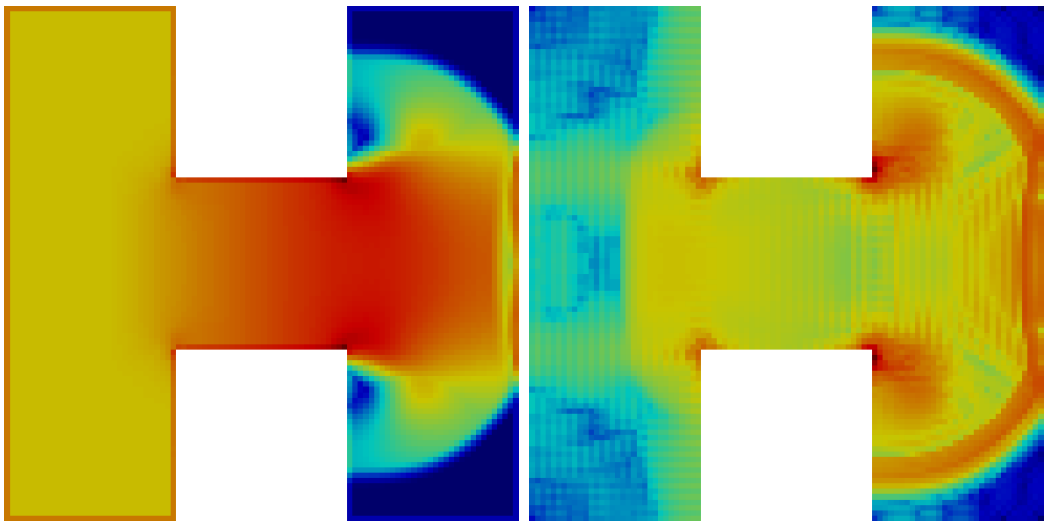
(a) Low-order scheme (b) High-order scheme

Figure 3.49: Comparison of Height Solutions for the 2-D Dam Break Test Problem



(a) Low-order scheme (b) High-order scheme

Figure 3.50: Comparison of Solution Surfaces for the 2-D Dam Break Test Problem



(a) Low-order Viscosity (b) Entropy Viscosity

Figure 3.51: Comparison of Viscosity Profiles for the 2-D Dam Break Test Problem

4. CONCLUSIONS

This research investigated a number of different numerical methods for the solution of hyperbolic PDEs with the continuous finite element method.

A first-order, positivity-preserving, DMP-satisfying method for scalar hyperbolic PDEs, recently developed by Guermond and Nazarov [11] for scalar hyperbolic PDEs, was extended to hyperbolic PDEs including a reaction term and extraneous source term.

A first-order, positivity-preserving, and domain-invariant method for systems of hyperbolic PDEs, recently developed by Guermond and Popov [14] for systems of hyperbolic PDEs, has been applied to the shallow water equations with flat bottom topography.

The entropy viscosity method developed by Guermond and others [13] was applied to scalar transport and the shallow water equations, with flat or non-flat bottom topography. Results show that addition of this entropy-based artificial dissipation results in convergence to the entropy solution and reduces the onset of spurious oscillations but in general does not eliminate them completely, and thus the entropy viscosity method is not immune to solution negativities.

The flux corrected transport (FCT) algorithm, originally developed by Boris and Book [7] was implemented in conjunction with the entropy viscosity method, as in [12]. In addition to the family of explicit SSPRK methods, steady-state and implicit θ time discretization methods were employed. For all time discretizations, the FCT algorithm could be used to guarantee the absence of solution negativities, and spurious oscillations are significantly reduced, if not eliminated entirely. The only case in which oscillations have been observed in the FCT solution is when an extraneous

source is present, which contributes the upper solution bound. The formation of spurious plateaus (also known as "terracing" or "stair-stepping") remains an open issue for FCT. This effect is generally observed when oscillations are particularly large, typically when using explicit Euler time discretization.

The selection of the solution bounds to impose in the FCT algorithm was found to be vital; usage of the low-order scheme DMP was found in general to produce first-order spatial convergence for radiation transport. Using solution bounds derived from the method of characteristics allowed second-order spatial convergence to be achieved. These analytic solution bounds have the advantage of being fully explicit; however, this is only valid for CFL less than one. Increasing the CFL above one requires widening the stencil in the min/max operations in the solution bounds, thus making it less restrictive. Thus for large CFL numbers, one must use the low-order DMP solution bounds, which would be implicit. The implicitness of the solution bounds necessitates the usage of nonlinear iteration, which can be problematic; severe convergence difficulties have been noted in many cases, and the success of the iteration process is sometimes dependent on the initial guess. These issues make implicit FCT unreliable; a remedy to these challenges has not yet been found.

The FCT algorithm was also applied to the shallow water equations, again in conjunction with the entropy viscosity method. In this case, no discrete maximum principle applies as in the scalar case. Therefore, the approach taken was to transform the system into characteristic variables to allow scalar FCT methodology to be applied. This was found to have some success; however, this approach was limited to 1-D because characteristic transformations could not be applied simultaneously in multiple directions.

REFERENCES

[1] Fukushima nuclear accident update log, April 2011.

[2] Magnitude 9.03 near the east coast of honshu, japan, 2011.

[3] Yukiya Amano. The fukushima daiichi accident. Technical report, International Atomic Energy Agency, 2015.

[4] W. Bangerth, R. Hartmann, and G. Kanschat. deal.II – a general purpose object oriented finite element library. *ACM Transactions on Mathematical Software*, 33(4):24/1–24/27, 2007.

[5] George I. Bell and Samuel Glasstone. *Nuclear Reactor Theory*. Litton Educational Publishing, Inc., 1970.

[6] R. Bernetti, V. A. Titarev, and E. F. Toro. Exact solution of the riemann problem for the shallow water equations with discontinuous bottom geometry. *J. Comput. Phys.*, 227(6):3212–3243, March 2008.

[7] Jay P. Boris and David L. Book. Flux-corrected transport i. SHASTA, a fluid transport algorithm that works. *Journal of Computational Physics*, 11:38–69, 1973.

[8] James J. Duderstadt and William R. Martin. *Transport Theory*. John Wiley & Sons, 1979.

[9] Ulrik S. Fjordholm, Siddhartha Mishra, and Eitan Tadmor. Well-balanced and energy stable schemes for the shallow water equations with discontinuous topography. *J. Comput. Phys.*, 230(14):5587–5609, June 2011.

[10] Sigal Gottlieb. On high order strong stability preserving runge-kutta and multi step time discretizations. *Journal of Scientific Computing*, 25(1), November 2005.

[11] Jean-Luc Guermond and Murtazo Nazarov. A maximum-principle preserving c^0 finite element method for scalar conservation equations. *Computational Methods in Applied Mechanics and Engineering*, 272:198–213, 2014.

[12] Jean-Luc Guermond, Murtazo Nazarov, Bojan Popov, and Yong Yang. A second-order maximum principle preserving lagrange finite element technique for nonlinear scalar conservation equations. *SIAM Journal on Numerical Analysis*, 52:2163–2182, 2014.

[13] Jean-Luc Guermond, Richard Pasquetti, and Bojan Popov. Entropy viscosity method for nonlinear conservation laws. *Journal of Computational Physics*, 230:4248–4267, 2011.

[14] Jean-Luc Guermond and Bojan Popov. Invariant domains and first-order continuous finite element approximation for hyperbolic systems. Electronic preprint, arXiv:1509.07461v1 [math.NA], 2015.

[15] Steven Hamilton and Michele Benzi. Negative flux fixups in discontinuous finite element sn transport. In *International Conference on Mathematics, Computational Methods & Reactor Physics*, 2009.

[16] David Hoff. Invariant regions for systems of conservation laws. *Transactions of the American Mathematical Society*, 289:591–610, 1985.

[17] Hiroshi Kanayama and Hiroshi Dan. A tsunami simulation of hakata bay using the viscous shallow-water equations. *Japan Journal of Industrial and Applied Mathematics*, 30(3):605–624, 2013.

[18] James T. Kirby, Fengyan Shi, Babak Tehranirad, Jeffrey C. Harris, and Stephan T. Grilli. Dispersive tsunami waves in the ocean: Model equations and sensitivity to dispersion and coriolis effects. *Ocean Modelling*, 62:39–55, 2013.

[19] D. Kuzmin, R. Löhner, and S. Turek. *Flux-Corrected Transport*. Springer-Verlag Berlin Heidelberg, Germany, first edition, 2005.

[20] K. D. Lanthrop. Spatial differencing of the transport equation: Positivity vs. accuracy. *Journal of Computational Physics*, 4:475–498, 1969.

[21] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge University Press, 2002.

[22] E. E. Lewis and W. F. Miller. *Computational Methods of Neutron Transport*. American Nuclear Society, La Grange Park, IL, 1993.

[23] Colin Barr Macdonald. Constructing high-order runge-kutta methods with embedded strong-stability-preserving pairs. Master's thesis, Acadia University, August 2003.

[24] Peter Maginot. A nonlinear positive extension of the linear discontinuous spatial discretization of the transport equation. Master's thesis, Texas A&M University, December 2010.

[25] Peter Maginot. A non-negative, non-linear petrov-galerkin method for bilinear discontinuous differencing of the sn equations. In *Joint International Conference on Mathematics and Computation, Supercomputing in Nuclear Applications, and the Monte Carlo Method (M&C 2015)*, Nashville, TN, 2015.

[26] B. M. Minor and K. A. Matthews. Exponential characteristic spatial quadrature for discrete ordinates radiation transport with rectangular cells. *Nuclear Science*

and Engineering, 120:165–186, 1995.

[27] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer-Verlag Berlin Heidelberg, 3rd edition, 2009.

[28] W. F. Walters and T. A. Wareing. An accurate, strictly-positive, nonlinear characteristic scheme for the discrete ordinates equations. *Transport Theory and Statistical Physics*, 25(2):197–215, 1996.

[29] Wallace F. Walters and Todd A. Wareing. A nonlinear positive method for solving the transport equation on coarse meshes. In *8th International Conference on Radiation Shielding*, Arlington, TX, 1994.

[30] Todd A. Wareing. An exponential discontinuous scheme for discrete-ordinate calculations in cartesian geometries. In *Joint International Conference on Mathematical Methods and Supercomputing in Nuclear Applications*, Saratoga Springs, NY, 1997.

[31] Steven T. Zalesak. Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics*, 31:335–362, 1979.

APPENDIX A

DERIVATION OF LOCAL SOLUTION BOUNDS FOR LINEAR TRANSPORT USING THE METHOD OF CHARACTERISTICS

A.1 Introduction

In this section, an analytic local maximum principle is derived for scalar conservation laws having a constant, linear flux $\mathbf{f}(u)$, i.e., $\mathbf{f}(u) = \mathbf{v}u$ with $\nabla \cdot (\mathbf{v}u) = \mathbf{v} \cdot \nabla u$, where $\mathbf{v}$ is the constant velocity field. This analysis is valid for radiation transport, where the constant velocity field is $\mathbf{v} = v\mathbf{\Omega}$, with v being the radiation speed.

The analytic DMPs are derived using the method of characteristics, whereby paths in the $x - t$ plane are found, along which the governing PDE becomes an ODE. This is simple for the case of constant linear transport because in this case the characteristics are constant.

A.2 Integral Form of the Linear Transport Equation

Theorem A.2.1 (Integral Form of the Linear Transport Equation) *An implicit solution to the initial value problem*

$$\frac{1}{v}\frac{\partial u}{\partial t} + \mathbf{\Omega} \cdot \nabla u(\mathbf{x}, t) + \sigma(\mathbf{x})u(\mathbf{x}, t) = q(\mathbf{x}, t), \qquad u(\mathbf{x}, 0) = u_0(\mathbf{x}) \tag{A.1}$$

is the following:

$$u(\mathbf{x}, t) = u_0(\mathbf{x} - vt\mathbf{\Omega})e^{-\int_0^t \sigma(\mathbf{x}-v(t-t')\mathbf{\Omega})vdt'} + \int_0^t q(\mathbf{x} - v(t-t')\mathbf{\Omega}, t')e^{-\int_{t'}^t \sigma(\mathbf{x}-v(t-\bar{t})\mathbf{\Omega})vd\bar{t}} vdt' . \tag{A.2}$$

Proof This proof will proceed by using the method of characteristics. The position

$\mathbf{x}$ will be regarded as a function of time: $\mathbf{x} = \mathbf{x}(t)$. The characteristic $\mathbf{x}(t)$ is the solution of the following initial value problem:

$$\frac{d\mathbf{x}}{dt} = v\mathbf{\Omega}\,, \qquad \mathbf{x}(0) = \mathbf{x}_0\,,$$

which is

$$\mathbf{x}(t) = \mathbf{x}_0 + vt\mathbf{\Omega}\,.$$

Taking the derivative of $u(\mathbf{x}(t), t)$ gives

$$\begin{aligned} \frac{du}{dt} &= \frac{\partial u}{\partial t} + \nabla \cdot u(\mathbf{x}(t), t)\frac{d\mathbf{x}}{dt} \\ &= \frac{\partial u}{\partial t} + v\mathbf{\Omega} \cdot \nabla u(\mathbf{x}(t), t)\,, \end{aligned}$$

which when combined with the PDE in Equation (A.1), gives

$$\frac{du}{dt} + v\sigma(\mathbf{x}(t))u(\mathbf{x}(t), t) = vq(\mathbf{x}(t), t)\,. \tag{A.3}$$

This is a 1st-order linear ODE, which may be solved using an integrating factor

$$\mu(t) = e^{\int_0^t \sigma(\mathbf{x}(t'))v dt'}\,.$$

Multiplying both sides of Equation (A.3) by this integrating factor and using the product rule,

$$\frac{d}{dt}\left[u(\mathbf{x}(t), t)\mu(t)\right] = vq(\mathbf{x}(t), t)\mu(t)\,,$$

and integrating from 0 to t gives

$$u(\mathbf{x}(t),t)\mu(t) - u(\mathbf{x}(0),0)\mu(0) = \int_0^t q(\mathbf{x}(t'),t')\mu(t')v dt' \,.$$

Simplifying,

$$\begin{aligned} u(\mathbf{x}(t),t) &= u(\mathbf{x}(0),0)e^{-\int_0^t \sigma(\mathbf{x}(t'))v dt'} + \left(\int_0^t q(\mathbf{x}(t'),t') e^{\int_0^{t'} \sigma(\mathbf{x}(\bar{t}))v d\bar{t}} v dt' \right) e^{-\int_0^t v\sigma(\mathbf{x}(t'))dt'} \,, \\ &= u(\mathbf{x}(0),0)e^{-\int_0^t \sigma(\mathbf{x}(t'))v dt'} + \int_0^t q(\mathbf{x}(t'),t') e^{-\int_{t'}^t \sigma(\mathbf{x}(\bar{t}))v d\bar{t}} v dt' \,. \end{aligned}$$

Finally, expressing $\mathbf{x}(t)$ in terms of $\mathbf{x}$, v, $\mathbf{\Omega}$, and t gives

$$\begin{aligned} u(\mathbf{x},t) = u_0(\mathbf{x} - vt\mathbf{\Omega}) &e^{-\int_0^t \sigma(\mathbf{x}-v(t-t')\mathbf{\Omega})v dt'} \\ &+ \int_0^t q(\mathbf{x}-v(t-t')\mathbf{\Omega},t') e^{-\int_{t'}^t \sigma(\mathbf{x}-v(t-\bar{t})\mathbf{\Omega})v d\bar{t}} v dt' \,. \end{aligned}$$

A.3 Local Maximum Principles

Before giving an analytic local discrete maximum principle, a local maximum principle applying to a general region is given by the following theorem.

Theorem A.3.1 (Analytic Local Maximum Principle) *Let $L(\mathbf{x},\tau)$ be the line segment that spans between $\mathbf{x} - v\tau\mathbf{\Omega}$ and $\mathbf{x}$:*

$$L(\mathbf{x},\tau) \equiv \left\{ \mathbf{y} \in \mathbb{R}^d : \mathbf{y} = \mathbf{x} - vt\mathbf{\Omega} \,, \qquad t \in (0,\tau) \right\} \,. \tag{A.4}$$

See Figure A.1 for an illustration. The following local maximum principle is valid

for the solution to the problem given by Equation (A.1)*:*

$$u_{min} \leq u(\mathbf{x}, \tau) \leq u_{max}\,, \tag{A.5a}$$

$$u_{min} \equiv \begin{cases} u_0(\mathbf{x} - v\tau\boldsymbol{\Omega})e^{-v\tau\sigma_{\max,L}} + \frac{q_{\min,L}}{\sigma_{\max,L}}(1 - e^{-v\tau\sigma_{\max,L}})\,, & \sigma_{\max,L} \neq 0 \\ u_0(\mathbf{x} - v\tau\boldsymbol{\Omega}) + v\tau q_{\min,L}\,, & \sigma_{\max,L} = 0 \end{cases}, \tag{A.5b}$$

$$u_{max} \equiv \begin{cases} u_0(\mathbf{x} - v\tau\boldsymbol{\Omega})e^{-v\tau\sigma_{\min,L}} + \frac{q_{\max,L}}{\sigma_{\min,L}}(1 - e^{-v\tau\sigma_{\min,L}})\,, & \sigma_{\min,L} \neq 0 \\ u_0(\mathbf{x} - v\tau\boldsymbol{\Omega}) + v\tau q_{\max,L}\,, & \sigma_{\min,L} = 0 \end{cases}, \tag{A.5c}$$

$$\sigma_{\min,L} \equiv \min_{\mathbf{y}\in L(\mathbf{x},\tau)} \sigma(\mathbf{y})\,, \quad \sigma_{\max,L} \equiv \max_{\mathbf{y}\in L(\mathbf{x},\tau)} \sigma(\mathbf{y})\,, \tag{A.5d}$$

$$q_{\min,L} \equiv \min_{\mathbf{y}\in L(\mathbf{x},\tau)} q(\mathbf{y})\,, \quad q_{\max,L} \equiv \max_{\mathbf{y}\in L(\mathbf{x},\tau)} q(\mathbf{y})\,. \tag{A.5e}$$

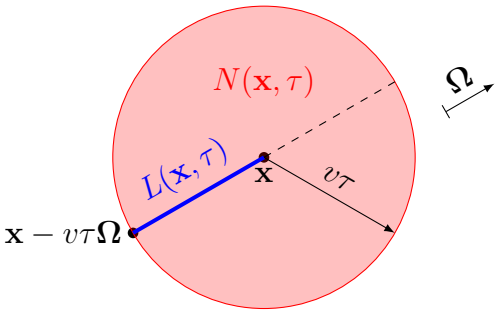


Figure A.1: Illustration of Neighborhoods $L(\mathbf{x},\tau)$ and $N(\mathbf{x},\tau)$

Proof Rewriting Equation (A.2) with $t = \tau$ gives

$$u(\mathbf{x},\tau) = u_0(\mathbf{x} - v\tau\boldsymbol{\Omega})e^{-\int_0^\tau \sigma(\mathbf{x}-v(\tau-t')\boldsymbol{\Omega})vdt'} + \int_0^\tau q(\mathbf{x} - v(\tau - t')\boldsymbol{\Omega}, t')e^{-\int_{t'}^\tau \sigma(\mathbf{x}-v(\tau-\bar{t})\boldsymbol{\Omega})vd\bar{t}} vdt'\,.$$

One can bound the first term in the right-hand-side of Equation (A.2) by considering the maximum and minimum cross section on the line segment $L(\mathbf{x}, \tau)$ for the lower and upper bounds, respectively:

$$u_0(\mathbf{x} - v\tau\mathbf{\Omega})e^{-v\tau\sigma_{\max,L}} \leq u_0(\mathbf{x} - v\tau\mathbf{\Omega})e^{-\int_0^\tau \sigma(\mathbf{x}-v(\tau-t')\mathbf{\Omega})vdt'} \leq u_0(\mathbf{x} - v\tau\mathbf{\Omega})e^{-v\tau\sigma_{\min,L}} .$$

The source term can be bounded as follows:

$$\begin{aligned}
u_q &\equiv \int_0^\tau q(\mathbf{x} - v(\tau - t')\mathbf{\Omega}, t')e^{-\int_{t'}^\tau \sigma(\mathbf{x}-v(\tau-\bar{t})\mathbf{\Omega})vd\bar{t}} vdt' \\
&\leq q_{\max,L} \int_0^\tau e^{-\int_{t'}^\tau \sigma(\mathbf{x}-v(\tau-\bar{t})\mathbf{\Omega})vd\bar{t}} vdt' \\
&\leq q_{\max,L} \int_0^\tau e^{-\sigma_{\min,L}\int_{t'}^\tau vd\bar{t}} vdt' \\
&= q_{\max,L} \int_0^\tau e^{-v(\tau-t')\sigma_{\min,L}} vdt' \\
&= q_{\max,L} e^{-v\tau\sigma_{\min,L}} \int_0^\tau e^{\sigma_{\min,L}vt'} vdt' \\
&= \begin{cases} \frac{q_{\max,L}}{\sigma_{\min,L}}(1 - e^{-v\tau\sigma_{\min,L}}), & \sigma_{\min,L} \neq 0 \\ v\tau q_{\max,L}, & \sigma_{\min,L} = 0 \end{cases}
\end{aligned}$$

A similar analysis is performed for the lower bound. Putting the two components together gives the bounds given by Equation (A.5). ∎

This result gives relatively tight solution bounds; however, its use as solution bounds for FCT may prove difficult in practice (especially for multi-dimensional problems), as one must compute the solution at the point $\mathbf{x} - v\tau\mathbf{\Omega}$ and must be able to evaluate the minimum and maximum of the reaction coefficients and sources on

the line segment $L(\mathbf{x}_i, \tau)$. The following corollary loosens the solution bounds for use in a more simple implementation of solution bounds for FCT. It considers not just the upstream line segment of length $v\tau$, but the sphere of radius $v\tau$ centered at $\mathbf{x}_i$.

Corollary A.3.1 (Loose Analytic Local Maximum Principle) *Let $N(\mathbf{x}, \tau)$ denote the sphere centered at $\mathbf{x}$ with radius $v\tau$, as shown in Figure A.1:*

$$N(\mathbf{x}, \tau) \equiv \left\{ \mathbf{y} \in \mathbb{R}^d : \|\mathbf{y} - \mathbf{x}\| \leq v\tau \right\} . \tag{A.6}$$

The following, looser, local maximum principle is valid for the solution to the problem given by Equation (A.1)*:*

$$u_{min} \leq u(\mathbf{x}, \tau) \leq u_{max} , \tag{A.7a}$$

$$u_{min} \equiv \begin{cases} u^0_{\min,N} e^{-v\tau\sigma_{\max,N}} + \frac{q_{\min,N}}{\sigma_{\max,N}} (1 - e^{-v\tau\sigma_{\max,N}}) , & \sigma_{\max,N} \neq 0 \\ u^0_{\min,N} + v\tau q_{\min,N} , & \sigma_{\max,N} = 0 \end{cases} , \tag{A.7b}$$

$$u_{max} \equiv \begin{cases} u^0_{\max,N} e^{-v\tau\sigma_{\min,N}} + \frac{q_{\max,N}}{\sigma_{\min,N}} (1 - e^{-v\tau\sigma_{\min,N}}) , & \sigma_{\min,N} \neq 0 \\ u^0_{\max,N} + v\tau q_{\max,N} , & \sigma_{\min,N} = 0 \end{cases} , \tag{A.7c}$$

$$u^0_{\min,N} \equiv \min_{\mathbf{y} \in N(\mathbf{x},\tau)} u(\mathbf{y}, 0) , \quad u^0_{\max,N} \equiv \max_{\mathbf{y} \in N(\mathbf{x},\tau)} u(\mathbf{y}, 0) , \tag{A.7d}$$

$$\sigma_{\min,L} \equiv \min_{\mathbf{y} \in L(\mathbf{x},\tau)} \sigma(\mathbf{y}) , \quad \sigma_{\max,L} \equiv \max_{\mathbf{y} \in L(\mathbf{x},\tau)} \sigma(\mathbf{y}) , \tag{A.7e}$$

$$q_{\min,L} \equiv \min_{\mathbf{y} \in L(\mathbf{x},\tau)} q(\mathbf{y}) , \quad q_{\max,L} \equiv \max_{\mathbf{y} \in L(\mathbf{x},\tau)} q(\mathbf{y}) . \tag{A.7f}$$

Proof Because $\mathbf{x} - v\tau\boldsymbol{\Omega} \in N(\mathbf{x}, \tau)$,

$$u_0(\mathbf{x} - v\tau\boldsymbol{\Omega}) \geq u^0_{\min,N} , \quad u_0(\mathbf{x} - v\tau\boldsymbol{\Omega}) \leq u^0_{\max,N} .$$

Because $L(\mathbf{x},\tau) \subset N(\mathbf{x},\tau)$ (see Figure A.1), the following is true:

$$q_{\min,N} \leq q_{\min,L}\,, \quad q_{\max,N} \geq q_{\max,L}\,,$$

$$\sigma_{\min,N} \leq \sigma_{\min,L}\,, \quad \sigma_{\max,N} \geq \sigma_{\max,L}\,.$$

Applying these inequalities to Equation (A.5) proves Equation (A.7). ∎

The following theorem applies Corollary A.3.1 to derive an analytic discrete maximum principle for radiation transport.

Theorem A.3.2 (Analytic Discrete Maximum Principle) *If the time step size Δt satisfies the condition*

$$v\Delta t \leq \Delta x_{\min}\,, \quad \Delta x_{\min} \equiv \min_K \Delta x_K\,, \tag{A.8}$$

where Δx_K is the diameter of cell K, then Theorem A.3.1 gives the following analytic discrete maximum principle.

$$W_i^{\text{analytic},-} \leq U_i^{n+1} \leq W_i^{\text{analytic},+}\,, \tag{A.9a}$$

$$W_i^{\text{analytic},-} \equiv \begin{cases} U_{\min,i}^n e^{-v\Delta t\sigma_{\max,i}} + \frac{q_{\min,i}}{\sigma_{\max,i}}(1 - e^{-v\Delta t\sigma_{\max,i}})\,, & \sigma_{\max,i} \neq 0 \\ U_{\min,i}^n + v\Delta t q_{\min,i}\,, & \sigma_{\max,i} = 0 \end{cases}, \tag{A.9b}$$

$$W_i^{\text{analytic},+} \equiv \begin{cases} U_{\max,i}^n e^{-v\Delta t\sigma_{\min,i}} + \frac{q_{\max,i}}{\sigma_{\min,i}}(1 - e^{-v\Delta t\sigma_{\min,i}})\,, & \sigma_{\min,i} \neq 0 \\ U_{\max,i}^n + v\Delta t q_{\max,i}\,, & \sigma_{\min,i} = 0 \end{cases}, \tag{A.9c}$$

where $U_{\max,i}^n \equiv \max_{j\in\mathcal{I}(S_i)} U_j^n$, $\sigma_{\max,i} \equiv \max_{\mathbf{x}\in S_i}\sigma(\mathbf{x})$, and $q_{\max,i} \equiv \max_{\mathbf{x}\in S_i} q(\mathbf{x})$, with $U_{\min,i}^n$, $\sigma_{\min,i}$, and $q_{\min,i}$ defined similarly.

Proof Due to the CFL condition, Equation (A.8), the support of test function i is

a superset of the neighborhood $N(\mathbf{x}_i)$ defined by Equation (A.6): $N(\mathbf{x}_i) \subset S_i$. Thus for an arbitrary function of space $f(\mathbf{x})$,

$$\max_{\mathbf{x}\in S_i} f(\mathbf{x}) \geq \max_{\mathbf{x}\in N(\mathbf{x}_i)} f(\mathbf{x}) \,, \quad \min_{\mathbf{x}\in S_i} f(\mathbf{x}) \leq \min_{\mathbf{x}\in N(\mathbf{x}_i)} f(\mathbf{x}) \,,$$

and

$$\tilde{u}_{\max,S_i} \geq \tilde{u}_{\max,N} \,, \quad \tilde{u}_{\min,S_i} \leq \tilde{u}_{\min,N} \,.$$

Since $\tilde{u}$ is a convex combination of nodal solution values, the local extremum are obtained only at nodal values:

$$\tilde{u}_{\max,S_i} = U_{\max,i} \,, \quad \tilde{u}_{\min,S_i} = U_{\min,i} \,. \quad \blacksquare$$

The following corollary extends the analytic discrete maximum principle given in Theorem A.3.2 to the steady-state case and is given without proof, as it follows the same logic as Theorem A.3.2.

Corollary A.3.2 (Analytic Steady-State Discrete Maximum Principle) *If one uses a parameter s such that $s \leq \Delta x_{\min}$, where $\Delta x_{\min}$ is defined by Equation* (A.8)*, then the following analytic discrete maximum principle bounds apply to the steady-state problem:*

$$W_i^{\text{analytic},-} \leq U_i \leq W_i^{\text{analytic},+} \,, \tag{A.10a}$$

$$W_i^{\text{analytic},-} \equiv \begin{cases} U_{\min,i} e^{-s\sigma_{\max,i}} + \frac{q_{\min,i}}{\sigma_{\max,i}}(1 - e^{-s\sigma_{\max,i}}) \,, & \sigma_{\max,i} \neq 0 \\ U_{\min,i} + s q_{\min,i} \,, & \sigma_{\max,i} = 0 \end{cases} \,, \tag{A.10b}$$

$$W_i^{\text{analytic},+} \equiv \begin{cases} U_{\max,i} e^{-s\sigma_{\min,i}} + \frac{q_{\max,i}}{\sigma_{\min,i}}(1 - e^{-s\sigma_{\min,i}}) \,, & \sigma_{\min,i} \neq 0 \\ U_{\max,i} + s q_{\max,i} \,, & \sigma_{\min,i} = 0 \end{cases} \,. \tag{A.10c}$$

Remark In practice, one can approximate the maximum/minimum operations by

taking the maximum/minimum over quadrature points: e.g., $\max_{\mathbf{x} \in S_i} \approx \max_{\mathbf{x} \in Q(S_i)}$, where $Q(S_i)$ is the set of quadrature points in S_i.

APPENDIX B

DERIVATION OF THE ENTROPY FLUX FOR THE SHALLOW WATER EQUATIONS

Recall from Section 2.1.2 the definition of the shallow water equations:

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot \mathbf{F} = \mathbf{s}(\mathbf{u}) \,,$$
$$\mathbf{u} = \begin{bmatrix} h \\ \mathbf{q} \end{bmatrix} \,, \quad \mathbf{F} = \begin{bmatrix} \mathbf{q} \\ \frac{\mathbf{q}\otimes\mathbf{q}}{h} + \frac{1}{2}gh^2\mathbf{I} \end{bmatrix} \,, \quad \mathbf{s}(\mathbf{u}) = \begin{bmatrix} 0 \\ -gh\nabla b \end{bmatrix} \,. \tag{B.1}$$

In this section, the following notation will be used to denote the fluxes for each component:

$$\mathbf{F} = \begin{bmatrix} \mathbf{f}^h(\mathbf{u}) \\ \mathbf{f}^{\mathbf{q}}(\mathbf{u}) \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ \frac{\mathbf{q}\otimes\mathbf{q}}{h} + \frac{1}{2}gh^2\mathbf{I} \end{bmatrix} \,,$$

where $\mathbf{f}^{\mathbf{q}}(\mathbf{u})$ is a vector of the momentum component fluxes:

$$\mathbf{f}^{\mathbf{q}}(\mathbf{u}) = \begin{bmatrix} \mathbf{f}^{q_x}(\mathbf{u}) \\ \mathbf{f}^{q_y}(\mathbf{u}) \end{bmatrix} = \begin{bmatrix} \frac{q_x^2}{h} + \frac{1}{2}gh^2 & \frac{q_x q_y}{h} \\ \frac{q_x q_y}{h} & \frac{q_y^2}{h} + \frac{1}{2}gh^2 \end{bmatrix} \,.$$

Here the dependence of the flux functions on $\mathbf{u}$ will be dropped for brevity.

Recall from Equation (2.111) the definition of the entropy function for the SWE:

$$\eta(h, \mathbf{q}, b) = \frac{1}{2}\frac{\mathbf{q}\cdot\mathbf{q}}{h} + \frac{1}{2}gh\left(h + b\right) \,.$$

The objective here is to derive an entropy equation, which gives the time rate of change of entropy $\partial_t \eta$. To yield such an equation, one can take advantage of the

derivative chain rule:

$$\partial_t \eta = \partial_h \eta \, \partial_t h + \partial_{\mathbf{q}} \eta \cdot \partial_t \mathbf{q} \,, \tag{B.2}$$

where the partial derivatives of the entropy function with respect to each solution variable are the following:

$$\partial_h \eta = -\frac{1}{2} \frac{\mathbf{q} \cdot \mathbf{q}}{h^2} + gh + \frac{1}{2} gb \,, \tag{B.3a}$$

$$\partial_{\mathbf{q}} \eta = \begin{bmatrix} \partial_{q_x} \eta \\ \partial_{q_y} \eta \end{bmatrix} = \begin{bmatrix} \frac{q_x}{h} \\ \frac{q_y}{h} \end{bmatrix} = \frac{\mathbf{q}}{h} \,. \tag{B.3b}$$

Remark The term $\partial_b \eta \partial_t b$ does not appear in Equation (B.2) due to the assumption that b is not a function of time.

To arrive at an entropy equality, each conservation equation in the system is multiplied by the respective derivative of the entropy function and then summed:

$$\partial_h \eta \, \partial_t h + \partial_{\mathbf{q}} \eta \cdot \partial_t \mathbf{q} + \partial_h \eta \, \nabla \cdot \mathbf{f}^h + \sum_{d=1}^{N_{\text{dim}}} \partial_{q_d} \eta \nabla \cdot \mathbf{f}^{q_d} + \partial_{\mathbf{q}} \eta \cdot gh \nabla b = 0 \,. \tag{B.4}$$

Using Equation (B.2), the temporal derivatives can be expressed as a partial derivative of entropy:

$$\partial_t \eta + \partial_h \eta \, \nabla \cdot \mathbf{f}^h + \sum_{d=1}^{N_{\text{dim}}} \partial_{q_d} \eta \nabla \cdot \mathbf{f}^{q_d} + \partial_{\mathbf{q}} \eta \cdot gh \nabla b = 0 \,. \tag{B.5}$$

An *entropy flux* $\mathbf{f}^\eta$, is defined such that its divergence matches the spatial derivative terms in Equation (B.5):

$$\partial_t \eta + \nabla \cdot \mathbf{f}^\eta = 0 \,. \tag{B.6}$$

Comparing Equations (B.5) and (B.6) gives the definition of the divergence of the

entropy flux:

$$\nabla \cdot \mathbf{f}^{\eta} = \partial_h \eta \, \nabla \cdot \mathbf{f}^{h} + \sum_{d=1}^{N_{\text{dim}}} \partial_{q_d} \eta \nabla \cdot \mathbf{f}^{q_d} + \partial_{\mathbf{q}} \eta \cdot g h \nabla b \tag{B.7}$$

The divergences of the component fluxes are the following:

$$\nabla \cdot \mathbf{f}^{h} = \nabla \cdot \mathbf{q} \,, \tag{B.8a}$$

$$\nabla \cdot \mathbf{f}^{q_d} = -\frac{q_d}{h^2} \mathbf{q} \cdot \nabla h + \frac{\mathbf{q}}{h} \cdot \nabla q_d + \frac{q_d}{h} \nabla \cdot \mathbf{q} + g h \partial_{x_d} h \,, \tag{B.8b}$$

and the momentum sum term simplifies as follows:

$$\sum_{d}^{N_{\text{dim}}} \frac{q_d}{h} \nabla \cdot \mathbf{f}^{q_d} = \sum_{d}^{N_{\text{dim}}} -\frac{q_d^2}{h^3} \mathbf{q} \cdot \nabla h + \sum_{d}^{N_{\text{dim}}} \frac{q_d \mathbf{q}}{h^2} \cdot \nabla q_d + \sum_{d}^{N_{\text{dim}}} \frac{q_d^2}{h^2} \nabla \cdot \mathbf{q} + \sum_{d}^{N_{\text{dim}}} g q_d \partial_{x_d} h \,, \tag{B.9}$$

$$\sum_{d}^{N_{\text{dim}}} \frac{q_d}{h} \nabla \cdot \mathbf{f}^{q_d} = \left(-\frac{(\mathbf{q} \cdot \mathbf{q}) \mathbf{q}}{h^3} \right) \cdot \nabla h + \sum_{d}^{N_{\text{dim}}} \left(\frac{q_d \mathbf{q}}{h^2} \right) \cdot \nabla q_d + \left(\frac{\mathbf{q} \cdot \mathbf{q}}{h^2} \right) \nabla \cdot \mathbf{q} + (g \mathbf{q}) \cdot \nabla h \,. \tag{B.10}$$

Substituting these definitions into Equation (B.7) gives

$$\begin{aligned} \nabla \cdot \mathbf{f}^{\eta} = \left(\frac{1}{2} \frac{\mathbf{q} \cdot \mathbf{q}}{h^2} + g h + \frac{1}{2} g b \right) \nabla \cdot \mathbf{q} + \sum_{d}^{N_{\text{dim}}} \left(\frac{q_d \mathbf{q}}{h^2} \right) \cdot \nabla q_d \\ + \left(g \mathbf{q} - \frac{(\mathbf{q} \cdot \mathbf{q}) \mathbf{q}}{h^3} \right) \cdot \nabla h + g \mathbf{q} \cdot \nabla b \,, \end{aligned} \tag{B.11}$$

which can be rewritten as

$$\nabla \cdot \mathbf{f}^{\eta} = \sum_{d}^{N_{\text{dim}}} \left(\left(\frac{1}{2} \frac{\mathbf{q} \cdot \mathbf{q}}{h^2} + gh + \frac{1}{2} gb \right) \hat{\mathbf{e}}_d + \frac{q_d \mathbf{q}}{h^2} \right) \cdot \nabla q_d + \left(g\mathbf{q} - \frac{(\mathbf{q} \cdot \mathbf{q})\mathbf{q}}{h^3} \right) \cdot \nabla h + g\mathbf{q} \cdot \nabla b \,. \quad \text{(B.12)}$$

Assuming the entropy flux to be a function of h, $\mathbf{q}$, and b, i.e., the entropy flux is $\mathbf{f}^{\eta}(h, \mathbf{q}, b)$, and applying chain rule for its divergence yields

$$\nabla \cdot \mathbf{f}^{\eta} = \partial_h \mathbf{f}^{\eta} \cdot \nabla h + \sum_{d=1}^{N_{\text{dim}}} \partial_{q_d} \mathbf{f}^{\eta} \cdot \nabla q_d + \partial_b \mathbf{f}^{\eta} \cdot \nabla b \,. \quad \text{(B.13)}$$

Matching the coefficients of ∇h, ∇q_d, and ∇b between this equation and Equation (B.12) gives the definitions of the partial derivatives of the entropy flux:

$$\partial_h \mathbf{f}^{\eta} = g\mathbf{q} - \frac{(\mathbf{q} \cdot \mathbf{q})\,\mathbf{q}}{h^3} \,, \quad \text{(B.14a)}$$

$$\partial_{q_d} \mathbf{f}^{\eta} = \left(\frac{1}{2} \frac{\mathbf{q} \cdot \mathbf{q}}{h^2} + gh + \frac{1}{2} gb \right) \hat{\mathbf{e}}_d + \frac{q_d \mathbf{q}}{h^2} \,, \quad \text{(B.14b)}$$

$$\partial_b \mathbf{f}^{\eta} = g\mathbf{q} \,. \quad \text{(B.14c)}$$

Integrating the equation for $\partial_h \mathbf{f}^{\eta}$ gives

$$\mathbf{f}^{\eta} = gh\mathbf{q} + \frac{1}{2} \frac{(\mathbf{q} \cdot \mathbf{q})\,\mathbf{q}}{h^2} + \mathbf{c}_1(\mathbf{q}, b) \,, \quad \text{(B.15)}$$

where $\mathbf{c}_1(\mathbf{q}, b)$ is a constant with respect to h. Taking the partial derivative of this

expression with respect to q_x gives

$$\partial_{q_x}\mathbf{f}^{\eta} = gh\hat{\mathbf{e}}_x + \frac{1}{2h^2}\begin{bmatrix} 3q_x^2 + q_y^2 \\ 2q_xq_y \end{bmatrix} + \frac{\partial \mathbf{c}_1}{\partial q_x}\,, \tag{B.16}$$

which when compared to Equation (B.14b) gives

$$\frac{\partial \mathbf{c}_1}{\partial q_x} = \frac{1}{2}gb\hat{\mathbf{e}}_x\,. \tag{B.17}$$

Integrating gives

$$\mathbf{c}_1(\mathbf{q}, b) = \frac{1}{2}gbq_x\hat{\mathbf{e}}_x + \mathbf{c}_2(q_y, b)\,. \tag{B.18}$$

Making this substitution into Equation (B.15) and taking the partial derivative with respect to q_y gives

$$\partial_{q_y}\mathbf{f}^{\eta} = gh\hat{\mathbf{e}}_y + \frac{1}{2h^2}\begin{bmatrix} 2q_xq_y \\ q_x^2 + 3q_y^2 \end{bmatrix} + \frac{\partial \mathbf{c}_2}{\partial q_y}\,. \tag{B.19}$$

Comparing this with Equation (B.14b) gives

$$\frac{\partial \mathbf{c}_2}{\partial q_y} = \frac{1}{2}gb\hat{\mathbf{e}}_y\,. \tag{B.20}$$

Integrating gives

$$\mathbf{c}_2(q_y, b) = \frac{1}{2}gbq_y\hat{\mathbf{e}}_y + \mathbf{c}_3(b)\,. \tag{B.21}$$

Making this substitution into Equation (B.15) and taking the partial derivative with respect to b gives

$$\frac{d\mathbf{c}_3}{db} = \frac{1}{2}g\mathbf{q}\,, \tag{B.22}$$

which gives

$$\mathbf{c}_3(b) = \frac{1}{2} g b \mathbf{q} + \mathbf{c}_4 \,, \tag{B.23}$$

where $\mathbf{c}_4$ is set to zero. Thus the final equation for the entropy flux is

$$\mathbf{f}^{\eta}(\mathbf{u}, b) = g(h+b)\mathbf{q} + \frac{1}{2}\frac{(\mathbf{q}\cdot\mathbf{q})\,\mathbf{q}}{h^2} \,. \tag{B.24}$$

APPENDIX C

BOUNDARY CONDITIONS FOR THE SHALLOW WATER EQUATIONS

C.1 Characteristic Boundary Conditions for the Shallow Water Equations

In 1-D, the boundary integrals in Equations (2.56) and (2.58) reduce to differences between the right and left boundaries:

$$\int_{\partial\mathcal{D}} \varphi_i^h \tilde{\mathbf{q}} \cdot \mathbf{n}\, dA = (hu)_R - (hu)_L \,, \tag{C.1a}$$

$$\int_{\partial\mathcal{D}} \varphi_i^{q_x} \left(\frac{\tilde{q_x}}{h} \tilde{\mathbf{q}} + \frac{1}{2} g \tilde{h}^2 \hat{\mathbf{e}}_x \right) \cdot \mathbf{n}\, dA = \left(hu^2 + \frac{1}{2} g h^2 \right)_R - \left(hu^2 + \frac{1}{2} g h^2 \right)_L \,, \tag{C.1b}$$

where R and L denote right and left boundaries, respectively.

In 1-D, there are 2 characteristics:

$$du - 2da = 0 \,, \tag{C.2a}$$

$$du + 2da = 0 \,, \tag{C.2b}$$

giving the Riemann invariants $u-2a$ and $u+2a$, which correspond to the eigenvalues $\lambda_1 = u - a$ and $\lambda_2 = u + a$, respectively. Integrating the characteristics from the boundary position x^{BC} to an interior position x^{in} gives

$$u^{\mathrm{in}} - u^{\mathrm{BC}} = 2\left(a^{\mathrm{in}} - a^{\mathrm{BC}}\right) \,, \tag{C.3a}$$

$$u^{\mathrm{in}} - u^{\mathrm{BC}} = 2\left(a^{\mathrm{BC}} - a^{\mathrm{in}}\right) \,, \tag{C.3b}$$

associated with λ_1 and λ_2, respectively. At each boundary, one must determine whether the waves associated with each eigenvalue are coming into the domain or going out of the domain; this determines how many external boundary conditions must be applied at each boundary. The Froude number $\text{Fr} \equiv \frac{|u|}{a}$, along with the sign of the velocity u, determines the sign of each of the 2 eigenvalues, which is summarized in Table C.1.

Table C.1: Signs of Eigenvalues for Different Cases

Froude Sign	$u < 0$	$u \geq 0$
$\text{Fr} < 1$	$\lambda_1 \leq 0$	$\lambda_1 < 0$
	$\lambda_2 > 0$	$\lambda_2 \geq 0$
$\text{Fr} \geq 1$	$\lambda_1 \leq 0$	$\lambda_1 \geq 0$
	$\lambda_2 \leq 0$	$\lambda_2 \geq 0$

If $\lambda_i n_x \leq 0$, then an external boundary condition must be applied for the i-wave; otherwise, internal information is used for that boundary condition.

For subcritical flow, i.e., $|u| \leq a$, the signs of each eigenvalue are $\lambda_1 \leq 0$ and $\lambda_2 \geq 0$. For supercritical flow, the signs are $\lambda_1 \geq 0$ and $\lambda_2 \geq 0$ for $u < 0$, and for $u \geq 0$, the signs are $\lambda_1 \leq 0$ and $\lambda_2 \leq 0$. Thus for supercritical flow, inlets, i.e., boundaries for which $un_x < 0$, require 2 external boundary conditions, whereas outlets use 2 internal boundary conditions. Tables C.2 and C.3 summarize the application of open and wall boundary conditions, respectively.

Table C.2: Summary of Open Boundary Conditions for the 1-D Shallow Water Equations

Case	**Equations**
Subcritical Left Boundary (Inlet or Outlet)	Provide a^{BC} $u^{\mathrm{BC}} = u^{\mathrm{in}} + 2\left(a^{\mathrm{BC}} - a^{\mathrm{in}}\right)$
Subcritical Right Boundary (Inlet or Outlet)	Provide a^{BC} $u^{\mathrm{BC}} = u^{\mathrm{in}} + 2\left(a^{\mathrm{in}} - a^{\mathrm{BC}}\right)$
Supercritical Inlet	Provide a^{BC} Provide u^{BC}
Supercritical Outlet	$a^{\mathrm{BC}} = a^{\mathrm{in}}$ $u^{\mathrm{BC}} = u^{\mathrm{in}}$

C.2 Wall Boundary Conditions for the Shallow Water Equations

For wall boundary conditions, the normal component of velocity is set to zero:

$$\mathbf{v} \cdot \mathbf{n} = 0\,, \tag{C.4}$$

and therefore the boundary fluxes reduce to the following:

$$\int_{\partial\mathcal{D}} \varphi_i^h \tilde{\mathbf{q}} \cdot \mathbf{n}\, dA = 0\,, \tag{C.5a}$$

$$\int_{\partial\mathcal{D}} \varphi_i^{q_d} \left(\frac{\tilde{q}_d}{h}\tilde{\mathbf{q}} + \frac{1}{2}g\tilde{h}^2\hat{\mathbf{e}}_d\right) \cdot \mathbf{n}\, dA = \int_{\partial\mathcal{D}} \varphi_i^{q_d} \left(\frac{1}{2}g\tilde{h}^2\hat{\mathbf{e}}_d\right) \cdot \mathbf{n}\, dA\,. \tag{C.5b}$$

Table C.3: Summary of Wall Boundary Conditions for the 1-D Shallow Water Equations

Case	**Equations**
Subcritical Left Boundary (Inlet or Outlet)	Provide $u^{\mathrm{BC}} = 0$ $a^{\mathrm{BC}} = a^{\mathrm{in}} + \frac{1}{2}\left(u^{\mathrm{BC}} - u^{\mathrm{in}}\right)$
Subcritical Right Boundary (Inlet or Outlet)	Provide $u^{\mathrm{BC}} = 0$ $a^{\mathrm{BC}} = a^{\mathrm{in}} + \frac{1}{2}\left(u^{\mathrm{in}} - u^{\mathrm{BC}}\right)$
Supercritical Inlet	Provide u^{BC} Provide a^{BC}
Supercritical Outlet	$u^{\mathrm{BC}} = u^{\mathrm{in}}$ $a^{\mathrm{BC}} = a^{\mathrm{in}}$

APPENDIX D

DERIVATION OF THE MAX WAVE SPEED FOR THE SHALLOW WATER EQUATIONS

The maximum wave speed in a multidimensional problem is equal to the maximum wave speed of the one-dimensional problem in the direction given by the normal vector $\mathbf{n}$:

$$\lambda^{\max}(\mathbf{n}, [h, \mathbf{q}]_L^{\mathrm{T}}, [h, \mathbf{q}]_R^{\mathrm{T}}) = \lambda^{\max}([h, q_n]_L^{\mathrm{T}}, [h, q_n]_R^{\mathrm{T}}), \tag{D.1}$$

where $q_n \equiv \mathbf{q} \cdot \mathbf{n}$ denotes the component of $\mathbf{q}$ along $\mathbf{n}$. The maximum wave speed in the one-dimensional Riemann problem is the maximum of the absolute values of the left-most and right-most wave speeds:

$$\lambda^{\max}(\mathbf{u}_L, \mathbf{u}_R) = \max(|\lambda_1^-(\mathbf{u}_L, \mathbf{u}_R)|, |\lambda_2^+(\mathbf{u}_L, \mathbf{u}_R)|), \tag{D.2}$$

where $\mathbf{u}_K \equiv [h, q_n]_K^{\mathrm{T}}$, and the "+" and "-" allow for the differentiation of the head and tail speeds in the case of a rarefaction.

For the 1-D shallow water equations, the Riemann problem divides the x-t plane into 3 sectors, separated by 2 waves, which each may be either a shock or rarefaction. The left sector shall be denoted with "L", the middle with "*", and the right with "R". The left-most and right-most wave speeds are

$$\lambda_1^-(\mathbf{u}_L, \mathbf{u}_R) = u_L - a_L \left(1 + \left(\frac{(h_* - h_L)(h_* + 2h_L)}{2h_L^2}\right)_+\right)^{\frac{1}{2}}, \tag{D.3}$$

$$\lambda_2^+(\mathbf{u}_L, \mathbf{u}_R) = u_R + a_R \left(1 + \left(\frac{(h_* - h_R)(h_* + 2h_R)}{2h_R^2}\right)_+\right)^{\frac{1}{2}}, \tag{D.4}$$

where $(z)_+ = \max(z, 0)$. These definitions are completely general in that they apply to both shocks and rarefactions. In the case of a rarefaction for the left side, $h_L \leq h_*$, and similarly for the right side, $h_R \leq h_*$. The wave speed of the head of the rarefaction in each case is

$$\lambda_1^-(\mathbf{u}_L, \mathbf{u}_R) = u_L - a_L \,, \tag{D.5}$$

$$\lambda_2^+(\mathbf{u}_L, \mathbf{u}_R) = u_R + a_R \,. \tag{D.6}$$

Otherwise (when $h_L > h_*$ or $h_R > h_*$), the wave is a shock, and the shock speed in each case is

$$\lambda_1^-(\mathbf{u}_L, \mathbf{u}_R) = u_L - a_L \left(1 + \left(\frac{(h_* - h_L)(h_* + 2h_L)}{2h_L^2}\right)\right)^{\frac{1}{2}} , \tag{D.7}$$

$$\lambda_2^+(\mathbf{u}_L, \mathbf{u}_R) = u_R + a_R \left(1 + \left(\frac{(h_* - h_R)(h_* + 2h_R)}{2h_R^2}\right)\right)^{\frac{1}{2}} . \tag{D.8}$$

Combining these equations with the rarefaction speeds gives the general definitions of the left-most and right-most wave speeds.

The height in the star (*) region is the solution of the nonlinear equation

$$\phi(h) \equiv \mathcal{W}_L(h, \mathbf{u}_L) + \mathcal{W}_R(h, \mathbf{u}_R) + u_R - u_L = 0 \,, \tag{D.9}$$

where $\mathcal{W}_L(h, \mathbf{u}_L)$ and $\mathcal{W}_R(h, \mathbf{u}_R)$ are the left and right wave strengths, each corresponding to either a shock or rarefaction. The derivation of this equation is given in Section D.3.

In the case of a shock,

$$\mathcal{W}_K(h, \mathbf{u}_K) = \mathcal{W}_K^{\text{shock}} = (h - h_K)\sqrt{\frac{1}{2} g \frac{h + h_K}{h h_K}} \,, \tag{D.10}$$

while in the case of a rarefaction,

$$\mathcal{W}_K(h, \mathbf{u}_K) = \mathcal{W}_K^{\text{rarefaction}} = 2(a - a_K)\,. \tag{D.11}$$

These wave strength functions are derived in the following sections.

D.1 Shock Wave

This derivation will correspond to the left wave; the right wave derivation proceeds similarly.

In the case of a shock, the discontinuous wave front moves with speed S_L, separating the left solution $\mathbf{u}_L$ and the right solution $\mathbf{u}_*$. Transforming to a reference frame moving with the shock, the reference frame velocities are

$$\hat{u}_L = u_L - S_L\,, \tag{D.12}$$

$$\hat{u}_* = u_* - S_L\,. \tag{D.13}$$

Applying the Rankine-Hugoniot condition for both the continuity equation and momentum equation gives

$$h_L \hat{u}_L = h_* \hat{u}_*\,, \tag{D.14}$$

$$h_L \hat{u}_L^2 + p_L = h_* \hat{u}_*^2 + p_*\,, \tag{D.15}$$

where $p = \frac{1}{2} g h^2$. Defining the reference discharge as

$$\hat{q}_L = h_L \hat{u}_L = h_* \hat{u}_* \tag{D.16}$$

and substituting into Equation (D.15) gives

$$\hat{q}_L = \frac{p_L - p_*}{\hat{u}_* - \hat{u}_L}\,, \tag{D.17}$$

which when combined with Equations (D.12) and (D.13) gives

$$\hat{q}_L = \sqrt{\frac{h_* h_L (p_L - p_*)}{h_L - h_*}}\,. \tag{D.18}$$

Remark Combining Equations (D.12), (D.16), and (D.18) and performing some algebra gives the expression for the shock speed S_L given by Equation (D.7).

Using $\hat{u}_* - \hat{u}_L = u_* - u_L$ with Equation (D.17) gives

$$\hat{q}_L = \frac{p_L - p_*}{u_* - u_L} \tag{D.19}$$

and combining with Equation (D.18) gives, after a bit of algebra,

$$u_* = u_L - \mathcal{W}_L^{\text{shock}}(h_*, \mathbf{u}_L)\,, \tag{D.20}$$

where

$$\mathcal{W}_L^{\text{shock}}(h, \mathbf{u}_L) = (h - h_L)\sqrt{\frac{1}{2} g \frac{h + h_L}{h h_L}}\,. \tag{D.21}$$

Performing a similar analysis for the right wave gives

$$u_* = u_R + \mathcal{W}_R^{\text{shock}}(h_*, \mathbf{u}_R)\,, \tag{D.22}$$

where

$$\mathcal{W}_R^{\text{shock}}(h, \mathbf{u}_R) = (h - h_R)\sqrt{\frac{1}{2} g \frac{h + h_R}{h h_R}}\,. \tag{D.23}$$

D.2 Rarefaction Wave

$$u_* = u_L - \mathcal{W}_L^{\text{rarefaction}}(h_*, \mathbf{u}_L)\,, \tag{D.24}$$

where

$$\mathcal{W}_L^{\text{rarefaction}}(h, \mathbf{u}_L) = 2(a - a_L) \tag{D.25}$$

Performing a similar analysis for the right wave gives

$$u_* = u_R + \mathcal{W}_R^{\text{rarefaction}}(h_*, \mathbf{u}_R)\,, \tag{D.26}$$

where

$$\mathcal{W}_R^{\text{rarefaction}}(h, \mathbf{u}_R) = 2(a - a_R) \tag{D.27}$$

D.3 Obtaining the Solution in the Star Region

Combining Equation (D.20) or (D.24) with (D.22) or (D.26) by eliminating u gives the nonlinear equation to solve for h_*, where the LHS is defined to be $\phi(h)$ this is Equation (D.9).

Then adding either Equation (D.20) or (D.24) with (D.22) or (D.26) gives the equation for u_*:

$$u_* = \frac{1}{2}\left(u_L + u_R + \mathcal{W}_R(h_*, \mathbf{u}_R) - \mathcal{W}_L(h_*, \mathbf{u}_L)\right)\,. \tag{D.28}$$

D.4 Fast Estimate of Maximum Wave Speed

The fast algorithm given in this section attempts to ease the computational burden of computing the height in the star region h_* (which requires a nonlinear solve that may require several iterations), which is needed in the computation of the wave

speeds given by Equations (D.3) (D.4).

The first condition needed by this algorithm is that the objective function $\phi(h)$ defined in Equation (D.9) be monotone increasing. This is given in the following theorem.

Theorem D.4.1 (Monotonicity of the Objective Function) *The objective function $\phi(h)$ defined in Equation* (D.9) *is monotone increasing: $\phi'(h) \geq 0$.*

Proof It is sufficient to prove that each wave strength function $\mathcal{W}_K^{\text{rarefaction}}$ and $\mathcal{W}_K^{\text{shock}}$ are monotone increasing with respect to h. This is trivial to prove for rarefaction waves:

$$\frac{\partial \mathcal{W}_K^{\text{rarefaction}}}{\partial h} = \sqrt{\frac{g}{h}} \geq 0\,. \tag{D.29}$$

For shock waves, the proof is more complicated. To simplify algebra, the following definition is made:

$$\alpha \equiv \sqrt{\frac{1}{2} g \left(\frac{1}{h} + \frac{1}{h_K} \right)}\,, \tag{D.30}$$

making the expression for wave strength the following:

$$\mathcal{W}_K^{\text{shock}} = (h - h_K)\,\alpha\,. \tag{D.31}$$

Taking the derivative gives

$$\begin{aligned} \frac{\partial \mathcal{W}_K^{\text{shock}}}{\partial h} &= \alpha + (h - h_K)\,\frac{\partial \alpha}{\partial h}\,, \\ \frac{\partial \mathcal{W}_K^{\text{shock}}}{\partial h} &= \alpha - \frac{1}{4} g \frac{h - h_K}{h^2} \frac{1}{\alpha}\,. \end{aligned}$$

This proof now proceeds by assumption that $\frac{\partial \mathcal{W}_K^{\text{shock}}}{\partial h} \geq 0$:

$$\begin{aligned} \alpha - \frac{1}{4} g \frac{h - h_K}{h^2} \frac{1}{\alpha} &\geq 0\,, \\ \alpha^2 &\geq \frac{1}{4} g \frac{h - h_K}{h^2}\,. \end{aligned}$$

Note that the last equation assumes that $\alpha \geq 0$.

$$\begin{aligned} \frac{1}{2} g \frac{h + h_K}{h h_K} &\geq \frac{1}{4} g \frac{h - h_K}{h^2}\,, \\ h^3 + h_K h^2 &\geq \frac{1}{2}\left(h_K h^2 - h_K^2 h\right)\,, \\ h^3 + \frac{1}{2} h_K h^2 + \frac{1}{2} h_K^2 h &\geq 0\,. \end{aligned}$$

This last statement is proved using the entropy condition. Thus the assumption $\frac{\partial \mathcal{W}_K^{\text{shock}}}{\partial h} \geq 0$ is verified, and the proof is complete. ∎

$$\lambda^{\max,k} = \max\left((\hat{\lambda}_2^{(k)})_+, (\check{\lambda}_1^{(k)})_-\right)\,, \tag{D.32}$$

$$\lambda_{\min}^{(k)} = \left(\max\left((\check{\lambda}_2^{(k)})_+, (\hat{\lambda}_1^{(k)})_-\right)\right)_+\,, \tag{D.33}$$

where $z_+ \equiv \max(z, 0)$, $z_- \equiv \max(-z, 0)$, and the bounds on the individual wave speeds are given by the following equations:

$$\check{\lambda}_1^{(k)} = u_L - a_L \left(1 + \left(\frac{(\hat{h}^{(k)} - h_L)(\hat{h}^{(k)} + 2h_L)}{2h_L^2}\right)_+\right)^{\frac{1}{2}}\,, \tag{D.34}$$

$$\hat{\lambda}_1^{(k)} = u_L - a_L \left(1 + \left(\frac{(\check{h}^{(k)} - h_L)(\check{h}^{(k)} + 2h_L)}{2h_L^2}\right)_+\right)^{\frac{1}{2}}\,, \tag{D.35}$$

$$\check{\lambda}_2^{(k)} = u_R + a_R \left(1 + \left(\frac{(\check{h}^{(k)} - h_R)(\check{h}^{(k)} + 2h_R)}{2h_R^2}\right)_+\right)^{\frac{1}{2}}\,, \tag{D.36}$$

Algorithm 2 Initialization

$h_{\min} \leftarrow \min(h_L, h_R)$
$h_{\max} \leftarrow \max(h_L, h_R)$
if $\phi(h_{\min}) \geq 0$ **then** ▷ Both waves are rarefactions
  Compute $\lambda^{\max}$ using Equations (D.2), (D.5), and (D.6)
  return
end if
if $\phi(h_{\max}) = 0$ **then** ▷ h_* is already known to be $h_{\max}$
  $h_* \leftarrow h_{\max}$
  Compute $\lambda^{\max}$ using Equations (D.2), (D.3), and (D.4)
  return
else if $\phi(h_{\max}) < 0$ **then** ▷ Both waves are shocks
  $\hat{h}^{(0)} \leftarrow \tilde{h}_*$
  $\check{h}^{(0)} \leftarrow \max\left(h_{\max}, \hat{h}^{(0)} - \frac{\phi(\hat{h}^{(0)})}{\phi'(\hat{h}^{(0)})}\right)$
  Call Algorithm 3 with $(\check{h}^{(0)}, \hat{h}^{(0)})$
  return
else ▷ One wave is rarefaction, one wave is shock
  $\hat{h}^{(0)} \leftarrow \min\left(h_{\max}, \tilde{h}_*\right)$
  $\check{h}^{(0)} \leftarrow \max\left(h_{\min}, \hat{h}^{(0)} - \frac{\phi(\hat{h}^{(0)})}{\phi'(\hat{h}^{(0)})}\right)$
  Call Algorithm 3 with $(\check{h}^{(0)}, \hat{h}^{(0)})$
  return
end if

$$\hat{\lambda}_2^{(k)} = u_R + a_R \left(1 + \left(\frac{(\hat{h}^{(k)} - h_R)(\hat{h}^{(k)} + 2h_R)}{2h_R^2}\right)_+\right)^{\frac{1}{2}}, \tag{D.37}$$

The interpolant functions h_d and h_u are given by the following equations:

$$h_d(h_1, h_2) = h_1 - \frac{2\phi(h_1)}{\phi'(h_1) + \sqrt{\phi'(h_1)^2 - 4\phi(h_1)\phi[h_1, h_1, h_2]}}, \tag{D.38}$$

$$h_u(h_1, h_2) = h_2 - \frac{2\phi(h_2)}{\phi'(h_2) + \sqrt{\phi'(h_2)^2 - 4\phi(h_2)\phi[h_1, h_2, h_2]}}, \tag{D.39}$$

Algorithm 3 Computation of $\lambda^{\max}$

Input: $(\check{h}^{(0)}, \hat{h}^{(0)}, \epsilon)$
loop
  Compute $\lambda^{\max,k}$ using Equation (D.32)
  Compute $\lambda_{\min}^{(k)}$ using Equation (D.33)
  if $\lambda_{\min}^{(0)} > 0$ **then**
    if $\frac{\lambda^{\max,k}}{\lambda_{\min}^{(k)}} - 1 \leq \epsilon$ **then**
      return
    end if
  end if
  if $\phi(\check{h}^{(k)}) > 0$ or $\phi(\hat{h}^{(k)}) < 0$ **then**
    return
  end if
  $\check{h}^{(k+1)} \leftarrow h_d(\check{h}^{(k)}, \hat{h}^{(k)})$
  $\hat{h}^{(k+1)} \leftarrow h_u(\check{h}^{(k)}, \hat{h}^{(k)})$
end loop
$\lambda^{\max} \leftarrow \lambda^{\max,k}$

where $\phi[x, y, z]$ denotes divided differences:

$$\phi[x, y, z] = \frac{\frac{1}{x-y}\left(\phi(x) - \phi(y)\right) - \frac{1}{y-z}\left(\phi(y) - \phi(z)\right)}{x - z}\,. \tag{D.40}$$