

A ROBUST LAGRANGIAN FRAMEWORK FOR COMPRESSIBLE FLOW &
HYPERELASTICITY

A Thesis
by
MADISON SPENCER SHERIDAN

Submitted to the Graduate and Professional School of
Texas A&M University
in partial fulfillment of the requirements for the degree of
DOCTOR OF PHILOSOPHY

Chair of Committee, Jean-Luc Guermond
Committee Members, Bojan Popov
Matthias Maier
Jim Morel
Head of Department, Peter Howard

December 2025

Major Subject: Mathematics

Copyright 2025 Madison Spencer Sheridan

ABSTRACT

Accurate and robust simulation of compressible flows in a Lagrangian frame presents significant numerical challenges, particularly in the presence of shocks, material interfaces, and high impact problems. These difficulties necessitate the development of numerical schemes that not only achieve high-order accuracy but also preserve key physical invariants of the governing equations.

This thesis will focus on the development of a robust first order invariant-domain preserving method for Lagrangian cell-centered hydrodynamics with an arbitrary equation of state. The scheme enforces fundamental thermodynamic bounds—positivity of density, a minimum principle for specific internal energy, and local mass conservation. The fundamental methodology to developing our first order method relies on the approximation of the maximum wave speed of a localized Riemann problem. We further ensure compatibility between mesh motion and thermodynamic updates through a velocity reconstruction algorithm that is consistent with the geometric conservation law.

Building on this robust baseline, we design a limiting strategy that constrains an existing high order approximation to the invariant domain, thereby transferring the low order robustness to a high order accurate scheme. The limiting process is modular, requiring only minimal, localized changes to an existing high order code and leaving core operators unchanged.

We extend the method to nonlinear hyperelastic solids by replacing fluid pressure with an energy-based stress, introducing a shear-energy variable dependent on the mesh deformation. We explore both isotropic and anisotropic closure models for the new shear energy. With only minor algorithmic changes, the robust low-order scheme preserves its core properties and provides a unified Lagrangian framework for fluids and solids.

DEDICATION

To my loving wife, Jessica, who has supported me at every phase of this journey. To my beautiful daughter, Sofia Aurélia da Silva. May you always be curious, courageous, and confident.

ACKNOWLEDGMENTS

I would like to thank my advisor, Dr. Jean-Luc Guermond, as well as my committee members Dr. Bojan Popov, Dr. Matthias Maier, and Dr. Jim Morel for their enthusiastic support throughout this toilsome journey. I certainly would not be able to complete this degree without their steady guidance. I would also like to thank Texas A&M University for allowing me the opportunity to pursue this research in such a supportive and inspiring environment.

I am grateful to my colleagues Dr. Eric Tovar, Dr. Bennett Clayton, and soon-to-be Dr. Seth Gerberding for their help, whether debugging code or serving as a sounding board for talks and ideas.

Above all, I would like to thank my wife for her admirable empathy, enduring with me through this period of my life. I do not think I could have completed this task without her.

CONTRIBUTORS AND FUNDING SOURCES

Contributors

This work was supported by a thesis committee consisting of Professor Jean-Luc Guermond [advisor], Professor Bojan Popov, and Professor Matthias Maier of the Department of Mathematics and Professor Jim Morel of the Department of Nuclear Engineering. In addition, part of the work was done in collaboration with Dr. Laura Saavedra from Universidad Politécnica de Madrid, with Dr. Vladimir Tomov from Lawrence Livermore National Laboratory, and with Dr. Nicolas Favrie from Aix-Marseille Université. All other work conducted for this thesis was completed by the student independently.

Funding Sources

This research was supported in part by the following grants: The National Science Foundation, grant DMS2110868, the Air Force Office of Scientific Research, USAF, under grant/contract number FA9550-18-1-0397, the Army Research Office, United States, under grant number W911NF-19-1-0431, under Subcontract No. B661293 with Lawrence Livermore National Security, LLC., and under the auspices of the U.S. Department of Energy in collaboration with Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

NOMENCLATURE

B/CS	Bryan and College Station
TAMU	Texas A&M University
LLNL	Lawrence Livermore National Laboratory
VDW	Van Der Waals
EOS	Equation of State
NASG	Noble-Abel Stiffened Gas
CFL	Courant-Friedrichs-Lowy
IDP	Invariant-Domain Preserving
DG	Discontinuous Galerkin
FEM	Finite Element Method
DoF	Degree of Freedom
GCL	Geometric Conservation Law
SGH	Staggered-Grid Hydrodynamics
CGH	Cell-centered Grid Hydrodynamics
ALE	Arbitrary Lagrangian Eulerian
LO	Low Order
HO	High Order
NH	Neo-Hookean
MR	Mooney-Rivlin

TABLE OF CONTENTS

	Page
ABSTRACT	ii
DEDICATION	iii
ACKNOWLEDGMENTS	iv
CONTRIBUTORS AND FUNDING SOURCES	v
NOMENCLATURE	vi
TABLE OF CONTENTS	vii
LIST OF FIGURES	xi
LIST OF TABLES.....	xv
1. EQUATIONS OF FLUID DYNAMICS	1
1.1 Introduction.....	1
1.2 Review on Hyperbolic Conservation Laws	6
1.3 The Riemann Problem	8
1.4 The Compressible Euler Equations.....	11
1.4.1 The Euler Equations in a Lagrangian Formulation	12
1.5 Thermodynamic Closure.....	17
1.5.1 Equations of State	18
1.5.2 Ideal Gas Law	19
1.5.3 Van der Waals.....	21
1.5.4 Noble-Abel Stiffened Gas.....	23
2. FINITE ELEMENT APPROXIMATION OF THE EULER EQUATIONS IN A LA- GRANGIAN FRAME	25
2.1 Finite Element Representation.....	26
2.1.1 Meshes	26
2.1.2 Finite Element Approximation of the State Variable	27
2.1.3 Finite Element Approximation of the Mesh Motion.....	28
2.1.3.1 Density Representation (Strong Mass Conservation).	30
2.2 The Fully Discrete Scheme	30
3. THE FIRST ORDER APPROXIMATION*	33

3.1	Stabilized Update	33
3.2	Graph Viscosity	35
3.3	Invariant-Domain Property	36
3.4	Mesh Velocity Reconstruction and Motion	37
3.4.1	Geometric Preliminaries	37
3.4.2	Reconstruction of v_h^n at the Vertices	39
3.4.2.1	1D Example	39
3.4.2.2	Higher Dimensions	40
3.5	Mass Conservation	41
3.5.1	Summary	42
4.	HIGHER ORDER APPROXIMATION VIA LIMITING	44
4.1	Mass Conservation	45
4.2	Review on Runge-Kutta Methods	46
4.2.1	General Explicit Runge-Kutta	46
4.2.2	Example Runge-Kutta Methods	47
4.3	Summary of the IDP-ERK Framework	51
4.4	Review on Limiters	52
4.4.1	The Setting	53
4.4.2	Obtaining the Local Bounds	54
4.4.3	Relaxation	56
4.4.4	Locally Conservative Limiting	56
4.5	Full Workflow	58
4.5.1	Summary of Modifications to Laghos	58
4.5.2	Interpolation from HO to LO	59
4.5.3	High Order Mesh Velocity	60
4.5.4	High-Order Density Evaluation (Pre-Limiting)	61
4.5.5	Stagewise Update of Thermodynamic and Kinematic Mass Matrices	61
5.	NONLINEAR HYPERELASTICITY	63
5.1	Introduction	63
5.2	Equations of Hyperelasticity Formulated in Lagrangian Coordinates	64
5.3	Closure of the System	65
5.3.1	Isotropic Closure	67
5.3.2	Anisotropic Closure	68
5.4	Extension of the Robust Low-Order Framework to Nonlinear Elasticity	71
6.	NUMERICAL RESULTS	73
6.1	Technical Details	73
6.2	Low Order Results*	73
6.2.1	Sod Shocktube	74
6.2.2	Distorted Sod Shocktube	74
6.2.3	Isentropic Vortex	76

6.2.3.1	Long Time Integration	76
6.2.4	Taylor-Green Vortex	78
6.2.5	Kidder Problem	80
6.2.6	Saltzman Problem	81
6.2.7	Van der Waals Equation of State	84
6.2.8	Radial Sod Problem	85
6.2.9	Noh Problem	86
6.2.10	Sedov Explosion	88
6.2.11	Multimaterial Triple Point	92
6.3	High Order Results	93
6.3.1	Approximation Space Notation	94
6.3.2	Smooth 1D	94
6.3.3	Van der Waals Equation of State	95
6.3.4	Isentropic Vortex	97
6.3.5	Taylor-Green Vortex	98
6.3.6	Multimaterial Triple Point	99
6.4	Nonlinear Hyperelasticity Results	100
6.4.1	Shocktube	101
6.4.2	Double Impact Problem	103
6.4.3	Shear Test	104
6.4.3.1	Neo Hookean Results	105
6.4.4	Impact Shear Test	107
6.4.5	Impact of a Projectile on a Solid Plate	109
6.4.5.1	Neo Hookean Results	110
6.4.5.2	Aortic EOS	111
	REFERENCES	116
	APPENDIX A. FUNDAMENTAL DERIVATIONS AND INVARIANCE PROPERTIES	129
A.1	Proof of Reynold's Transport Theorem	129
A.2	Alternate Derivation of the Euler Equations in a Lagrangian Frame	131
A.3	Galilean Invariance	132
	APPENDIX B. ALTERNATIVE MESH VELOCITY APPROACHES	135
B.1	Strategy 1: Optimization	135
B.2	Strategy 2: Face Correction	137
B.3	Comparison with Post-Processing	139
B.4	Clear Mesh Distortion	139
	APPENDIX C. SELECTED ALGORITHMS	141
C.1	IDP-ERK Time Stepping	141
C.2	Limiter	141
C.3	Relaxation	143

C.4	Single Stage of Limited Laghos	144
C.5	Nonlinear Hyperelasticity.....	144
	APPENDIX D. ELASTIC DEFINITIONS AND DERIVATIONS	146
D.1	Some Linear Algebra	146
D.2	Some Calculus.....	147
D.3	Cauchy Green Strain Tensor (Isotropic Case)	147

LIST OF FIGURES

FIGURE	Page
1.1 Example of a Riemann fan. The 1-wave, 2-wave, and the 4-wave are shocks or contact discontinuities, while the 4-wave is an expansion. The states $\mathbf{u}_L, \mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_R$ are constant.	9
1.2 Illustration of the Lagrangian mapping. The initial mesh (left) deforms into a distorted configuration (right) as material points are advected with the flow. Blue squares represent material point DoFs and red circles denote the Lagrange nodes with which the mesh is moved.	13
2.1 Jump of a piecewise constant function across the interface $F = K_l \cap K_r$	27
2.2 A typical Q_1 basis function.	29
3.1 Schematic of a single one-dimensional cell element. The red circles denote cell vertices/faces, and the blue square represents the dG0 approximation of our hydrodynamics quantities.	39
4.1 Illustration of the various stencils used in the high order implementation. Red DoFs represent the node indices that are part of the stencil. In the case of the max/min stencil, red submesh cells indicate the contained DoFs. Blue nodes indicate node i . Left: low order state update stencil $\mathcal{J}^*(i)$. Middle: max/min stencil $\mathcal{V}^*(i)$ defined on a Q_2 approximation space. Right: mass conservation stencil $\mathcal{K}^*(i)$ defined on a Q_2 approximation space.....	55
4.2 Examples of high order finite element mesh elements and the associated low order mesh elements. The low order mesh element boundaries are given by the dashed lines. Each dashed cell corresponds to a single low order degree of freedom.	55
4.3 Nodal limiting on a Q_1 element. One node (red) exceeds its local upper bound and the excess mass is redistributed within the element to neighboring DoFs (blue) that have remaining capacity per their bounds. Dashed lines indicate local admissible range for each DoF.	57
5.1 Examples of various fiber alignments. Left: $\theta = 0$, middle: $\theta = \frac{\pi}{4}$, right: $\theta = \frac{\pi}{2}$	69
6.1 Sod shocktube on an unstructured mesh at $t = 0.225$. Top left: 3D view of the density field. Top right: scatter plot of the density. Bottom: final mesh configuration. Reprinted with permission from [45].	75

6.2	Density field and mesh configuration at $t = 15$ for the isentropic vortex on a mesh composed of 65536 cells. Reprinted with permission from [45].	77
6.3	Velocity field and mesh at $t = 0$ and $t_f = 0.5$ for the Taylor-Green Vortex. Reprinted with permission from [45].	79
6.4	Density field and mesh configuration at $t = 0$ and $t = t_f$ for the Kidder problem on a mesh composed of 2000 cells. Reprinted with permission from [45].	82
6.5	Kidder problem. Time evolution of the external and internal radius of the shell versus the analytical solution. Reprinted with permission from [45].	82
6.6	Saltzman problem. Density field and mesh configuration at $t = 0.6$. Top: Solution obtained on the uniform mesh. Bottom: Solution obtained on the distorted meshes. Reprinted with permission from [45].	83
6.7	Saltzman problem on the distorted mesh. Scatter plots of the density and y -component of the velocity as functions of the x coordinate (blue dots) compared to the exact solution (black line). Left: density. Right: y -component of the velocity. Reprinted with permission from [45].	84
6.8	Van der Waals equation of state. Solution to the Riemann problem (6.2.7) at $t = 0.005$ with $CFL = 1$. From left to right: density; pressure; sound speed. Reprinted with permission from [45].	85
6.9	Radial Sod problem in 2D at $t = 0.25$ on a mesh composed of 57600 cells. Left: 3D rendering of the density field. Right: Scatter plot of density and 1D reference solution from [30]. Reprinted with permission from [45].	86
6.10	Density field and mesh from Noh problem on uniform meshes at $t = 0.6$. Reprinted with permission from [45].	88
6.11	Noh problem. Scatter plots of the density at $t = 0.6$ comparing the approximation (blue dots) with the exact solution (black solid line). Left: mesh with 14400 cells. Right: mesh with 57600 cells. Reprinted with permission from [45].	88
6.12	Noh problem on non-uniform mesh. Density and mesh at final time. Reprinted with permission from [45].	89
6.13	Density scatter plots at $t = 0.9$. Top row: Cartesian meshes. From left to right: quadrangular meshes with 14400 cells and 57600 cells. Bottom row: Results comparing cartesian vs. radial coarse meshes. Left: Cartesian mesh (900 cells) vs. radial mesh (896 cells). Right: Cartesian mesh (3600 cells) vs. radial mesh (3584 cells). Reprinted with permission from [45].	90
6.14	Sedov problem. Density and mesh composed of 230400 cells at $t = 0.9$. Reprinted with permission from [45].	91

6.15 Initial configuration for the triple point problem.....	92
6.16 Left: Density field for the triple-point problem at time of collapse $t = 1.101$ on a coarse mesh composed of 1,344 cells. Right: zoom of the mesh at the material boundaries at the time of collapse.	93
6.17 Density profiles for the van der Waals test (6.2.7) with a CFL number 1 using a $Q_2 - Q_1$ approximation space. The high-order Laghos solution with a Q_1 approximation (blue) becomes unstable and crashes within a single timestep in the absence of limiting. In contrast, the low-order invariant domain preserving method (black dashed) remains robust, and the limited solution (red) shows strict adherence to the local bounds. Left: Mesh composed of 8 cells run until $t_f = 0.0031$. Right: Mesh composed of 16 cells run until $t_f = 0.0017$	97
6.18 Time evolution of the relative amount of limited density constrained by the limiter for the isentropic vortex problem. Results are shown for both Q_1 and Q_2 approximations, with solid lines denoting average quantities across DoFs and dashed lines denoting maximum quantities.	99
6.19 Time evolution of the relative amount of limited density constrained by the limiter for the Taylor-Green vortex problem. Results are shown for both Q_1 and Q_2 approximations, with solid lines denoting average quantities across DoFs and dashed lines denoting maximum quantities.....	101
6.20 Triple-point problem at final time $t_f = 5$. Top: limited high-order density field (pair $Q_2 - Q_1$). Middle: low-order material indicator on the submesh. Bottom: limiter activity for density—the relative correction applied at the Q_1 DoFs.	102
6.21 Solution to the Riemann problem (6.4.2) at $t = 0.00005$ with $\text{CFL} = .5$. From left to right: density, velocity, stress.	103
6.22 Solution to the double impact problem at $t = 0.00005$ with $\text{CFL} = .5$. From left to right: density, velocity, stress.	104
6.23 Solution to the Riemann problem (6.4.3) at $t = 0.00005$ with $\text{CFL} = .5$ using the Mooney-Rivlin closure model. From top left to bottom right: density, normal velocity, tangential velocity, pressure, normal stress, tangential stress.	106
6.24 Solution to the Riemann problem (6.4.3) at $t = 0.00005$ with $\text{CFL} = .5$ using the Neo-Hookean closure model. From top left to bottom right: density, normal velocity, tangential velocity, pressure, normal stress, tangential stress.	107
6.25 Solution to the Riemann problem (6.4.4) at $t = 0.00005$ with $\text{CFL} = .5$ and the Mooney-Rivlin equation of state using the Mooney-Rivlin closure model. From top left to bottom right: density, normal velocity, tangential velocity, pressure, normal stress, tangential stress.	108

6.26 Initial configuration for the projectile-plate impact problem. The initial velocity of the projectile is 800 m/s while the plate is initially at rest.....	109
6.27 Projectile-plate impact problem with $\mu_s = 9.2 \times 10^9$. The schlieren image of the density is represented here at times $t = 3.2 \times 10^{-5}$, $t = 1.28 \times 10^{-4}$, $t = 2.27 \times 10^{-4}$, $t = 5.59 \times 10^{-4}$, $t = 9.16 \times 10^{-4}$, and $t = 1.417 \times 10^{-3}$. The Neo-Hookean equation of state is used for the sheer energy calculation.	110
6.28 Projectile-plate impact problem with $\mu_s = 0$. The schlieren plot of the density is represented here at collapse time 2.12×10^{-4} , along with a zoom in on the cells causing the collapse. We observe large gradients in the density, which is the likely culprit to early termination of the simulation.	111
6.29 Evolution of the density field in the projectile-plate impact problem ($\mu_s = 9.2 \times 10^9$) using the anisotropic aortic equation of state. Time increases from left to right ($t = 6.5 \times 10^{-5}$, 1.62×10^{-4} , 2.93×10^{-4}), and fiber orientation varies from top to bottom ($\theta = 0, \pi/4, \pi/2$).	113
6.30 Projectile-plate impact problem using the anisotropic aortic equation of state defined in Eq. 5.3.16, with fiber orientation angle $\theta = \pi/4$. The plots compare the tangential velocity component v_y at selected corner nodes indicated in the mesh (a). Subfigures (b) and (c) illustrate the temporal evolution of v_y for opposing corners in the projectile and plate, respectively, highlighting the directional dependence introduced by the anisotropic fiber alignment.....	114
6.31 Projectile-plate impact problem comparing fiber orientations at angles $\theta = 0, \pi/4$, and $\pi/2$. When the equation of state is isotropic (i.e., for $\theta = 0$ and $\pi/2$), the velocity profiles exhibit symmetry across opposing vertices. In contrast, for $\theta = \pi/4$, we observe an antisymmetric behavior in the tangential velocity component v_y , indicating directional sensitivity introduced by the fiber orientation.....	115
B.1 Motion of the face which preserves mass.	138
B.2 Meshes obtained with the mass conservative method using strategy 2 consisting of correcting mass defects at cell interfaces. Left: distorted Sod shocktube. Right: isentropic vortex.	140

LIST OF TABLES

TABLE	Page
1.1 Reference equations of state in caloric form $p = p(\rho, e)$	23
1.2 Reference speeds of sound for various equations of state.	24
5.1 Reference equations of state for shear energy.	70
6.1 Convergence tests for the Sod shocktube problem. Reprinted with permission from [45].	74
6.2 Convergence tests for the distorted Sod shocktube. Reprinted with permission from [45].	75
6.3 Composite L^1 -error for the isentropic vortex problem at $t = 1$ with $CFL = 0.5$. Reprinted with permission from [45].	77
6.4 Error and convergence rates for the Taylor-Green Vortex problem at final time $t_f = 0.5$. Reprinted with permission from [45].	79
6.5 Kidder problem. Error indicator $\delta^1(t_f)$ and convergence order obtained on different meshes. Reprinted with permission from [45].	81
6.6 Noh problem. Error indicator $\delta^1(0.6)$ and convergence order obtained on different meshes. Reprinted with permission from [45].	87
6.7 Modified composite L^1 -error for the Sedov problem at $t = 0.9$ using $CFL = 1$. Convergence is observed in the density, velocity, and total energy. Reprinted with permission from [45].	89
6.8 Convergence results for the smooth 1D test case with Q_1 approximation, demonstrating the expected order accuracy for all variants of the method.	95
6.9 Convergence results for the smooth 1D test case with Q_2 approximation, demonstrating the expected order accuracy for all variants of the method.	96
6.10 Convergence results for the smooth 1D test case with Q_3 approximation.	96
6.11 Convergence results for the isentropic vortex test case with Q_1 approximation, demonstrating the expected order accuracy for all variants of the method.	98

6.12	Convergence results for the isentropic vortex test case with Q_2 approximation, demonstrating the expected order accuracy for all variants of the method until the finer refinements where we believe the temporal error dominates the spatial error due to the CFL condition.....	98
6.13	Convergence results for the Taylor-Green vortex test case with Q_1 approximation, demonstrating the expected order accuracy for all variants of the method.	100
6.14	Convergence results for the Taylor-Green vortex test case with Q_2 approximation, demonstrating the expected order accuracy for all variants of the method until the finer refinements where we believe again the temporal error dominates the spatial error due to the CFL condition.....	100
6.15	Convergence test for the Riemann problem (6.4.2). Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.	103
6.16	Convergence test for the double impact problem. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.....	104
6.17	Convergence test for the 2D shear problem. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.	105
6.18	Convergence test for the 2D shear problem using the Neo-Hookean equation of state. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.	106
6.19	Convergence test for the 2D impact problem subject to shear velocity. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.	108
B.1	Convergence tests for the distorted Sod shocktube on quadrilateral meshes. Optimization-based method vs. post-processing vs. face correction.	139

1. EQUATIONS OF FLUID DYNAMICS

1.1 Introduction

The equations of hydrodynamics provide a continuum description of fluid motion, expressing conservation of mass, momentum, and total energy and have wide-ranging applicability. In biomedicine, the Navier-Stokes equations can be used to determine arterial wall stresses in the cardiovascular system. In hydrology and coastal engineering, the shallow water equations provide the core predictive models for large-scale surface flows, such as floods due to heavy rain or dam-break circumstances. These models can then be used to assess risk and ensure life-saving preventative measures are in place before disaster arises. In hypersonics, the governing equations of compressible flow inform the design of defense systems and entry/reentry vehicles under severe heating. The equations of radiation hydrodynamics help us to understand how laser energy sets ablation pressure in inertial confinement fusion prior to running a costly experiment at the National Ignition Facility, informing design and safety procedures. These examples are more illustrative than exhaustive.

All of the above examples characterize the necessity for the development of numerical methods that are robust in the following senses. First, the numerical method should preserve physical properties of the system, such as positivity of the density and temperature. Second, it should satisfy an entropy condition to select the physically admissible weak solution. Third, we will be particularly interested in interface tracking by employing pure Lagrangian methods and thus conservation of mass at a local scale becomes paramount. Fourth, our method should be flexible and equation of state agnostic, and in addition should be compatible with tabulated data. Fifth, there should be no knob tuning parameters involved in the approximation. That is to say that parameters remain consistent across all applications and test cases. Finally, the method should be high order accurate, meaning that the approximation converges quickly to the exact solution as the mesh is refined.

There are two natural formalisms that arise in the study of fluid mechanics, namely Eulerian and Lagrangian. The Eulerian framework is characterized by a fixed computational grid in which one can observe transfer of mass between cells. Eulerian methods are characterized by strong numerical diffusion due to the discretization of the convection terms. On the contrary, in Lagrangian hydrodynamics, the computational mesh moves with the fluid. In practice this means that the cell vertices move according to a computed velocity and the cell faces are uniquely specified by the updated vertex positions. By construction, there is no mass flowing between neighbor cells and hence no diffusion due to the advection of the fluid. For this reason, Lagrangian methods excel when material interface tracking is a priority. The main drawback of Lagrangian methods is the introduced complexity that comes with moving the mesh. The mesh velocity must be computed to be consistent with the conservative update of the hydrodynamic state to guarantee the Geometric Conservation Law (GCL) is satisfied. In addition, it is well known that the Lagrangian formalism is limited to certain types of flows that do not deform the mesh too much. In strongly vortical or shear-dominated flows, pure Lagrangian schemes suffer from severe distortion, producing non-convex or even inverted (tangled) cells. A common remedy is an Arbitrary Lagrangian–Eulerian (ALE) formulation, in which an explicit Lagrangian step is followed by (i) a rezoning step that restores cell geometry by constructing a new mesh, and (ii) a conservative remap of the Lagrangian solution onto that mesh. The ALE approach, introduced in the seminal work [51] and studied extensively thereafter [3, 65, 39, 2], is not treated further in this thesis.

The very first Lagrangian scheme was proposed by von Neumann and Richtmyer [85] for problems involving shocks using a finite difference method. This method employs a staggered arrangement where the position, velocity, and kinetic energy are defined at the nodes while the thermodynamic fields, such as density, pressure, and internal energy, are defined on cell centers. This class of moving-mesh methods is known as Staggered-Grid Hydrodynamics (SGH). It is unsurprising that this was the earliest moving-mesh model as the kinematic space is most naturally defined on the mesh entities that directly govern motion. To facilitate the case of shock wave propagation, the authors augment the Euler equations with a diffusion term which scales with the

mesh, known as artificial viscosity. In [86], a two-dimensional extension of the staggered-grid methodology of [85] for elastic–plastic flows revealed two major shortcomings: lack of energy conservation and the emergence of spurious numerical modes in 2D (the hourglass deformation of the mesh). Several remedies were proposed. The Damped Excess Vorticity (DEV) method of [9] computes vorticity directly from zone (cell) velocities—rather than integrating it as an independent variable, as in [29], which reduced sensitivity to unphysical nodal motion. Subsequently, Caramana and co-authors [12, 13, 11] introduced subcell forces to suppress hourglass modes and resolved the energy-conservation issue via a support operator discretization of the momentum and energy equations.

The alternative to SGH methods is to employ the cell-centered discretization where all conserved quantities are defined on cell centers, including the velocity. We refer to these as Cell-centered Grid Hydrodynamics (CGH) schemes. These methods are Godunov-based schemes [40], which solve the Euler equations in conservative form, for which conservation arises naturally, in contrast to staggered-grid schemes which advance a non-conservative internal-energy equation rather than a conservative total energy balance. The principal drawback of the cell-centered regime is that there is no clear notion on how to move the mesh. In [1], the authors introduce the CAVEAT scheme in which the node velocity is computed from the intermediate face velocity fluxes using a weighted least squares approach. The main drawback is that this calculation violates the Geometric Conservation Law (GCL) and leads to artificial grid motion. In [27, 14], the authors introduce the GLACE (Godunov-type LAgrangian scheme Conservative for total Energy) scheme, which is an extension of subcell forces employed in the SGH methods. The GLACE scheme solved the issue of having a mesh velocity compatible with the GCL that is conservative in both momentum and total energy. Subsequently, an alternative scheme named EUCLHYD (Explicit Unstructured Cell-Centered Lagrangian HYDrodynamics) was proposed in [64, 62], in which two nodal pressures are defined per corner in an effort to remove the scheme dependency on the cell aspect ratio. Both the GLACE and EUCLHYD schemes are based on the finite volume approach.

In addition, there has been an effort to approximate the equations of Lagrangian hydrodynamics

using finite elements, which discretize the weak form of the governing equations. In [60], Abgrall et al. present a new Lagrangian Discontinuous Galerkin-type method using a basis of Bernstein polynomials defined on the barycentric coordinates on a mesh composed of triangles. In [79, 80], Scovazzi et al. propose a Streamline-upwind Petrov-Galerkin (SUPG) stabilized CGH approach using a piecewise linear approximation, which prevents locking and artificial stiffness prevalent in approximations using linear elements on simplex-type meshes. In [53, 82, 83, 58], the dG scheme is formulated on a fixed reference (initial) configuration. The deformation is carried by the time-dependent Jacobian of the Lagrangian map, so the conservation laws are written entirely in Lagrangian coordinates. This is the formulation adopted here. By contrast, [87, 37] advance the conserved variables in the Eulerian frame and therefore do not require the Jacobian map. In [87], mesh velocities come from an approximate nodal Riemann solver (with convergence demonstrated only in 1D), whereas [37] uses a CAVEAT-style averaging to compute nodal velocities.

The purpose of this thesis is three-fold. First, we develop a first order accurate invariant-domain preserving (IDP) method for the compressible Euler equations formulated in a purely Lagrangian frame. We do this by adopting the cell-centered pure Lagrangian framework, where we discretize all thermodynamic quantities (specific volume, velocity, specific total energy) on cell centers using the discontinuous Q_0 approximation. We also introduce a "graph viscosity" term, which is essential to proving invariant-domain properties. The update is written as a convex combination of states, and the viscosity coefficients are chosen under a Courant-Friedrichs-Lowy (CFL) restriction so that the intermediate states remain inside the invariant set; these coefficients are determined from an upper bound on the maximum wave speed of the local (face-aligned) Riemann problem. We approximate the mesh velocity field with continuous nodal Q_1 elements, and compute their values based on a local linear matrix solve. Though the GCL is preserved exactly in 1D, violation of the GCL in 2D is mitigated by a post-processing step which ensures that the update to the specific volume is locally mass conservative. We show that this method is indeed robust in the aforementioned senses.

The second purpose of this thesis is to then implement our robust first order method as a base-

line method to establish bounds on an existing high order method. The high order method is then limited to respect those bounds, making it invariant-domain preserving in the specific volume. The limiting strategy we describe is not restricted to a single implementation but rather can be applied to any high order approximation technique that solves the system of Lagrangian Hydrodynamics. In this thesis, we specify how our robust first order CGH method is used to limit a high order, possibly invariant-domain violating SGH method. High order accuracy in time is obtained by adopting the Invariant-Domain Preserving Explicit Runge-Kutta (ERK) time stepping scheme described in [33].

Finally, we extend our formulation for compressible fluids to the case of nonlinear hyperelastic solids. The model introduced in Section 3 is modified by replacing the isotropic pressure term in 1.4.16 with a general Cauchy stress tensor that captures elastic response to deformation. We then introduce an additional degree of freedom associated with the energy due to deformation, which we call shear energy, for which we must solve. In this work, we adopt both isotropic and anisotropic closure models. Several previously mentioned methods have adapted their schemes for hyperelastic solids. EUCLHYD was extended to hyperelasticity using ADER (Arbitrary high order schemes using DERivatives) time integration and the (Multidimensional Optimal Order Detection) MOOD a posteriori limiting strategy to achieve robust high-order accuracy [6]. GLACE has similarly been adapted to hyperelastic materials [55].

The remainder of this Chapter will be dedicated to a review of thermodynamics. We begin with a general discussion on hyperbolic conservation laws, then introduce the Riemann problem which will be essential in ensuring that our robust low order scheme is IDP. We then present a derivation of the compressible Euler equations in a Lagrangian frame and end with a general discussion on various equations of state that will be used in this thesis. In Chapter 2 we provide a general discussion of the finite element approximation to the Euler equations in a Lagrangian frame. In Chapter 3 we describe our first order method and prove that it is indeed invariant-domain preserving. In Chapter 4 we outline how our low order method is used to establish bounds and limit an existing high order method. In Chapter 5, we extend the first order method to the nonlinear

hyperelasticity regime. Notably this requires only modest adjustments to the scheme presented in Chapter 3. Finally, numerical results are presented in Chapter 6.

1.2 Review on Hyperbolic Conservation Laws

We begin with some background on the theory of hyperbolic conservation laws. Consider the partial differential equation (PDE)

$$\partial_t \mathbf{u} + \nabla \cdot \mathbf{f}(\mathbf{u}) = 0, \quad \mathbf{x} \in \mathbb{R}^d, \quad t > 0, \quad (1.2.1a)$$

$$\mathbf{u}(\mathbf{x}, 0) = \mathbf{u}_0(\mathbf{x}) \in \mathcal{B} \subset \mathbb{R}^m, \quad (1.2.1b)$$

where d is the spatial dimension and

$$\mathbf{u}(\mathbf{x}, t) = (u_1(\mathbf{x}, t), \dots, u_m(\mathbf{x}, t))^T$$

is the vector of conserved variables. The flux $\mathbf{f} \in C^2(\mathbb{R}^m; \mathbb{R}^{m \times d})$ is given by

$$\mathbf{f}(\mathbf{u}) = (f_1(\mathbf{u}), \dots, f_d(\mathbf{u})),$$

where each $f_k : \mathcal{B} \subset \mathbb{R}^m \rightarrow \mathbb{R}^m$ has the component form $f_k(\mathbf{u}) = (f_{k1}(\mathbf{u}), \dots, f_{km}(\mathbf{u}))^T$. The set \mathcal{B} is a subset of the phase space and will be further examined in Section 3.3.

The system (1.2.1) is written in conservative form, but it is often useful to rewrite it in the quasilinear form

$$\partial_t \mathbf{u} + \sum_{k=1}^d \mathbb{A}_k(\mathbf{u}) \partial_{x_k} \mathbf{u} = 0, \quad \mathbf{x} \in \mathbb{R}^d, \quad t > 0, \quad (1.2.2)$$

where

$$\mathbb{A}_k(\mathbf{u}) = \left(\frac{\partial f_{ki}}{\partial u_j}(\mathbf{u}) \right)_{1 \leq i, j \leq m}.$$

Definition 1 (Hyperbolic System). Define

$$\mathbb{A}(\mathbf{u}; \mathbf{n}) := \sum_{k=1}^d \mathbb{A}_k(\mathbf{u}) n_k, \quad \mathbf{n} = (n_1, \dots, n_d)^T, \quad \|\mathbf{n}\|_{\ell^d} = 1.$$

We say that the system (1.2.1) is hyperbolic if $\mathbb{A}(\mathbf{u}; \mathbf{n})$ has m real eigenvalues

$$\lambda_1(\mathbf{u}; \mathbf{n}), \dots, \lambda_m(\mathbf{u}; \mathbf{n})$$

for all $\mathbf{u} \in \mathcal{B}$ and $\mathbf{n} \in \mathbb{S}^{d-1}$. The system is called strictly hyperbolic if these eigenvalues are all distinct.

It is well known that even for smooth initial data, solutions to hyperbolic conservation laws may develop discontinuities in finite time. Therefore, we look for weak solutions to (1.2.1).

Definition 2 (Weak Solution). A function $\mathbf{u} \in [L^1(\mathbb{R}^d \times (0, \infty))]^m$ is called a weak solution to (1.2.1) if

$$\int_{\mathbb{R}^d} \int_0^\infty (\mathbf{u} \partial_t \varphi + \mathbf{f}(\mathbf{u}) \cdot \nabla \varphi) dt d\mathbf{x} = - \int_{\mathbb{R}^d} \mathbf{u}_0(\mathbf{x}) \varphi(\mathbf{x}, 0) d\mathbf{x}, \quad (1.2.3)$$

holds for all $\varphi \in C_c^1(\mathbb{R}^d \times [0, \infty))$.

We are interested in physically relevant weak solutions, namely those obtained as limits of vanishing viscosity approximations. Such solutions satisfy entropy inequalities.

Definition 3 (Entropy Solution). A weak solution $\mathbf{u} \in [L^\infty(\mathbb{R}^d \times [0, \infty))]^m$ of (1.2.1) is called an entropy solution if

$$\int_{\mathbb{R}^d} \int_0^\infty (\eta(\mathbf{u}) \partial_t \varphi + \mathbf{q}(\nu) \cdot \nabla \varphi) dt d\mathbf{x} \geq 0, \quad (1.2.4)$$

for all nonnegative $\varphi \in C_c^1(\mathbb{R}^d \times [0, \infty))$ and for every entropy–entropy flux pair (η, \mathbf{q}) with η convex. Equivalently, one may write that \mathbf{u} satisfies in the weak sense

$$\partial_t \eta(\mathbf{u}) + \nabla \cdot \mathbf{q}(\mathbf{u}) \leq 0$$

Entropy solutions are important since they exclude many non-physical weak solutions. To better understand the structure of entropy solutions, we now consider the one-dimensional Riemann problem.

1.3 The Riemann Problem

This canonical problem, involving initial data with a single discontinuity, serves as the building block for analyzing waves and discontinuities in hyperbolic systems.

Definition 4. Let $\mathbf{g} \in C^1(\mathbb{R}^m; \mathbb{R}^m)$ be the flux. The **Riemann problem** is written as: Find $\mathbf{u} \in L^\infty(\mathbb{R} \times (0, \infty); \mathbb{R}^m) \cap C^0((0, \infty); L_{loc}^1(\mathbb{R}; \mathbb{R}^m))$ satisfying

$$\partial_t \mathbf{u}(x, t) + \partial_x \mathbf{g}(\mathbf{u}(x, t)) = 0 \quad (1.3.1a)$$

$$\mathbf{u}(x, 0) = \mathbf{u}_0(x) = \begin{cases} \mathbf{u}_L & \text{if } x < 0 \\ \mathbf{u}_R & \text{if } x > 0, \end{cases} \quad (1.3.1b)$$

where \mathbf{u}_L and \mathbf{u}_R represent a left and a right state that are sufficiently close.

We assume that the system is strictly hyperbolic and order the real and distinct eigenvalues as

$$\lambda_1 < \lambda_2 < \dots < \lambda_m.$$

The structure of the solution of the Riemann problem (1.3.1) can be constructed using Lax's method [56, Sec.9]. The solution consists of $m + 1$ constant states separated by m waves corresponding to the distinct eigenvalues. Specifically, there are $2m$ numbers

$$\lambda_1^- \leq \lambda_1^+ \leq \lambda_2^- \leq \dots \leq \lambda_m^- \leq \lambda_m^+ \quad (1.3.2)$$

which define $2m + 1$ sectors in the (x, t) plane, some of which may be empty, such that the solution is \mathbf{u}_L in the first sector $\{\frac{x}{t} \in (-\infty, \lambda_1^-)\}$, \mathbf{u}_R in the last sector $\{\frac{x}{t} \in (\lambda_m^+, \infty)\}$, and the solution in the remaining sectors is either a constant state or an expansion wave. See Figure 1.1 for a depiction

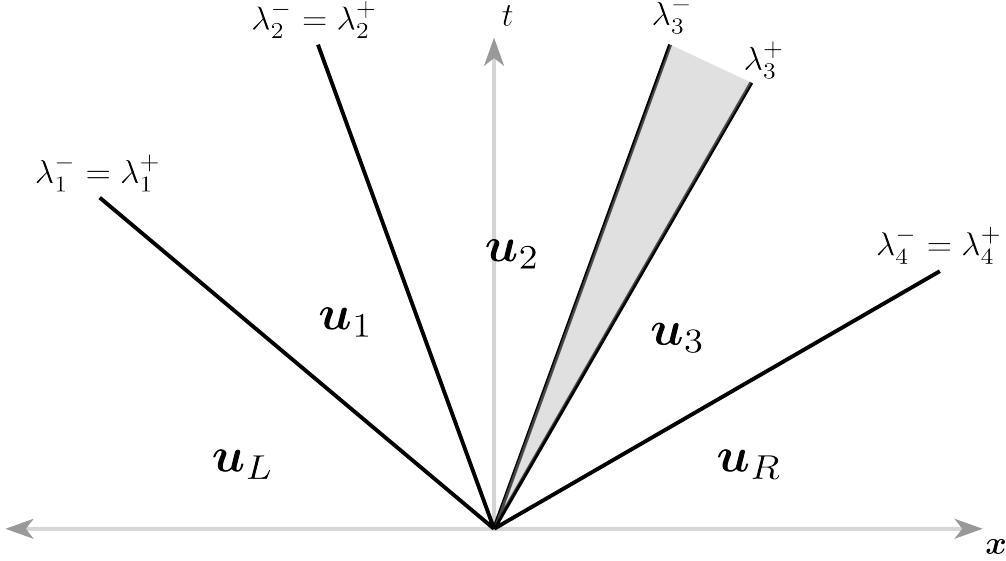


Figure 1.1: Example of a Riemann fan. The 1-wave, 2-wave, and the 4-wave are shocks or contact discontinuities, while the 3-wave is an expansion. The states $\mathbf{u}_L, \mathbf{u}_1, \mathbf{u}_2, \mathbf{u}_3, \mathbf{u}_R$ are constant.

in the $x - t$ plane of this reconstruction technique.

For the purposes of this thesis, it is not necessary to reconstruct a solution to the Riemann problem, but to estimate the max wave speed. That is, we need an upper bound on

$$\lambda_{\max} := \max \{ |\lambda_1^-|, |\lambda_m^+| \}. \quad (1.3.3)$$

In regions of the Riemann problem where discontinuities appear, the conserved variables experience finite jumps between neighboring constant states. The propagation speed of such discontinuities is not arbitrary; it must satisfy a compatibility condition derived from the integral form of the conservation law. This condition, known as the Rankine–Hugoniot condition which relates the speed of the discontinuity S to the states on either side of the jump.

Definition 5 (Rankine–Hugoniot Conditions). *The Rankine–Hugoniot conditions are defined as*

$$S (\mathbf{u}_R - \mathbf{u}_L) = f(\mathbf{u}_R) - f(\mathbf{u}_L), \quad (1.3.4)$$

where \mathbf{u}_L and \mathbf{u}_R are the left and right states adjacent to the discontinuity and f is the flux.

In the context of our numerical method, one of the guiding principles is that the solution should remain within the physically admissible phase space. This motivates the following.

Definition 6 (Invariant Region). A set $\mathcal{B} \subset \mathbb{R}^m$ is called an invariant region for (1.2.1) if $\mathbf{u}(\mathbf{x}, 0) \in \mathcal{B}$ for all $\mathbf{x} \in \mathbb{R}^d$ and there exists $T > 0$ such that $\mathbf{u}(\mathbf{x}, t) \in \mathcal{B}$ for all $\mathbf{x} \in \mathbb{R}^d$, $0 < t < T$.

Intuitively, an invariant region is a multidimensional analogue of a maximum principle: if the initial state lies within \mathcal{B} , the evolution of the system guarantees that the solution remains inside \mathcal{B} for all later times. This property ensures that physically meaningful bounds, such as positivity of density and a notion of positivity of temperature, are preserved throughout the simulation.

We now state a result for the solution to the Riemann problem (1.3.1) that will be essential when we prove our first order method is invariant-domain preserving in Chapter 3.

Lemma 1 (Average). Let $\mathbf{u} \in L^\infty(\mathbb{R} \times (0, \infty); \mathbb{R}^m) \cap C^0((0, \infty); L^1_{loc}(\mathbb{R}; \mathbb{R}^m))$ be a solution to the Riemann problem (1.3.1) with $\mathbf{u}_L, \mathbf{u}_R \in \mathcal{B}$. Let $\bar{\mathbf{u}}(t) := \int_{-\frac{1}{2}}^{\frac{1}{2}} \mathbf{u}(x, t) dx$. Then, for all $t \in (0, \frac{1}{2\lambda_{\max}}]$, where λ_{\max} is the maximal wave speed for the Riemann problem defined in (1.3.3), the following holds:

$$(i) \quad \bar{\mathbf{u}}(t) = \frac{1}{2}(\mathbf{u}_L + \mathbf{u}_R) - t(\mathbf{f}(\mathbf{u}_R) - \mathbf{f}(\mathbf{u}_L))\mathbf{n}.$$

$$(ii) \quad \bar{\mathbf{u}}(t) \in \mathcal{B}.$$

Proof. The proof of this lemma is taken from [45, Lem. A.5]. Fix $t \in (0, \frac{1}{2\lambda_{\max}}]$. Let u_1, \dots, u_3 be the components of \mathbf{u} and g_1, g_2, g_3 be the components of the flux \mathbf{g} . Let $l \in \{1, 2, 3\}$. Let $x \in \mathbb{R}$ be the dummy variable for space and $s \in [0, \infty)$ be the dummy variable for time. (i) Since \mathbf{u} is a weak solution to 1.3.1, we have

$$0 = \int_{-\infty}^{\infty} \int_0^{\infty} (-u_l \partial_s \phi - g_l(\mathbf{u}) \partial_x \phi) ds dx - u_{l,L} \int_{-\infty}^0 \phi(x, 0) dx - u_{l,R} \int_0^{\infty} \phi(x, 0) dx$$

for all $\phi \in W^{l,\infty}(\mathbb{R} \times [0, \infty); \mathbb{R})$ with compact support in $\mathbb{R} \times [0, \infty)$ and such that $\partial_s \phi$ is continuous at $x = 0$. Here, $u_{l,Z}$ denotes the l th component of $Z \in \{L, R\}$ and g_l denotes the l th component

of \mathbf{g} . We define a sequence of smooth functions $(\phi_\epsilon)_{\epsilon>0}$ with $\phi_\epsilon(x, s) = \phi_{1,\epsilon}(|x|)\phi_{2,\epsilon}(s)$

$$\phi_{1,\epsilon}(x) = \begin{cases} 1 & 0 \leq x \leq \frac{1}{2}, \\ \frac{1}{\epsilon}(-x + \frac{1}{2} + \epsilon) & \frac{1}{2} \leq x \leq \frac{1}{2} + \epsilon, \\ 0 & \frac{1}{2} + \epsilon \leq x, \end{cases} \quad \phi_{2,\epsilon}(s) = \begin{cases} 1 & 0 \leq s \leq t, \\ \frac{1}{\epsilon}(-s + t + \epsilon) & t \leq s \leq t + \epsilon, \\ 0 & t + \epsilon \leq s. \end{cases}$$

Using that $u_l \in C^0([0, \infty); L^1_{\text{loc}}(\mathbb{R}))$, we infer that $\int_{-\infty}^{\infty} \int_0^{\infty} -u_l \partial_s \phi_\epsilon \, ds \, dx \rightarrow \int_{-\frac{1}{2}}^{\frac{1}{2}} u_l(x, t) \, dx$ as $\epsilon \rightarrow 0$. Similarly in the limit as $\epsilon \rightarrow 0$, we have

$$\int_{-\infty}^{\infty} \int_0^{\infty} -g_l(\mathbf{u}) \partial_x \phi_\epsilon \, ds \, dx \rightarrow \int_0^t [g_l(\mathbf{u}_R) - g_l(\mathbf{u}_L)] \, ds = [g_l(\mathbf{u}_R) - g_l(\mathbf{u}_L)] t.$$

Additionally, $-u_{l,L} \int_{-\infty}^0 \phi_\epsilon(x, 0) \, dx - u_{l,R} \int_0^{\infty} \phi_\epsilon(x, 0) \, dx \rightarrow -\frac{1}{2}(u_{l,L} + u_{l,R})$ as $\epsilon \rightarrow 0$. In conclusion, we have established that

$$0 = \bar{\mathbf{u}}(t) + (\mathbf{g}(\mathbf{u}_R) - \mathbf{g}(\mathbf{u}_L))t - \frac{1}{2}(\mathbf{u}_L + \mathbf{u}_R).$$

(ii) Since \mathcal{B} is convex, $\mathbf{u}(x, t) \in \mathcal{B}$ for almost every $x \in \mathbb{R}$ and $t > 0$. Since the length of the interval $[-\frac{1}{2}, \frac{1}{2}]$ is 1, we infer that $\bar{\mathbf{u}}(t) \in \mathcal{B}$. \square

1.4 The Compressible Euler Equations

We begin now with the Euler equations for an inviscid compressible fluid, which represent the conservation of mass, momentum and total energy

$$\frac{\partial}{\partial t} \rho + \nabla \cdot (\rho \mathbf{v}) = 0, \tag{1.4.1a}$$

$$\frac{\partial}{\partial t} (\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v} + \sigma) = \mathbf{0}, \tag{1.4.1b}$$

$$\frac{\partial}{\partial t} (\rho E) + \nabla \cdot (\rho E \mathbf{v} + \sigma \mathbf{v}), \tag{1.4.1c}$$

where ρ is the density, $\mathbf{v} = (v_1, v_2, \dots, v_d)$ is the velocity, E is the specific total energy, \otimes is the outer product, d is the spatial dimension, and σ is the stress tensor. Unless otherwise noted, particularly in the elastic implementation discussed in 5, the stress tensor will have the form $\sigma = -p\mathbb{I}_d$ where $\mathbb{I}_d \in \mathbb{R}^{d \times d}$ is the identity matrix and p is the thermodynamics pressure, which can be expressed as a function of the density and specific internal energy $e = E - \frac{1}{2}\|\mathbf{v}\|^2$ through a closure equation of state of the form $p = \text{EOS}(\rho, e)$. For more information on the equations of state used in this thesis, see Section 1.5.1.

1.4.1 The Euler Equations in a Lagrangian Formulation

The goal of this section is to recall how the gas dynamics equations are obtained in the Lagrangian formulation. In Lagrangian motion, we consider a continuous medium D^t of particles $\{\mathbf{x}\}$ which deforms in time. The initial configuration of this medium at time t_0 will be denoted $\tilde{D} = D(t_0) = D^0$, with its corresponding particles $\{\xi\}$. In the Lagrangian description of motion, a fluid's particle location can be written as a function of initial particle location ξ and time t with the Lagrange mapping $\Phi(\cdot, t) : \tilde{D} \rightarrow D^t$, defined by

$$\mathbf{x} = \Phi(\xi, t) \quad \forall \xi \in \tilde{D}, t \geq t_0. \quad (1.4.2)$$

The Lagrangian mapping from the reference configuration to the current configuration is illustrated in Figure 1.2, where material points and mesh nodes are advected according to the mesh velocity field.

Remark 1.4.1 (Coordinates). *In the literature, the coordinates of point ξ are referred to as the Lagrangian coordinates and the coordinates of point \mathbf{x} are referred to as the Eulerian coordinates and correspond to the particle position relative to an outside observer.*

The mapping Φ is determined by the velocity field of the fluid. Specifically, if $\mathbf{v}(x, t)$ denotes the Eulerian velocity, then the particle trajectories satisfy the system of ordinary differential equations

$$\partial_t \Phi(\xi, t) = \mathbf{v}(\Phi(\xi, t), t), \quad \Phi(\xi, t_0) = \xi. \quad (1.4.3)$$

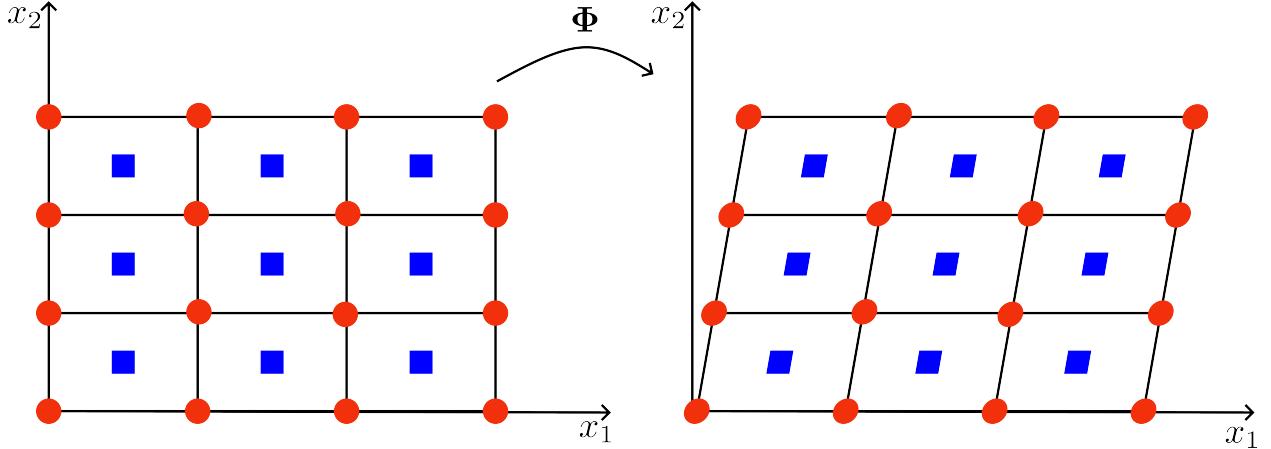


Figure 1.2: Illustration of the Lagrangian mapping. The initial mesh (left) deforms into a distorted configuration (right) as material points are advected with the flow. Blue squares represent material point DoFs and red circles denote the Lagrange nodes with which the mesh is moved.

Provided the velocity field \mathbf{v} is sufficiently smooth, the system (1.4.3) admits a unique solution. Thus, $\Phi(\xi, t)$ advances each fluid particle from its initial position at time t_0 to its current position at time t . This function corresponds to an observer moving with the medium. We introduce as well, the Jacobian of this mapping

$$\mathbb{J}(\xi, t) := \nabla_\xi \Phi(\xi, t). \quad (1.4.4)$$

Notice that $|\mathbb{J}(\xi, 0)| = 1$. Since we assume Φ_t is invertible, we must have for all $t > 0$ that $|\mathbb{J}(\xi, t)| > 0$.

Remark 1.4.2 (Notation (Jacobian)). *In the literature, there is no ubiquitous notation when it comes to referring to the Jacobian. In this thesis, $\mathbb{J} \in \mathbb{R}^{d \times d}$ will denote the derivative of the mapping Φ with respect to the Lagrangian coordinates ξ and $|\mathbb{J}| =: J \in \mathbb{R}$ will denote the determinant. Occasionally, both these quantities are referred to as the "Jacobian" in the literature.*

Remark 1.4.3 (Stopping criteria). *If during our simulation we have $|\mathbb{J}(\xi, t)| \leq 0$, this indicates that our mesh has deformed too much and is no longer viable, at which point the Lagrangian simulation must end.*

The conservation laws of Lagrangian hydrodynamics are a consequence of a well-known result called the *Reynolds Transport Theorem*.

Theorem 1.4.4 (Reynold's Transport Theorem). *Let $\alpha(\mathbf{x}, t)$ be a function that is defined over a time dependent region D^t . Then*

$$d_t \int_{D^t} \alpha d\mathbf{x} = \int_{D^t} \left\{ \frac{\partial \alpha}{\partial t} + \alpha (\nabla \cdot \mathbf{v}) \right\} d\mathbf{x}. \quad (1.4.5)$$

For a detailed proof of 1.4.4, see Section A.1. In addition, an alternative derivation of the Euler Equations in the Lagrangian Formalism can be found in Section A.2, which follows the more traditional approach based on Theorem 1.4.4 found in [64] and [17].

Remark 1.4.5 (Time–derivative notation). *To avoid ambiguity in Lagrangian settings, we distinguish partial versus along-trajectory time derivatives. Let $q(\Phi(\xi, t), t)$ be an arbitrary quantity expressed in Lagrangian coordinates. Then*

i $\partial_t q$ denotes the **partial derivative** with respect to the explicit time argument while holding the other arguments fixed (e.g., $\partial_t q(\Phi(\xi, t), t)$ at fixed ξ).

ii $d_t q$ denotes the time derivative **along the Lagrangian trajectory** of a material point (i.e., with ξ fixed while $\mathbf{x} = \Phi(\xi, t)$ evolves). Consequently, $d_t q(\xi, t) = \partial_t q(\xi, t)$. If a field is expressed in Eulerian variables, then along the path $\mathbf{x} = \Phi(\xi, t)$ one has

$$d_t q(\mathbf{x}, t) = \partial_t q(\mathbf{x}, t) + \mathbf{v}(\mathbf{x}, t) \cdot \nabla q(\mathbf{x}, t). \quad (1.4.6)$$

This is commonly referred to in the literature as the **material derivative**.

The same convention applies componentwise to vector/tensor fields.

To begin our derivation, we start by defining the vector

$$\hat{\mathbf{u}} := \begin{pmatrix} 1 \\ \mathbf{v} \\ E \end{pmatrix} \in \mathbb{R}^{d+2}. \quad (1.4.7)$$

Then, the Euler system in Eqs. 1.4.1 can be written

$$\partial_t (\rho \hat{\mathbf{u}}) + \nabla \cdot \begin{pmatrix} \rho \mathbf{v} \\ \rho \mathbf{v} \otimes \mathbf{v} + p \\ \rho \mathbf{v} E + \mathbf{v} p \end{pmatrix} = \mathbf{0}.$$

By the product rule we have

$$\rho \partial_t \hat{\mathbf{u}} + \hat{\mathbf{u}} \partial_t \rho + \begin{pmatrix} \nabla \cdot (\rho \mathbf{v}) \\ \rho \mathbf{v} \cdot \nabla \mathbf{v} + \mathbf{v} \nabla \cdot (\rho \mathbf{v}) + \nabla p \\ \rho \mathbf{v} \cdot \nabla E + E \nabla \cdot (\rho \mathbf{v}) + \nabla \cdot (\mathbf{v} p) \end{pmatrix} = \mathbf{0}.$$

Recall that the conservation of mass equation given in (1.4.1a) implies that

$$\hat{\mathbf{u}} \partial_t \rho + \hat{\mathbf{u}} \nabla \cdot (\rho \mathbf{v}) = 0. \quad (1.4.8)$$

Using (1.4.8), we can eliminate these terms from (1.4.1) to yield

$$\rho \partial_t \hat{\mathbf{u}} + \nabla \cdot \begin{pmatrix} 0 \\ p \mathbb{I} \\ \mathbf{v} p \end{pmatrix} + \begin{pmatrix} 0 \\ \rho \mathbf{v} \cdot \nabla \mathbf{v} \\ \rho \mathbf{v} \cdot \nabla E \end{pmatrix} = \mathbf{0}. \quad (1.4.9)$$

Notice however that the first term in (1.4.9) is already the partial derivative with respect to t of the vector of quantities $\hat{\mathbf{u}}$. Our goal is to write the Euler equations in the Lagrangian formalism, where the spatial argument of the conserved quantities is in terms of the Lagrangian mapping function

defined in (1.4.2). So, by the chain rule we have

$$\rho d_t [\hat{\mathbf{u}} (\Phi (\xi, t), t)] = \rho \nabla_{\Phi} \hat{\mathbf{u}} \cdot \partial_t \Phi + \rho \partial_t \hat{\mathbf{u}} = \begin{pmatrix} 0 \\ \rho \mathbf{v} \cdot \nabla \mathbf{v} \\ \rho \mathbf{v} \cdot \nabla E \end{pmatrix} + \rho \partial_t \hat{\mathbf{u}}. \quad (1.4.10)$$

Using (1.4.10), we can rewrite (1.4.9) to finally get

$$\rho d_t [\hat{\mathbf{u}} (\Phi (\xi, t), t)] + \nabla \cdot \begin{pmatrix} 0 \\ p \mathbb{I} \\ p \mathbf{v} \end{pmatrix} = \mathbf{0}. \quad (1.4.11)$$

At this point, we still require a closure equation for the specific volume. We look at the specific volume $\tau (\Phi (\xi, t), t) = \frac{1}{\rho} (\Phi (\xi, t), t)$. By the chain rule, we have

$$\rho d_t [\tau (\Phi (\xi, t), t)] = \rho d_t \left[\frac{1}{\rho} (\Phi (\xi, t), t) \right] = \rho \left(\nabla \left(\frac{1}{\rho} \right) \cdot \partial_t \Phi + \partial_t \frac{1}{\rho} \right) (\Phi (\xi, t), t) \quad (1.4.12)$$

$$= -\frac{1}{\rho} (\nabla \rho \cdot \mathbf{v} + \partial_t \rho) (\Phi (\xi, t), t). \quad (1.4.13)$$

By the product rule, we can rewrite the equation for conservation of mass in (1.4.1a) as

$$\partial_t \rho + \nabla \rho \cdot \mathbf{v} + \rho (\nabla \cdot \mathbf{v}) = 0 \implies \partial_t \rho + \nabla \rho \cdot \mathbf{v} = -\rho (\nabla \cdot \mathbf{v}). \quad (1.4.14)$$

This then allows us to write

$$\rho d_t \tau (\Phi (\xi, t), t) - \nabla \cdot \mathbf{v} (\Phi (\xi, t), t) = 0. \quad (1.4.15)$$

Putting together (1.4.11) and (1.4.15) yields the compressible Euler equations formulated in La-

grangian coordinates

$$\frac{1}{\tau(\Phi(\xi, t), t)} d_t(\mathbf{u}(\Phi(\xi, t), t)) + (\nabla \cdot \mathbf{f}(\mathbf{u}))(\Phi(\xi, t), t) = \mathbf{0}, \quad \mathbf{u}(\xi, 0) = \mathbf{u}_0(\xi) \quad (1.4.16a)$$

$$\text{with } \partial_t \Phi(\xi, t) := \mathbf{v}(\Phi(\xi, t), t), \quad \Phi(\xi, 0) = \xi, \quad (1.4.16b)$$

$$\text{and } \mathbf{f}(\tau, \mathbf{v}, E) := (-\mathbf{v}, p(\mathbf{u})\mathbb{I}_d, vp)^\top. \quad (1.4.16c)$$

where the components of the dependent variable $\mathbf{u} := (\tau, \mathbf{v}, E)^\top \in \mathbb{R}^{d+2}$ (considered to be a column vector) are the specific volume, τ , the velocity, \mathbf{v} , and the specific total energy, E . Here \mathbb{I}_d is the $d \times d$ identity matrix and $D^t := \Phi(D^0, t)$ for all $t \in [0, T]$.

1.5 Thermodynamic Closure

The governing equations of compressible fluid dynamics, given in Eqs. (1.4.1), are not by themselves sufficient to completely describe the system. In their conservative form, these equations contain more unknowns than relations, and thus an additional condition is required to close the system. This closure is provided by constitutive relations that express the thermodynamic properties of the material.

In particular, the total energy involves the specific internal energy e , a thermodynamic variable not directly determined by the conservation laws. Closure is provided by an equation of state (EOS) that links e to the other state variables (e.g., density/specific volume τ , pressure p , temperature T). We model the EOS as an oracle: a pressure mapping $p(\mathbf{u})$, defined on a suitable thermodynamic admissible set $\mathcal{B}(b, q, p_\infty) \subset \mathbb{R}^{d+2}$, given by

$$\mathcal{B}(b, q, p_\infty) := \{\mathbf{u} := (\tau, \mathbf{v}, E) \in \mathbb{R}^{d+2} \mid b < \tau < \infty; p_\infty(\tau - b) < e(\mathbf{u}) - q\}, \quad (1.5.1)$$

where the parameters q , $0 \leq p_\infty$, and $0 \leq b$ are given by the user. In regions where the EOS is invertible with respect to e , one may equivalently write $e = e(\tau, p)$. Treating $p(\mathbf{u})$ as an oracle allows us to accommodate general materials without committing to a specific analytic form.

Remark 1.5.1 (Parameters given by oracle). *The constraint $b < \tau$ in 1.5.1 means that the density*

cannot exceed $\frac{1}{b}$; we say that $\frac{1}{b}$ is the maximum compressibility constant. This constant naturally arises in the Nobel-Abel equation of state (also known as the co-volume equation of state). The constant q is a reference specific internal energy and $-p_\infty$ is a reference pressure. The reference pressure serves as a global lower bound, i.e.

$$0 < p(\mathbf{u}) + p_\infty < \infty, \quad \forall \mathbf{u} \in \mathcal{B}(b, q, p_\infty). \quad (1.5.2)$$

The notion of reference pressure naturally arises in the stiffened gas equation of state and the so-called Noble-Abel Stiffened-Gas equation of state; see [66]. One can take $b = 0$, $q = 0$, $p_\infty = 0$ if one does not have any *a priori* knowledge on b , q , and p_∞ . Notice that $\mathcal{B}(b, q, p_\infty)$ is a convex subset of \mathbb{R}^{d+2} .

1.5.1 Equations of State

Physically, the EOS encapsulates the response of a fluid to compression or expansion, heating or cooling, and provides the means to compute pressure from other state variables—typically density and internal energy in the context of compressible flow simulations. Without such a relation, the governing equations would remain underdetermined.

The theoretical development of equations of state (EOS) extends well before the advent of modern computational fluid dynamics. While the ideal gas law provided a foundational model for thermodynamic behavior, it proved insufficient for accurately describing dense fluids, high-pressure environments, or materials subjected to extreme conditions. The need for more comprehensive EOS models became particularly acute during the early 20th century, culminating in significant advances during the era surrounding World War II. In particular, as noted in [23] the scientists involved in the Manhattan Project played a pivotal role in extending the theoretical framework of EOS beyond idealized assumptions, developing models that could support the analysis of high-energy and shock-driven phenomena.

Many of the mathematical and physical principles that underpin modern EOS formulations were, however, established even earlier, during the late 19th and early 20th centuries. Notable

contributions came from a range of pioneering scientists: Rankine’s early thermodynamic theory of vaporization [73]; Van der Waals’ correction to the ideal gas law accounting for molecular size and intermolecular forces [81]; Hugoniot’s formulation of the conservation laws across shock waves [52]; Bethe’s work on the stability of shock waves for arbitrary EOS [4], and foundational work in statistical mechanics and quantum theory by Einstein, Lindemann, Debye, Grüneisen, Thomas, and Fermi. These researchers laid the groundwork for the sophisticated EOS models used today in high-energy-density physics, astrophysics, detonation theory, and material science.

The choice of EOS depends strongly on the fluid under consideration and the regime in which it operates. For example, an ideal gas EOS may suffice for high-temperature gases at low densities, where intermolecular interactions are negligible. In contrast, more sophisticated models, such as the Van der Waals EOS or the Noble-Abel Stiffened-Gas EOS, are necessary to account for non-ideal effects like molecular volume exclusion or intermolecular forces—particularly relevant in high-pressure or multiphase flow environments.

There are hundreds of different EOS and proving invariant-domain preserving properties is non-trivial for each of these equations. Furthermore, in practice it is often the case that the thermodynamics of a simulation cannot be adequately modeled by an analytical function. In this case, all that becomes available is experimental, tabulated data from which the idea of a tabulated equation of state is derived. In this case, there is no analytical expression that closes the system of equations, but rather experimental data in a table. Invariant-domain preserving properties can still be proven for tabulated EOS, see for example [19].

1.5.2 Ideal Gas Law

While real gases are influenced by factors like intermolecular forces and molecular volume, early researchers developed a simplified model—the ideal gas law—by neglecting these complexities. This model assumes that gas particles experience no interactions and occupy no volume, allowing for tractable analysis based on the principles of kinetic-molecular theory. Though ideal gases do not exist in reality, the ideal gas law serves as a foundational tool for understanding gas behavior and provides a baseline from which more accurate, non-ideal models are developed.

The ideal gas law is given as

$$pV = nRT, \quad (1.5.3)$$

where p is the pressure, V is the volume, n is the number of moles, R is the ideal gas constant, and T is the temperature. From the ideal gas law, simpler gas laws can be derived, such as Boyle's law, Charles's law, and Avogadro's law. While this equation is useful for many applications, we seek a form that more clearly will close our system; one that relates the pressure to our quantities of interest: density ρ and specific internal energy e . To derive this form, we first note that the number of moles n can be expressed in terms of the density ρ and the molar mass M as

$$n = \frac{\rho V}{M}.$$

Substituting this into the ideal gas law and eliminating V , we have

$$p = \frac{\rho}{M} RT.$$

Define the specific gas constant $R^* = \frac{R}{M}$. By substitution of R^* , we obtain

$$p = \frac{\rho}{M} RT = \rho R^* T.$$

For a calorically perfect gas, meaning the specific heats c_p and c_v are constant and independent of temperature, the specific internal energy e and enthalpy h are given by

$$e = c_v T \quad h = c_p T, \quad (1.5.4)$$

where c_v is the specific heat at constant volume and c_p is the specific heat at constant pressure. Since the enthalpy is defined as $h = e + p\tau$, we have the relationship

$$c_p T = c_v T + R^* T,$$

from which we get $R^* = c_p - c_v$. Substituting this and equation (1.5.4) into the equation for pressure, we have

$$p = \rho (c_p - c_v) \frac{e}{c_v}.$$

Since $\gamma := \frac{c_p}{c_v}$, we have

$$p(\rho, e) = (\gamma - 1) \rho e. \quad (1.5.5)$$

This equation relates the pressure of an ideal gas to its density and specific internal energy, allowing us to close our system of equations. The ideal gas law is a fundamental equation in thermodynamics, providing a simplified model for the behavior of gases under various conditions. It serves as a starting point for understanding more complex gas behaviors and is widely used in both theoretical and practical applications.

1.5.3 Van der Waals

Motivated by the inability of ideal gas to handle inelastic collisions between particles, Johannes D. van der Waals in 1873 proposed this EOS [81] to take into account molecular size and molecular interaction forces. The van der Waals equation of state is given as

$$\left(p + a \frac{n^2}{V^2} \right) (V - nb) = nRT, \quad (1.5.6)$$

where p is the pressure, V is the volume, n is the number of moles, R is the ideal gas constant, T is the temperature, a is a material dependent constant describing the attractive forces of the fluid and b represents the maximum compression of the fluid. In other words, this equation of state imposes a restriction that the specific volume satisfies $\tau > b$. The term $a \frac{n^2}{V^2}$ corrects for the attractive forces between molecules. Since these forces effectively reduce the pressure, this term is added to the pressure to account for that. The term nb corrects for the volume occupied by the molecules themselves. Since this volume cannot be compressed, it is subtracted from the total volume to account for that.

To get the mass specific form of the van der Waals equation, we first note that the relation

between the specific volume τ and the number of moles n , the molar mass M , and volume V is given by

$$\tau = \frac{V}{nM}.$$

Beginning with equation (1.5.6), we divide both sides by nM to obtain

$$\left(p + a \frac{n^2}{V^2} \right) \left(\frac{V}{nM} - \frac{b}{M} \right) = \frac{R}{M} T.$$

Substituting in the specific volume and using the relationship $R^* = \frac{R}{M}$, we have

$$\left(p + \frac{a}{M^2} \frac{1}{\tau^2} \right) \left(\tau - \frac{b}{M} \right) = R^* T.$$

By defining the rescaled constants $a^* := \frac{a}{M^2}$ and $b^* := \frac{b}{M}$, we can rewrite the equation as

$$\left(p + a^* \frac{1}{\tau^2} \right) (\tau - b^*) = R^* T.$$

Solving for the pressure and omitting the superscript * on the attraction and maximum compressibility constants, we have

$$p(\tau, T) = \frac{R^* T}{\tau - b} - \frac{a}{\tau^2}. \quad (1.5.7)$$

It remains to be shown how to express the pressure in terms of the specific internal energy e . Recall in the ideal gas case we had a relationship to express the temperature in terms of the specific internal energy. In [10, Ch. 3, p. 76], Herbert Callen suggests that the simplest extrapolation of this assumption that remains consistent with the van der Waals equation is

$$T = \frac{(\gamma - 1)}{R^*} \left(e + \frac{a}{\tau} \right). \quad (1.5.8)$$

With this choice, we can substitute this into equation (1.5.7) and replace $\rho = \frac{1}{\tau}$ to obtain

$$p(\rho, e) = (\gamma - 1) \frac{\rho e + a\rho^2}{1 - b\rho} - a\rho^2, \quad (1.5.9)$$

1.5.4 Noble-Abel Stiffened Gas

In [69], the compatibility of NASG with Maxwell's rules is examined. See [25] for the thermal version, which is given by

$$p(\tau, T) = \frac{(\gamma - 1) c_v T}{\tau - b} - p_\infty.$$

The caloric Noble-Abel Stiffened Gas equation of state was first introduced by Le Métayer & Saurel in [66]. This equation of state is a combination of the Stiffened gas model [47, Ch. II, Sec. B] and the Noble-Abel model [49], and is given by

$$p(\rho, e) = (\gamma - 1) \frac{e - q}{\tau - b} - \gamma p_\infty, \quad (1.5.10)$$

where p , τ , e , and q are respectively the pressure, specific volume, specific internal energy, and the reference specific internal energy.

In the case that $b = 0$, the Stiffened Gas equation of state is recovered. Furthermore, if $b = q = p_\infty = 0$ the ideal gas law is recovered. The parameters γ , p_∞ , q , and b are material constants representing the ratio of specific heats, the pressure at infinite density, the reference specific internal energy, and the covolume of the fluid, and describe the material being modeled.

Table 1.1: Reference equations of state in caloric form $p = p(\rho, e)$.

Equations of state in caloric form	
Ideal gas	$p(\rho, e) = (\gamma - 1) \rho e$
van der Waals	$p(\rho, e) = (\gamma - 1) \frac{\rho e + a\rho^2}{1 - b\rho} - a\rho^2$
Noble-Abel Stiffened Gas	$p(\rho, e) = (\gamma - 1) \frac{e - q}{\tau - b} - \gamma p_\infty$

Table 1.2: Reference speeds of sound for various equations of state.

Speed of sound in $a = a(\rho, p)$ form	
Ideal gas	$a^2 = \frac{\gamma p}{\rho}$
van der Waals	$a^2 = \gamma \frac{p+a\rho^2}{\rho(1-b\rho)} - 2a\rho$
Noble-Abel Stiffened Gas	$a^2 = \frac{\gamma(p+p_\infty)}{\rho(1-b\rho)}$

2. FINITE ELEMENT APPROXIMATION OF THE EULER EQUATIONS IN A LAGRANGIAN FRAME

We are going to use the weak form of (1.4.16) to construct the space and time approximation. We assume that there is $t^* > 0$ such that the mapping $\Phi_t : D^0 \ni \xi \mapsto \Phi_t(\xi) := \Phi(\xi, t) \in \mathbb{R}^d$ is invertible for all $t \in [0, t^*]$ and smooth enough so that everything that is said below makes some sense. Being precise on the expected smoothness of Φ_t is an outstanding question that is well beyond the scope of this thesis as the vector field v appearing in (1.4.16b) solves the incompressible Euler equations. To simplify some expressions, we are going to use the notation $\Phi(\xi, t)$ when the dependence with respect to t is invoked and the notation $\Phi_t(\xi)$ when t is fixed. Note that $\Phi_0(\xi) = \xi$ and $v(\Phi(\xi, t), t) = \partial_t \Phi(\xi, t)$, for all $(\xi, t) \in D^0 \times [0, t^*]$. We recall that $\rho := \frac{1}{\tau}$ is the density. Then using the Reynolds transport theorem (1.4.5), the Lagrangian weak formulation of the compressible Euler equations consists of seeking $u := (\tau, v, E)^\top$ so that the following holds in the weak sense in time and for all test function $\psi \in L^\infty(D_0)$ (see e.g., [14, Eq. (1)], [26, Eq. (5.29)], [84, Eq. (1)]):

$$\partial_t \int_{D_t} \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} = 0, \quad (2.0.1a)$$

$$\partial_t \int_{D_t} \tau(\mathbf{x}, t) \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} = \int_{D_t} \nabla \cdot [v(\mathbf{x}, t)] \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} \quad (2.0.1b)$$

$$\partial_t \int_{D_t} v(\mathbf{x}, t) \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} = \int_{D_t} -\nabla [p(u(\mathbf{x}, t))] \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} \quad (2.0.1c)$$

$$\partial_t \int_{D_t} E(\mathbf{x}, t) \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} = \int_{D_t} -\nabla \cdot [v(\mathbf{x}, t) p(u(\mathbf{x}, t))] \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x}. \quad (2.0.1d)$$

The equation (2.0.1b) is a reformulation of the definition of the Lagrangian velocity (1.4.16b). The equations (2.0.1a)-(2.0.1c)-(2.0.1d) express the conservation of mass, momentum and energy, respectively.

Remark 2.0.1 (Geometric Conservation Law (GCL)). *Equation (2.0.1b) is known as the Geometric Conservation Law and governs the time rate of change of a cell volume. This means that the*

volume from the updated node positions must agree with that obtained by discretizing the specific volume equation, requiring a compatibility between the mesh motion and the update to the specific volume. Our mesh motion is defined to ensure this compatibility, and is further explored in section 3.4 (specifically, see equation (3.4.6)).

2.1 Finite Element Representation

We introduce here the finite element setting used by the low-order scheme. Let $Q_{k,d}$ denote the space of d -variate polynomials of partial degree at most k . We approximate the mesh motion (kinematics) with continuous high-order Lagrange elements, and we approximate the state variable $\mathbf{u} = (\tau, \mathbf{v}, E)$ with piecewise constants (dG0).

Remark 2.1.1. *The node-based mesh velocity is required to be defined on a continuous finite element space for our mesh motion to be well defined at each discrete point in time. This, however, is not the case for the thermodynamic space for which we employ a discontinuous approximation. dG preserves sharp interfaces and prevents unphysical mixing across materials. This choice also reflects the locality of the equation of state and the invariance of cell mass in a Lagrangian description.*

2.1.1 Meshes

Let $\{\mathcal{T}_h^0\}_{h \in \mathcal{H}}$ be a shape-regular sequence of matching meshes on the initial domain D^0 , meaning that the intersection of two distinct cells is either empty, a vertex, a full edge, or (in 3D) a full face. Cells are simplices (triangles/tetrahedra) or parallelotopes (quadrilaterals/hexahedra). The superscript 0 refers to the initial configuration; \mathcal{T}_h^0 covers D^0 exactly.

The meshes deform over time under the finite element approximation of the velocity field. We denote by \mathcal{T}_h^n the mesh at time t^n (with $t^0 := 0$), and enumerate its cells by an index set \mathcal{N}^{Cel} , so that $\mathcal{T}_h^n := \{K_c^n\}_{c \in \mathcal{N}^{\text{Cel}}}$. Then $\bigcup_{c \in \mathcal{N}^{\text{Cel}}} K_c = D$. All cells are generated from a fixed reference element $\widehat{K} \subset \mathbb{R}^d$ (simplex, square, or cube). We enumerate the faces of \widehat{K} by $\widehat{\mathcal{N}}_{\text{fac}}$ and write $\widehat{\mathcal{F}}(\widehat{K}) := \{\widehat{F}_m\}_{m \in \widehat{\mathcal{N}}_{\text{fac}}}$ for the set of its faces.

Definition 7 (Face orientation, traces, average, and jump). *Let $F = \partial K_\ell \cap \partial K_r$ be an interior face oriented by the unit normal \mathbf{n}_F pointing from K_ℓ (left) to K_r (right), so that $\mathbf{n}_{K_\ell} = \mathbf{n}_F = -\mathbf{n}_{K_r}$ on F . For a quantity q with traces q^ℓ, q^r on F , define the **average** and **jump***

$$\{q\}_F := \frac{1}{2}(q^\ell + q^r), \quad [q]_F := q^r - q^\ell.$$

For vectors \mathbf{w} , we use the **normal jump** $[\![\mathbf{w}]\!]_F \cdot \mathbf{n}_F = \mathbf{w}^r \cdot \mathbf{n}_F - \mathbf{w}^\ell \cdot \mathbf{n}_F$. On boundary faces $F \subset \partial D$, take \mathbf{n}_F as the outward normal and set $\{q\}_F := q$, $[q]_F := q$ (and $[\![\mathbf{w}]\!]_F \cdot \mathbf{n}_F := \mathbf{w} \cdot \mathbf{n}_F$) against the boundary data.

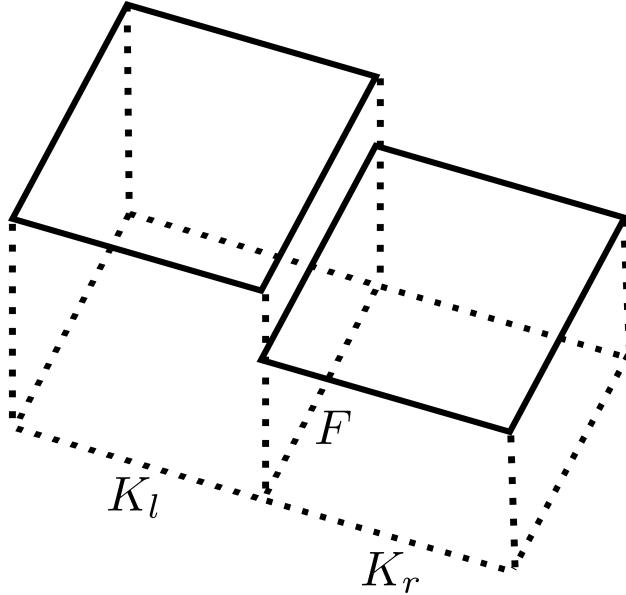


Figure 2.1: Jump of a piecewise constant function across the interface $F = K_l \cap K_r$.

2.1.2 Finite Element Approximation of the State Variable

We approximate the state variable $\mathbf{u} = (\tau, \mathbf{v}, E)$ with piecewise constants (cell-centered unknowns). For each $c \in \mathcal{N}^{\text{Cel}}$ and $n \geq 0$, let $\mathbb{1}_{K_c^n}$ be the indicator of the cell $K_c^n \in \mathcal{T}_h^n$. Define

$$P_h(\mathcal{T}_h^n) := \text{span}\{\mathbb{1}_{K_c^n}\}_{c \in \mathcal{N}^{\text{Cel}}}, \quad \mathbf{P}_h(\mathcal{T}_h^n) := [P_h(\mathcal{T}_h^n)]^{d+2}. \quad (2.1.1)$$

Notice that the dG space defined above is nonconforming since the functions can be discontinuous across interfaces and take nonzero values on the boundary (see Figure 2.1). The approximation of \mathbf{u} at time $t^n \in [0, t^*]$ is

$$\mathbf{u}_h^n := \sum_{c \in \mathcal{N}^{\text{Cel}}} \mathbf{U}_c^n \mathbb{1}_{K_c^n}, \quad \mathbf{U}_c^n = (T_c^n, \mathbf{V}_c^n, E_c^n)^\top, \quad (2.1.2)$$

where T_c^n is the cellwise specific volume, \mathbf{V}_c^n the cellwise velocity, and E_c^n the cellwise specific total energy.

Remark 2.1.2 (Nonconformity of the dG space). *Functions in the discontinuous Galerkin space $\mathbf{P}_h \not\subset H_0^1(D)$ may exhibit jumps across inter-element faces and may take nonzero values on ∂D . In contrast, membership in H_0^1 requires single-valued (continuous) traces across all interfaces and a zero boundary trace.*

2.1.3 Finite Element Approximation of the Mesh Motion

We use H^1 -conforming Lagrange finite elements to represent the mesh motion. Following Ciarlet's notation (see [18, p. 84]), let $(\hat{K}, \hat{P}^{\text{geo}}, \hat{\Sigma}^{\text{geo}})$ be the reference Lagrange geometric element. The purpose of the geometric reference element is to construct the geometric mapping $\mathbf{T}_K : \hat{K} \rightarrow K$ for all $K \in \mathcal{T}_h^n$ and represent the velocity field that is used for the mesh motion. The Lagrange nodes of the reference finite element are enumerated with the index set $\hat{\mathcal{N}}^{\text{geo}}$. We denote $\{\hat{\mathbf{a}}_\ell\}_{\ell \in \hat{\mathcal{N}}^{\text{geo}}}$ the collection of Lagrange nodes and $\{\hat{\theta}_\ell\}_{\ell \in \hat{\mathcal{N}}^{\text{geo}}}$ the Lagrange shape functions. We will also refer to $\{\hat{\mathbf{a}}_\ell\}_{\ell \in \hat{\mathcal{N}}^{\text{geo}}}$ as geometric nodes. Recall that by definition $\hat{\theta}_\ell(\hat{\mathbf{a}}_\ell) = \delta_{\ell\ell}$. Let $\{\mathbf{a}_i^n\}_{i \in \mathcal{N}^{\text{geo}}}$ be the global geometric nodes of the mesh at t^n . With the usual connectivity map $j^{\text{geo}} : \hat{\mathcal{N}}^{\text{geo}} \times \mathcal{N}^{\text{Cel}} \rightarrow \mathcal{N}^{\text{geo}}$, the geometric mapping for a cell K_c^n is

$$T_{K_c^n}(\hat{\mathbf{x}}) := \sum_{\ell \in \hat{\mathcal{N}}^{\text{geo}}} \mathbf{a}_{j^{\text{geo}}(\ell, c)}^n \hat{\theta}_\ell(\hat{\mathbf{x}}). \quad (2.1.3)$$

We assume the degrees of freedom are enumerated so that $T_{K_c^0}$ is invertible and that $T_{K_c^n}$ remains invertible for all cells $K_c^n \in \mathcal{T}_h^n$.

Define the scalar geometric space

$$P_h^{\text{geo}}(\mathcal{T}_h^n) := \{v_h \in C^0(D^n; \mathbb{R}) \mid v_h \circ T_K^{-1} \in \hat{P}^{\text{geo}}, \forall K \in \mathcal{T}_h^n\}. \quad (2.1.4)$$

A basis for $P_h^{\text{geo}}(\mathcal{T}_h^n)$ is given by $\text{span } \{\theta_i^n\}_{i \in \mathcal{N}^{\text{geo}}}$, where

$$\theta_i^n(\mathbf{a}) := \begin{cases} \widehat{\theta}_{(j^{\text{geo}})^{-1}(i)}(T_K^{-1}(\mathbf{a})), & \text{if } \mathbf{a} \in K \in \mathcal{T}_h^n(i) \\ 0, & \text{otherwise.} \end{cases} \quad (2.1.5)$$

Here, $\mathcal{T}_h^n(i) \subset \mathcal{T}_h^n$ denotes the set of cells $K \in \mathcal{T}_h^n$ that contain the vertex \mathbf{a}_i . Notice that $(j^{\text{geo}})^{-1}$

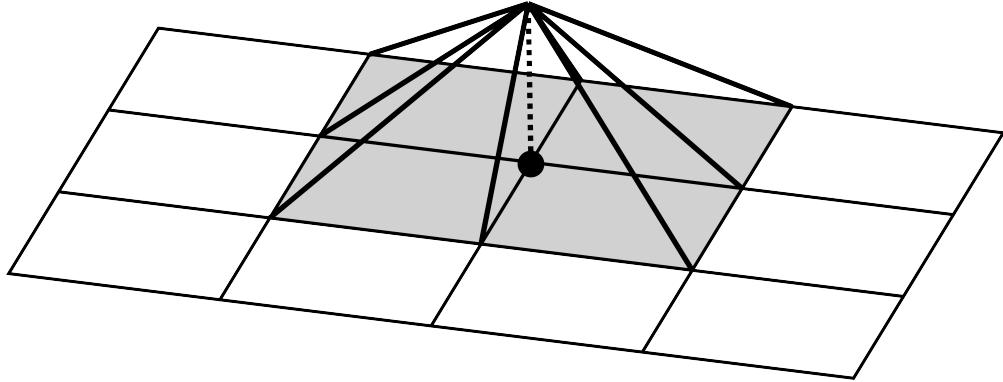


Figure 2.2: A typical Q_1 basis function.

is well defined since the element K is fixed. The θ_i are often referred to as *tent functions* based on their shape in two dimensions (see Figure 2.2). For all \mathbf{a} , each tent function θ_i^n satisfies $\theta_i^n(\mathbf{a}) \geq 0$, and the family $\{\theta_i^n\}_{i \in \mathcal{N}^{\text{geo}}}$ forms a partition of unity:

$$\sum_{i \in \mathcal{N}^{\text{geo}}} \theta_i^n(\mathbf{a}) = 1. \quad (2.1.6)$$

The mesh velocity field will be defined on the vector space $\mathbf{P}_h^{\text{geo}}(\mathcal{T}_h^n) := [P_h^{\text{geo}}(\mathcal{T}_h^n)]^d$. We denote this yet to be defined mesh velocity $\mathbf{v}_h(\mathbf{x}, t) \in \mathbf{P}_h^{\text{geo}}(\mathcal{T}_h^n)$, where $t \in [0, t^*]$, where $[0, t^*]$

is a time interval during which the mesh motion is well defined. This velocity will be defined to be piecewise constant in time. For a full discussion on the mesh velocity, see Section 3.4. We denote by Φ_h the geometric flow induced by this velocity, i.e., $\Phi_h(\xi, 0) = \xi$ and $\partial_t \Phi_h(\xi, t) = \mathbf{v}_h(\Phi_h(\xi, t), t)$ for $(\xi, t) \in D^0 \times [0, t^*]$. (For the coupled HO/LO algorithm in Chapter 4, the LO mesh shares the vertex DOFs with the HO mesh and moves by one-to-one vertex mapping, so the geometries coincide for all time.)

2.1.3.1 Density Representation (Strong Mass Conservation).

The counterpart of the continuous identity is as follows.

Lemma 2 (Density representation). *Let $\rho_h^{\text{Lag}}(\cdot, t) : D^t \rightarrow \mathbb{R}$ solve, in the weak sense, the conservation law $\partial_t \rho_h^{\text{Lag}} + \nabla \cdot (\mathbf{v}_h \rho_h^{\text{Lag}}) = 0$ with initial data ρ_0 . Then, for all $\psi \in L^\infty(D^0)$ and all $t \in [0, t^*]$,*

$$\int_{D^t} \rho_h^{\text{Lag}}(\mathbf{x}, t) \psi(\Phi_h^{-1}(\mathbf{x}, t)) d\mathbf{x} = \int_{D^0} \rho_0(\xi) \psi(\xi) d\xi. \quad (2.1.7)$$

The field ρ_h^{Lag} is used only for consistency arguments; it is not computed in practice.

2.2 The Fully Discrete Scheme

Let t^n be the current time stage and let Δt be the current time step (we should write Δt^n , but we omit the superscript n to simplify the notation). Let $\{\mathbf{U}_c^n\}_{c \in \mathcal{N}^{\text{Cel}}}$ be the piecewise constant approximation of the specific volume, velocity, and specific total energy at time t^n with $\mathbf{U}_c^n := (\mathbf{T}_c^n, \mathbf{V}_c^n, \mathbf{E}_c^n)^\top$. We explain here how to compute the update \mathbf{U}_c^{n+1} using the centered discontinuous Galerkin numerical flux.

Let $K_c^n \in \mathcal{T}_h^n$, $c \in \mathcal{N}^{\text{Cel}}$, be a cell and let $\mathcal{F}(K_c^n)$ be the collection of the faces of K_c^n . The collection of the internal faces is denoted $\mathcal{F}^\circ(K_c^n)$ and the collection of the boundary faces is denoted $\mathcal{F}^\partial(K_c^n)$. The outward unit normal over ∂K_c^n is denoted \mathbf{n}_c . We introduce the index sets

$$\mathcal{I}(c) := \{c' \in \mathcal{N}^{\text{Cel}} \mid K_c^n \cap K_{c'}^n \in \mathcal{F}(K_c^n)\}, \quad \text{and} \quad \mathcal{I}^*(c) := \mathcal{I}(c) \setminus \{c\}, \quad (2.2.1)$$

and we call the set $\mathcal{I}(c)$ stencil of the cell K_c^n . (This set does not depend on n .)

Let m_c be the mass contained in the cell K_c^0 at the initial time. Let $|K_c^0|$ be the measure of K_c^0 and let T_c^0 be the approximation of the specific volume. Then

$$\frac{|K_c^0|}{T_c^0} := m_c := \int_{K_c^0} \rho_0(\zeta) d\zeta. \quad (2.2.2)$$

Although we do not know how to compute the field ρ_h^{Lag} induced by the mesh motion v_h (see Lemma 2), the identity (2.1.7) with $\psi(\xi) = \mathbb{1}_{K_c^0}$ implies that

$$m_c = \dots = \int_{K_c^n} \rho_h^{\text{Lag}}(\mathbf{x}, t^n) dx = \int_{K_c^{n+1}} \rho_h^{\text{Lag}}(\mathbf{x}, t^{n+1}) dx. \quad (2.2.3)$$

We approximate the flux $\mathbf{f}(\mathbf{u}_h^n)$ by projecting it onto the discrete finite element space. We define this projection $\Pi_h : C^0(\mathbb{R}^{d+2}; \mathbb{R}^{(d+2) \times d}) \rightarrow [P_h(\mathcal{T}_h^n)]^{(d+2) \times d}$ by

$$\mathbf{f}(\mathbf{u}_h^n) \approx \Pi_h \mathbf{f}(\mathbf{u}_h^n) = \sum_{c \in \mathcal{N}^{\text{geo}}} \mathbf{f}(\mathbf{U}_c^n) \mathbb{1}_{K_c^n}. \quad (2.2.4)$$

Combining this with the forward Euler method, we have the following numerical method: find \mathbf{u}_h^{n+1} satisfying

$$\rho_h^n \frac{\mathbf{u}_h^{n+1} - \mathbf{u}_h^n}{\Delta t} + \nabla \cdot (\Pi_h \mathbf{f}(\mathbf{u}_h^n)) = \mathbf{0}. \quad (2.2.5)$$

Let $\psi(\xi) = \mathbb{1}_{K_c^0}(\xi)$ and $K_c^n := \Phi_h(K_c^0, t^n)$ in (2.0.1b)-(2.0.1c)-(2.0.1d). Assuming for the time being that the mesh velocity v_h is a consistent approximation of the velocity v that solves the Euler equation, it is legitimate to postulate that ρ_h^{Lag} approximate the density. Then, we observe

$$0 = \partial_t \int_{D_t} \rho(\mathbf{x}, t) \mathbf{u}(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) dx - \int_{D_t} [\nabla \cdot \mathbf{f}(\mathbf{u})](\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) dx \quad (2.2.6)$$

$$= \partial_t \int_{D_t} \rho(\mathbf{x}, t) \mathbf{u}(\mathbf{x}, t) \mathbb{1}_{K_c^n}(\mathbf{x}) dx - \int_{D_t} [\nabla \cdot \mathbf{f}(\mathbf{u})](\mathbf{x}, t) \mathbb{1}_{K_c^n}(\mathbf{x}) dx \quad (2.2.7)$$

$$= \partial_t \int_{K_c^n} \rho(\mathbf{x}, t) \mathbf{u}(\mathbf{x}, t) dx - \int_{K_c^n} [\nabla \cdot \mathbf{f}(\mathbf{u})](\mathbf{x}, t) dx \quad (2.2.8)$$

Then using (2.2.3) and integrating by parts, (2.2.8) is approximated by

$$\approx \frac{1}{\Delta t} \left[\mathbf{U}_c^{n+1} \int_{K_c^{n+1}} \rho_h^{\text{Lag}}(\mathbf{x}, t^{n+1}) dx - \mathbf{U}_c^n \int_{K_c^n} \rho_h^{\text{Lag}}(\mathbf{x}, t^n) dx \right] - \int_{\partial K_c^n} \mathbf{f}(\mathbf{u}) \cdot \mathbf{n}_c ds \quad (2.2.9)$$

$$= \frac{m_c}{\Delta t} (\mathbf{U}_c^{n+1} - \mathbf{U}_c^n) - \sum_{F \in \mathcal{F}(K_c^n)} \hat{\mathbf{f}}_F \cdot \int_F \mathbf{n}_c ds \quad (2.2.10)$$

where \mathbf{n}_c is the outward unit normal on $\partial K_c(t)$, $\mathbf{U} = (\tau, \mathbf{v}, E)$, and the centered numerical flux is defined by

$$\hat{\mathbf{f}}_F = \begin{cases} \frac{1}{2}(\mathbf{f}(\mathbf{U}_c) + \mathbf{f}(\mathbf{U}_{c'})) & \text{if } F = K_c^n \cap K_{c'}^n \in \mathcal{F}^\circ(K_c^n), \\ \mathbf{f}(\mathbf{U}_c) & \text{if } F \in \mathcal{F}^\partial(K_c^n), \end{cases} \quad (2.2.11)$$

where we recall that \mathbf{f} is defined in equation (1.4.16c).

3. THE FIRST ORDER APPROXIMATION*

In this chapter, we present the first-order scheme and establish the invariant-domain preserving (IDP) property for general equations of state.

3.1 Stabilized Update

For convenience, we introduce the geometric quantities

$$\mathbf{c}_{cc'} := \frac{1}{2} \int_{F_{cc'}} \mathbf{n}_c ds, \quad F_{cc'} = K_c^n \cap K_{c'}^n, \quad c' \in \mathcal{I}^*(c), \quad (3.1.1a)$$

$$\mathbf{c}_{cF}^\partial := \frac{1}{2} \int_F \mathbf{n}_c ds, \quad F \subset \mathcal{F}^\partial(K_c^n), \quad (3.1.1b)$$

so that (2.2.10) can be written compactly as a sum over the stencil $\mathcal{I}^*(c)$ of K_c^n . Note that we will use the ^L superscript to indicate this is the low order update. This will help in the future chapters of this thesis to distinguish between the high order update. Then the system (2.2.10) can be re-written

$$\frac{\mathbf{m}_c}{\Delta t} \mathbf{U}_c^{L,n+1} = \frac{\mathbf{m}_c}{\Delta t} \mathbf{U}_c^n + \sum_{c' \in \mathcal{I}^*(c)} -(\mathbf{f}(\mathbf{U}_{c'}^n) + \mathbf{f}(\mathbf{U}_c^n)) \mathbf{c}_{cc'} + \sum_{F \in \mathcal{F}^\partial(K_c^n)} -2\mathbf{f}(\mathbf{U}_c^n) \mathbf{c}_{cF}^\partial. \quad (3.1.2)$$

Remark 3.1.1 (Consequence of Divergence Theorem). *For any closed volume $V \subset \mathbb{R}^d$ with closed surface $\partial V \subset \mathbb{R}^{d-1}$, the integral of the outward normal vector $\mathbf{n} \in \mathbb{R}^d$ over the boundary is always 0. Indeed, by the divergence theorem we have*

$$\int_{\partial V} \mathbf{1} \cdot \mathbf{n} dS = \int_V \nabla \cdot \mathbf{1} d\mathbf{x} = 0.$$

In our context that means definitions (3.1.1) satisfy $\forall c \in \mathcal{N}^{\text{Cel}}$

$$\sum_{c' \in \mathcal{I}^*(c)} \mathbf{c}_{cc'} + \sum_{F \in \mathcal{F}^\partial(K)} \mathbf{c}_{cF}^\partial = \mathbf{0} \quad (3.1.3)$$

*A majority of this chapter is a modification of the work done in [45] and is reprinted with permission from the authors.

Definition 8 (Exact Lagrangian mass conservation). *We say that there is exact Lagrangian mass conservation when the following identity holds for all $n \geq 0$ and all cells $c \in \mathcal{N}^{\text{Cel}}$:*

$$\mathbf{m}_c := \frac{|K_c^0|}{\mathbb{T}_c^0} = \frac{|K_c^n|}{\mathbb{T}_c^n}. \quad (3.1.4)$$

Remark 3.1.2 (Lagrangian mass conservation). *Notice that so far there is no mechanism in (3.1.2) to ensure exact Lagrangian mass conservation. This question is addressed later in §3.5.*

Remark 3.1.3 (Boundary conditions). *The numerical flux at the boundary depends on the boundary conditions that one wants to enforce. In the majority of the tests presented in Chapter 6, we use the “do nothing” fluxes in (2.2.11) just to simplify the presentation. One specific test case in which the pressure is imposed at the boundaries can be found in the case of the Kidder problem, further discussed in section 6.2.5. Other boundary conditions can be applied.*

To obtain an invariant-domain preserving scheme, we augment (3.1.2) with graph viscosity. Introduce nonnegative coefficients $d_{cc'} \geq 0$ for all $c' \in \mathcal{I}^*(c)$ later to be specified in §3.2. Then, the stabilized update is defined by:

$$\begin{aligned} \frac{\mathbf{m}_c}{\Delta t} \mathbf{U}_c^{L,n+1} &= \frac{\mathbf{m}_c}{\Delta t} \mathbf{U}_c^n + \sum_{c' \in \mathcal{I}^*(c)} -(\mathbf{f}(\mathbf{U}_{c'}^n) + \mathbf{f}(\mathbf{U}_c^n)) \mathbf{c}_{cc'} \\ &\quad + \sum_{F \in \mathcal{F}^\partial(K_c^n)} -2\mathbf{f}(\mathbf{U}_c^n) \mathbf{c}_{cF}^\partial + \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n (\mathbf{U}_{c'}^n - \mathbf{U}_c^n). \end{aligned} \quad (3.1.5)$$

Remark 3.1.4 (Conservation). *Notice that the unstabilized scheme given in equation (3.1.2) is by itself conservative, meaning $\sum_{c \in \mathcal{N}^{\text{Cel}}} \mathbf{m}_c \mathbf{U}_c^{n+1} = \sum_{c \in \mathcal{N}^{\text{Cel}}} \mathbf{m}_c \mathbf{U}_c^n$. This is a direct consequence of (3.1.3).*

Remark 3.1.5 (Galilean Invariance). *The low order update presented in (3.1.5) is Galilean invariant. For further discussion on Galilean invariance and a proof, see section A.3.*

3.2 Graph Viscosity

We now explain how to define the graph viscosity $d_{cc'}^n$ to make the method invariant-domain preserving; see (3.2.6). Let $L \in \mathcal{N}^{\text{Cel}}$ and $R \in \mathcal{I}^*(L)$ be two cells (the letter R will later stand for “right” and the letter L will stand for “left”). To be general, we use the notation $\mathbf{u}_Z := (\tau_Z, \mathbf{v}_Z, E_Z)$ with $Z \in \{L, R\}$ instead of $(\mathbf{T}_L^n, \mathbf{V}_L^n, \mathbf{E}_L^n)$ and $(\mathbf{T}_R^n, \mathbf{V}_R^n, \mathbf{E}_R^n)$. We assume by induction on the time index n that $(\tau_Z, \mathbf{v}_Z, E_Z)$ is in $\mathcal{B}(b, q, p_\infty)$. We also set $p_Z := p(\mathbf{u}_Z)$ where $p : \mathcal{B}(b, q, p_\infty) \rightarrow (-p_\infty, \infty)$ is the pressure oracle. Recall that we assumed that the pressure oracle satisfies (1.5.2). In summary, we assume that the following holds for all $Z \in \{L, R\}$:

$$\mathbf{u}_Z \in \mathcal{B}(b, q, p_\infty), \quad (3.2.1a)$$

$$0 < p_Z + p_\infty. \quad (3.2.1b)$$

We are going to define the graph viscosity d_{LR}^n by estimating from above the maximum wave speed in a Riemann problem with left data $(\tau_L, \mathbf{v}_L, E_L)$ and right data $(\tau_R, \mathbf{v}_R, E_R)$.

In order to define a Riemann problem that will allow us to handle tabulated equations of state, we proceed as in [19, 20] and introduce a new scalar variable γ . With this new variable, we define an extended state, an extended flux, and an extended pressure as follows:

$$\tilde{\mathbf{u}} := (\tau, \mathbf{v}, E, \gamma)^\top \quad \tilde{\mathbb{f}}(\tilde{\mathbf{u}}) := (-\mathbf{v}, \tilde{p}(\tilde{\mathbf{u}})\mathbb{I}_d, \mathbf{v}\tilde{p}(\tilde{\mathbf{u}}), \mathbf{0})^\top, \quad (3.2.2)$$

$$\tilde{p}(\tilde{\mathbf{u}}) := (\gamma - 1) \frac{\tilde{e}(\tilde{\mathbf{u}}) - q}{\tau - b} - \gamma p_\infty, \quad \text{where} \quad \tilde{e}(\tilde{\mathbf{u}}) := e(\mathbf{u}) = E - \frac{1}{2} \|\mathbf{v}\|_{\ell^2}^2. \quad (3.2.3)$$

After defining the unit vector $\mathbf{n} := \mathbf{c}_{cc'}/\|\mathbf{c}_{cc'}\|_{\ell^2}$, we consider the extended Riemann problem consisting of seeking $\tilde{\mathbf{u}} := (\tau, \mathbf{v}, E, \gamma)^\top : \mathbb{R} \times (0, \infty) \rightarrow \mathbb{R}^{d+3}$ so that

$$\partial_t \tilde{\mathbf{u}} + \partial_x (\tilde{\mathbb{f}}(\tilde{\mathbf{u}}) \mathbf{n}) = \mathbf{0}, \quad \tilde{\mathbf{u}}(x, 0) = \begin{cases} \tilde{\mathbf{u}}_L & \text{if } x \leq 0 \\ \tilde{\mathbf{u}}_R & \text{if } x > 0, \end{cases} \quad (3.2.4)$$

with left and right data $\tilde{\mathbf{u}}_Z := (\tau_Z, \mathbf{v}_Z, E_Z, \gamma_Z)^\top$, where $Z \in \{L, R\}$. Using $e_Z := e(\mathbf{u}_Z)$, the scalar γ_Z is defined so that $\tilde{p}(\tilde{\mathbf{u}}_Z) = p_Z$, i.e.,

$$\gamma_Z := 1 + \frac{p_Z + p_\infty}{\frac{e_Z - q}{\tau_Z - b} - p_\infty}. \quad (3.2.5)$$

Notice that this definition makes sense owing to (3.2.1), i.e., $\frac{e_Z - q}{\tau_Z - b} - p_\infty \neq 0$. Note also that (3.2.1) implies $\gamma_Z > 1$. Let $\hat{\lambda}(\mathbf{n}, \tilde{\mathbf{u}}_L, \tilde{\mathbf{u}}_R)$ be any upper bound on $\lambda_{\max}(\mathbf{n}, \tilde{\mathbf{u}}_L, \tilde{\mathbf{u}}_R)$, then we set

$$d_{cc'}^n := \max \left(\hat{\lambda}_{\max}(\mathbf{n}, \tilde{\mathbf{u}}_c, \tilde{\mathbf{u}}_{c'}) \|\mathbf{c}_{cc'}\|_{\ell^2}, \hat{\lambda}_{\max}(-\mathbf{n}, \tilde{\mathbf{u}}_{c'}, \tilde{\mathbf{u}}_c) \|\mathbf{c}_{c'c}\|_{\ell^2} \right). \quad (3.2.6)$$

Remark 3.2.1 (Symmetry). *For all $F = K_c \cap K_{c'}$, we have $\mathbf{c}_{cc'} = -\mathbf{c}_{c'c}$ and $\lambda_{\max}(\mathbf{n}, \tilde{\mathbf{u}}_c, \tilde{\mathbf{u}}_{c'}) = \lambda_{\max}(-\mathbf{n}, \tilde{\mathbf{u}}_{c'}, \tilde{\mathbf{u}}_c)$. Hence, $d_{cc'}^n = d_{c'c}^n = \max \left(\hat{\lambda}_{\max}(\mathbf{n}, \tilde{\mathbf{u}}_c, \tilde{\mathbf{u}}_{c'}), \hat{\lambda}_{\max}(-\mathbf{n}, \tilde{\mathbf{u}}_{c'}, \tilde{\mathbf{u}}_c) \right) \|\mathbf{c}_{c'c}\|_{\ell^2}$.*

This then means that this chosen viscosity does not violate conservation of our scheme. Indeed by Fubini's theorem and this symmetry,

$$\begin{aligned} \sum_{c \in \mathcal{N}^{\text{Cel}}} \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n (\mathbf{U}_{c'}^n - \mathbf{U}_c^n) &= \sum_{c \in \mathcal{N}^{\text{Cel}}} \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n \mathbf{U}_{c'}^n - \sum_{c \in \mathcal{N}^{\text{Cel}}} \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n \mathbf{U}_c^n \\ &= \sum_{c' \in \mathcal{N}^{\text{Cel}}} \sum_{c \in \mathcal{I}^*(c)} d_{c'c}^n \mathbf{U}_{c'}^n - \sum_{c \in \mathcal{N}^{\text{Cel}}} \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n \mathbf{U}_c^n = 0, \end{aligned}$$

where the final equality follows since the indices are arbitrary.

3.3 Invariant-Domain Property

The goal of this section is to prove that the update defined in (3.1.5) is invariant-domain preserving. Using equation (3.1.3), we re-write (3.1.5) as follows

$$\mathbf{U}_c^{L,n+1} = \mathbf{U}_c^n + \frac{\Delta t}{m_c} \sum_{c' \in \mathcal{I}^*(c)} -(\mathbb{f}(\mathbf{U}_{c'}^n) - \mathbb{f}(\mathbf{U}_c^n)) \mathbf{c}_{cc'} + d_{cc'}^n (\mathbf{U}_{c'}^n - \mathbf{U}_c^n).$$

This then allows us to write the low order update as a summation of states

$$\mathbf{U}_c^{L,n+1} = \left(1 - \sum_{c' \in \mathcal{I}^*(c)} \frac{2\Delta t d_{cc'}^n}{m_c}\right) \mathbf{U}_c^n + \sum_{c' \in \mathcal{I}^*(c)} \frac{2\Delta t d_{cc'}^n}{m_c} \bar{\mathbf{U}}_{cc'} \left(\frac{\|\mathbf{c}_{cc'}\|_{\ell^2}}{2d_{cc'}^n}\right). \quad (3.3.1)$$

where the bar states are defined

$$\bar{\mathbf{U}}_{cc'}^n(t) := \frac{1}{2}(\mathbf{U}_c^n + \mathbf{U}_{c'}^n) - t(\mathbb{f}(\mathbf{U}_{c'}^n) - \mathbb{f}(\mathbf{U}_c^n))\mathbf{n}. \quad (3.3.2)$$

Theorem 3.3.1 (local invariance). *Let $c \in \mathcal{N}^{\text{Cel}}$. Let $\mathbf{U}_{c'}^n \in \mathcal{B}(b, q, p_\infty)$ for all $c' \in \mathcal{I}(c)$. Assume that the pressure oracle satisfies $0 < p(\mathbf{u}) + p_\infty$ for all $\mathbf{u} \in \mathcal{B}(b, q, p_\infty)$. Let $d_{cc'}^n$ be defined in (3.2.6) for all $c' \in \mathcal{I}^*(c)$. Assume that $2\Delta t \sum_{c' \in \mathcal{I}^*(c)} \frac{d_{cc'}^n}{m_c} \leq 1$. Then $\mathbf{U}_c^{n+1} \in \mathcal{B}(b, q, p_\infty)$*

Proof. Under the CFL condition $2\Delta t \sum_{c' \in \mathcal{I}^*(c)} \frac{d_{cc'}^n}{m_c} \leq 1$, we infer that \mathbf{U}_c^{n+1} is in the convex hull of $\{\bar{\mathbf{U}}_{cc'}(\frac{\|\mathbf{c}_{cc'}\|_{\ell^2}}{2d_{cc'}^n})\}_{c' \in \mathcal{I}(c)}$ (with the convention $\bar{\mathbf{U}}_{cc}(t) = \mathbf{U}_c^n$). It is proved in item (ii) in Lemma 1 that $\bar{\mathbf{U}}_{cc'}(\frac{\|\mathbf{c}_{cc'}\|_{\ell^2}}{2d_{cc'}^n}) \in \mathcal{B}(b, q, p_\infty)$ because $\frac{\|\mathbf{c}_{cc'}\|_{\ell^2}}{2d_{cc'}^n} \leq \frac{1}{2\hat{\lambda}_{\max}(\mathbf{n}, \tilde{\mathbf{U}}_c, \tilde{\mathbf{U}}_{c'})} \leq \frac{1}{2\lambda_{\max}(\mathbf{n}, \tilde{\mathbf{U}}_c, \tilde{\mathbf{U}}_{c'})}$, by definition of $d_{cc'}^n$, and $\hat{\lambda}_{\max}(\mathbf{n}, \tilde{\mathbf{U}}_c, \tilde{\mathbf{U}}_{c'})$ is an upper bound on $\lambda_{\max}(\mathbf{n}, \tilde{\mathbf{U}}_c, \tilde{\mathbf{U}}_{c'})$. This concludes the proof. \square

Corollary 3.3.2 (global invariance). *Let $n \in \mathbb{N}$. Assume that Δt is small enough to satisfy the global CFL condition $\min_{c \in \mathcal{N}^{\text{Cel}}} \left(2\Delta t \sum_{c' \in \mathcal{I}^*(c)} \frac{d_{cc'}^n}{m_c}\right) \leq 1$. Let \mathcal{B} be a convex invariant set. Assume that $\{\mathbf{U}_c^n \mid c \in \mathcal{N}^{\text{Cel}}\} \subset \mathcal{B}$. Then $\{\mathbf{U}_c^{n+1} \mid c \in \mathcal{N}^{\text{Cel}}\} \subset \mathcal{B}$.*

3.4 Mesh Velocity Reconstruction and Motion

In this section, we specify the mesh motion that yields an approximately mass-conservative scheme. In Section 3.5, we then introduce our procedure that enforces mass conservation in the sense of Definition 8.

3.4.1 Geometric Preliminaries

In order to update the mesh, we need to move the geometric nodes $\{\mathbf{a}_i^n\}_{i \in \mathcal{N}^{\text{geo}}}$ over the time interval $[t^n, t^{n+1})$. For this purpose, we reconstruct a velocity field that is continuous in space and constant over $[t^n, t^{n+1})$. We recall from §2.1.3 that this velocity field is denoted \mathbf{v}_h^n , and we use

the continuous geometric finite element space $\mathbf{P}_h^{\text{geo}}(\mathcal{T}_h^n) := (P_h^{\text{geo}}(\mathcal{T}_h^n))^d$ introduced in (2.1.4) to represent \mathbf{v}_h^n :

$$\mathbf{v}_h^n := \sum_{i \in \mathcal{N}^{\text{geo}}} \mathbf{V}_i^n \theta_i^n, \quad \text{with } \mathbf{V}_i^n := \mathbf{v}_h^n(\mathbf{a}_i^n). \quad (3.4.1)$$

At this point the vector field \mathbf{v}_h^n is not yet defined as $\{\mathbf{V}_i^n\}_{i \in \mathcal{N}^{\text{geo}}}$ is yet undefined. A fundamental problem of Lagrangian hydrodynamics is to define \mathbf{v}_h^n so that the total mass in each cell remains constant in time (see Definition 8) and the motion is compatible with thermodynamics (i.e., the method leaves $\mathcal{B}(b, q, p_\infty)$ invariant and (1.5.2) is satisfied); see also Remark 3.4.1.

Assuming \mathbf{v}_h^n to be known, we define the mesh motion over the time interval $[t^n, t^{n+1})$ by assigning the following displacement to the geometric Lagrange nodes:

$$\mathbf{a}_i(t) := \mathbf{a}_i^n + (t - t^n) \mathbf{V}_i^n, \quad (3.4.2)$$

and we define the Lagrangian mapping $\Phi_h^n : D^n \times [t^n, t^{n+1}) \rightarrow D^t$ and the associated velocity field $\mathbf{v}_h(., t) : D^t \rightarrow \mathbb{R}^d$ by setting

$$\Phi_h^n(\xi, t) := \sum_{i \in \mathcal{N}^{\text{geo}}} \mathbf{a}_i(t) \theta_i^n(\xi), \quad \mathbf{v}_h(\mathbf{x}, t) := \mathbf{v}_h^n(\Phi_{ht}^{-1}(\mathbf{x})), \quad \forall t \in [t^n, t^{n+1}). \quad (3.4.3)$$

Remark 3.4.1 (Traditional Lagrangian hydrodynamics). *It is common in the Lagrangian hydrodynamics literature to separate the specific volume from the velocity and the specific total energy and only solve (2.0.1c)-(2.0.1d). The density is usually evaluated pointwise by using the identity (2.0.1a) in strong form, $\rho_h(\Phi_h(\xi_l, t)) = \rho_0(\xi_l) \frac{1}{\det(\nabla \Phi_h(\xi_l, t))}$, where ξ_l are Gauss points on the reference element \widehat{K} , see e.g., [28, Eq. (2.9)]. As in this process mass conservation is decoupled from thermodynamics, it is not clear to us how to make this method compatible with the invariant domain property. Recall that whichever method that is used, it should leave $\mathcal{B}(b, q, p_\infty)$ invariant.*

3.4.2 Reconstruction of v_h^n at the Vertices

The mass conservation equation in (3.1.5) can be rewritten as follows:

$$\frac{m_c}{\Delta t}(\mathbf{T}_c^{n+1} - \mathbf{T}_c^n) = \sum_{m \in \hat{\mathcal{N}}^{\text{Fac}}} \hat{\mathbf{v}}_{F_m^n} \cdot \int_{F_m^n} \mathbf{n}_c \, ds + \sum_{c' \in \mathcal{L}^*(c)} d_{cc'}^n (\mathbf{T}_{c'}^n - \mathbf{T}_c^n). \quad (3.4.4)$$

We then define the following geometric objects and face-based velocity for all $F_m^n \in \mathcal{F}(K_c^n)$:

$$|F_m^n| := \left\| \int_{F_m^n} \mathbf{n}_c \, ds \right\|_{\ell^2}, \quad \text{and} \quad \mathbf{n}_{F_m^n} := |F_m^n|^{-1} \int_{F_m^n} \mathbf{n}_c \, ds. \quad (3.4.5a)$$

$$\mathbf{V}_{F_m^n} := \begin{cases} \frac{1}{2}(\mathbf{V}_c^n + \mathbf{V}_{c'}^n) + |F_m^n|^{-1} d_{cc'}^n (\mathbf{T}_{c'}^n - \mathbf{T}_c^n) \mathbf{n}_{F_m^n} & \text{if } F_m^n = K_c^n \cap K_{c'}^n, \\ \mathbf{V}_c & \text{if } F_m^n \in \mathcal{F}^\partial(K_c^n). \end{cases} \quad (3.4.5b)$$

Notice that the definition of $\mathbf{V}_{F_m^n}$ does not change when c and c' are exchanged; i.e., the definition does not depend on the orientation of the normal vector. With these definitions, (3.4.4) is re-written

$$\frac{m_c}{\Delta t}(\mathbf{T}_c^{n+1} - \mathbf{T}_c^n) = \sum_{m \in \hat{\mathcal{N}}^{\text{Fac}}} \mathbf{V}_{F_m^n} \cdot \mathbf{n}_{F_m^n} |F_m^n|. \quad (3.4.6)$$

3.4.2.1 1D Example

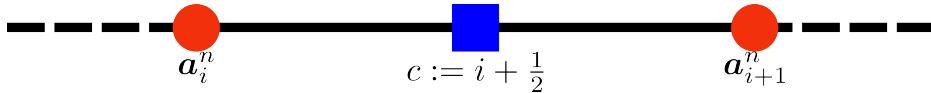


Figure 3.1: Schematic of a single one-dimensional cell element. The red circles denote cell vertices/faces, and the blue square represents the dG0 approximation of our hydrodynamics quantities.

Recalling that faces and vertices coincide in one space dimension, i.e., $a_i^n = F_i^n$ and $a_{i+1}^n = F_{i+1}^n$, the identity (3.4.6) shows that setting $v_h(a_i^n) = \mathbf{V}_{F_i^n}$ and $v_h(a_{i+1}^n) = \mathbf{V}_{F_{i+1}^n}$ yields exact mass conservation in one space dimension. See Figure 3.1 for our single cell configuration in 1D. Setting

the cell index to be $c := i + \frac{1}{2}$, this definition of the vertex velocity gives $\mathbf{a}_i^{n+1} = \mathbf{a}_i^n + \Delta t \mathbf{V}_{F_i^n}$, $\mathbf{a}_{i+1}^{n+1} = \mathbf{a}_{i+1}^n + \Delta t \mathbf{V}_{F_{i+1}^n}$. Then (3.4.6) yields

$$\mathbf{m}_c(\mathbf{T}_{i+\frac{1}{2}}^{n+1} - \mathbf{T}_{i+\frac{1}{2}}^n) = \Delta t(\mathbf{V}_{F_{i+1}^n} - \mathbf{V}_{F_i^n}) \quad (3.4.7)$$

$$= \mathbf{a}_{i+1}^{n+1} - \mathbf{a}_{i+1}^n - (\mathbf{a}_i^{n+1} - \mathbf{a}_i^n) \quad (3.4.8)$$

$$= |K_{i+\frac{1}{2}}^{n+1}| - |K_{i+\frac{1}{2}}^n|. \quad (3.4.9)$$

Assuming that exact mass conservation is achieved at t^n , i.e., $\mathbf{m}_c \mathbf{T}_{i+\frac{1}{2}}^n = |K_{i+\frac{1}{2}}^n|$, we deduce from the above identity that $\mathbf{m}_c \mathbf{T}_{i+\frac{1}{2}}^{n+1} = |K_{i+\frac{1}{2}}^{n+1}|$, i.e., exact mass conservation is achieved at t^{n+1} as well. This one-dimensional argument is an indication that the face velocity $\mathbf{V}_{F_m^n}$ may possibly be a good candidate to reconstruct \mathbf{v}_h in any space dimension.

3.4.2.2 Higher Dimensions

Recall that the reference geometric finite element $(\hat{K}, \hat{P}^{\text{geo}}, \hat{\Sigma}^{\text{geo}})$ is of Lagrange type. Most of the tests reported in Chapter 6 are done with $\hat{P}^{\text{geo}} = Q_1$, i.e., bilinear Lagrange elements on quadrangular meshes. We thus distinguish the geometric nodes that are vertices (i.e., images of the vertices of \hat{K} by the geometric mapping \mathbf{T}_K) from the rest of the nodes that are on edges or faces. The collection of the geometric nodes that are vertices is henceforth denoted $\mathcal{V}_h^{\text{vtx},n}$. When $\hat{P}^{\text{geo}} = Q_2$, we will use the extra nodes on the edges or faces to either help enforce mass conservation, as explained in §B.2.

Given a vertex $\mathbf{a}_i^n \in \mathcal{V}_h^{\text{vtx},n}$, let us denote by $\mathcal{F}(\mathbf{a}_i^n) \subset \mathcal{F}_h^n$ the set of all the mesh faces containing the node \mathbf{a}_i^n . In view of the above argument, our task now consists of constructing a value of $\mathbf{V}_i^n := \mathbf{v}_h(\mathbf{a}_i^n)$ that is consistent with the collection of the adjacent face velocities $\{\mathbf{V}_F^n\}_{F \in \mathcal{F}(\mathbf{a}_i^n)}$ given in (3.4.5b). We have tried many possible reconstructions (Crouzeix-Raviart, Raviart-Thomas, Rannacher-Turek, etc.,), but the reconstruction that seems to be the most robust across all the tests we have run (we started this project in 2020) consists of solving the following

local linear system whose size is $d \times d$:

$$\mathbf{M}_i \mathbf{V}_i^n = \mathbf{R}_i, \quad \text{where,} \quad (3.4.10a)$$

$$\mathbf{M}_i := \sum_{F \in \mathcal{F}(\mathbf{a}_i^n)} \mathbf{n}_F \otimes \mathbf{n}_F, \quad \text{and} \quad \mathbf{R}_i := \sum_{F \in \mathcal{F}(\mathbf{a}_i^n)} (\mathbf{n}_F \otimes \mathbf{n}_F) \mathbf{V}_F^n. \quad (3.4.10b)$$

Remark 3.4.2. *The above process consisting of solving the linear system (3.4.10) is identical to what is done in the Caveat scheme to compute the velocity at the cell vertices [1, p. 31]. A linear system similar to (3.4.10a) can also be found in [14] (see Eq. (13) therein), [64] (see Eq. (4.13) therein), and [63] (see Eq. (27) therein). In these papers, the linear system is used to reconstruct the vertex velocities $(\mathbf{V}_i^n)_{i \in \mathcal{N}^{\text{geo}}}$ from the cell velocities $(\mathbf{V}_c^n)_{c \in \mathcal{N}^{\text{cel}}}$, adding some numerical viscosities on the way. In our case as in [1], (3.4.10) is used to reconstruct the vertex velocities from the dG fluxes $(\mathbf{V}_F^n)_{F \in \mathcal{F}(\mathbf{a}_i^n)}$. The dG fluxes in the paper are constructed to make the update $(\mathbf{T}_c^{n+1}, \mathbf{V}_c^{n+1}, \mathbf{E}_c^{n+1})$ invariant-domain preserving, whereas this may not be the case in the above references.*

3.5 Mass Conservation

Over the course of this research, numerous methods have been explored to guarantee local cell mass conservation using the vertex velocity defined in (3.4.10a) as a baseline. In this work we enforce mass conservation by a post-processing correction applied to the density/specific volume after each time step. We explored several alternatives—an optimization-based redistribution, face-flux corrections, and stagewise adjustments—but these are deferred to Appendix B and are not part of the production algorithm described in the main text. Unless stated otherwise, “mass conservation” below refers exclusively to the post-processing procedure.

Remark 3.5.1 (Literature). *In [64], the authors develop a Lagrangian method that is locally mass conservative in two space dimensions. The key to this method is that the vertex velocities $(\mathbf{V}_i^n)_{i \in \mathcal{N}^{\text{geo}}}$ are computed first, and then the dG numerical fluxes for the mass conservation equation on each face F_m^n , i.e., $\mathbf{V}_{F_m^n}$, is defined by (B.2.4) with the left-hand side computed with one quadrature point $\zeta_1 = \frac{1}{2}$. Taking the arithmetic average of the vertex velocities at the extremities of the face*

does the job. This numerical flux cannot *a priori* ensure that the update $(\mathbf{T}_c^{n+1}, \mathbf{V}_c^{n+1}, \mathbf{E}_c^{n+1})$ is in the local invariant domain. It cannot also ensure that the update is in the global invariant domain: the specific internal energy may violate the inequality $e(\mathbf{u}) - q > p_\infty(\tau - b)$. That said, the method presented in the present paper has many similarities with [64]. The essential difference is that we define the numerical flux $\mathbf{V}_{F_m^n}$ so that the update $(\mathbf{T}_c^{n+1}, \mathbf{V}_c^{n+1}, \mathbf{E}_c^{n+1})$ is in the local invariant domain and we reconstruct the vertex velocities afterwards, whereas it is done the other way around in [64]. A second-order extension of [64] is proposed in [63]. An alternative first-order method based on [64] where the entropy is updated instead of the specific total energy is presented in [7]. The analysis therein is only semi-discrete (discrete in space and continuous in time).

Following the update to the state variable, described in section 3.1, we move the mesh with the non conservative mesh velocity found by solving the linear system given by equation (3.4.10). This results in a method that is not mass conservative. To remedy this, we add a post-processing step where the density is updated according to the updated geometry of the mesh by setting

$$\mathbf{T}_c^{n+1} := \frac{|K_c^{n+1}|}{m_c}, \quad \forall c \in \mathcal{N}^{\text{Ccl}}. \quad (3.5.1)$$

The above update is accepted only if $(\mathbf{T}_c^{n+1}, \mathbf{V}_c^{n+1}, \mathbf{E}_c^{n+1})^\top \notin \mathcal{B}(b, q, p_\infty)$.

3.5.1 Summary

We now summarize the full algorithm using this method.

Step 1: Given the update $(\mathbf{T}_c^n, \mathbf{V}_c^n, \mathbf{E}_c^n)_{c \in \mathcal{N}^{\text{Ccl}}}$ at time t^n , the first step of the algorithm consists of computing the update $(\tilde{\mathbf{T}}_c^{n+1}, \mathbf{V}_c^{n+1}, \mathbf{E}_c^{n+1})_{c \in \mathcal{N}^{\text{Ccl}}}$ from (3.1.5). We use the symbol $\tilde{\mathbf{T}}_c^{n+1}$ to remind the reader that the actual specific volume update is computed at the last step.

Step 2: Then, one computes the vertex velocities by solving the $d \times d$ linear system (3.4.10a) for every vertex in the mesh. This gives a Q_1 representation of the vertex velocity \mathbf{v}_h^n . A higher-order representation of \mathbf{v}_h can be obtained by using the butterfly algorithm, initially introduced in [31], or consistent variations thereof to make the mesh motion less stiff.

Step 3: Finally, the actual value of the specific volume update T_c^{n+1} is obtained by using (3.5.1). Notice that the update \tilde{T}_c^{n+1} obtained from (3.1.5) is not used at all. The equation (3.5.1) is the actual definition of the specific volume update T_c^{n+1} .

4. HIGHER ORDER APPROXIMATION VIA LIMITING

Our goal in this section is to take an existing high-order Lagrangian method that may violate the invariant-domain property (IDP) and apply a limiting procedure at each stage of the solution process to enforce IDP while preserving mass conservation. Rather than employing convex limiting, as was done in the Eulerian frame in [44] and later for an arbitrary equation of state in [21], we instead follow the mass-conservative limiting strategy outlined in [42, Sec. 2.2], originally motivated by applications to steady-state radiation transport equations.

Importantly, the limiting strategy we describe is not restricted to a particular implementation; it can, in principle, be applied to any high-order Lagrangian code whose update may leave the invariant domain. In this work, we focus on the miniapp Laghos, developed at Lawrence Livermore National Laboratory [28], as a representative example of a high-order Lagrangian hydrodynamics solver.

In the discrete setting, one typically distinguishes between a low-order scheme that is guaranteed to satisfy the invariant-domain property, and a high-order scheme that may not. The low-order update serves as a fallback approximation that remains IDP under suitable CFL restrictions, while the high-order update provides accuracy but can step outside the admissible set. The role of the limiter is to combine the two updates in a nonlinear fashion so that the final update remains inside the invariant set, while staying as close as possible to the high-order solution. This approach follows the structural assumptions introduced in [33], where the update at each time step is constructed as

$$U^{n+1} = \mathcal{L}(U^n, \Phi_L, \Phi_H) \in \mathcal{B}^I, \quad (4.0.1)$$

with Φ_L and Φ_H denoting the low- and high-order flux contributions, respectively, and \mathcal{L} denoting the nonlinear limiting operator. The nonlinear operator is always designed so that U^{n+1} remains as close as possible to the high-order update while preserving the invariant domain and global conservation.

The remainder of this chapter is organized as follows. We first recall the principle of mass conservation in the context of Laghos. Next, we review explicit Runge–Kutta (ERK) methods and then summarize the IDP-ERK framework introduced in [33]. Finally, we present the specific limiter chosen for this work and describe its implementation within the Laghos framework.

4.1 Mass Conservation

In this section, we break down what is meant by mass conservation in the context of Laghos. A fundamental postulate of Lagrangian hydrodynamics is that the mass in any volume $U(t)$ does not change in time:

$$\frac{d}{dt} \int_{U(t)} \rho dV = 0. \quad (4.1.1)$$

The general framework of Laghos is to eliminate the density through the strong mass conservation principle, which takes the form

$$\rho(t) |\det \mathbf{J}_z(t)| = \rho(t_0) |\det \mathbf{J}_z(t_0)|, \quad (4.1.2)$$

noting that the density defined here is not a finite element or even a polynomial function. This notion allows the authors to define the density at any particular point in time and space as a function. In summary, to compute zonal mass, a quadrature rule is employed

$$\int_{D_z^t} \rho dV \approx \sum_{q \in Q} \omega_q \rho(x_q, t) |\det \mathbf{J}_z(x_q)|, \quad (4.1.3)$$

where $x_q = \Phi_{D_z^t}(\xi_q, t)$ is the quadrature point in the Lagrangian coordinates, ω_q is the quadrature weight, and $\mathbf{J}_z(x_q)$ is the Jacobian of the mapping from Eulerian to Lagrangian coordinates, and Q is the chosen quadrature rule. In practice, Q is taken to be a tensor product Gauss-Legendre quadrature rule.

For visualization purposes, the density at any time can be projected onto the dG space. For a given cell c , to construct the “mass conservative” density on the dG degrees of freedom, first the initial density field, or RHS of 4.1.2 is projected onto the degrees of freedom via the DensityInte-

grator class, call this object $\bar{\rho}_{c,0} \in \mathbb{R}^{|\eta^c|}$. This object is defined as

$$\bar{\rho}_{c,0,i} = \sum_{q \in Q} \omega_q |\det \mathbf{J}_0(x_q)| \phi_i(x_q) \rho_0(x_q) \quad \forall i \in \eta^c, \quad (4.1.4)$$

Then, to solve for the density at time t , a local mass matrix is inverted

$$\rho_{c,i} = (M_\rho^{-1} \bar{\rho}_0)_i \quad \text{where} \quad M_{i,j} := \sum_{q \in Q} \omega_q |\det \mathbf{J}_z(x_q)| \phi_j(x_q) \phi_i(x_q). \quad (4.1.5)$$

The object $\rho_{c,i}(t)$ defined in 4.1.5 is the high order density we wish to limit.

4.2 Review on Runge-Kutta Methods

Since first-order time-stepping schemes inherently limit the temporal accuracy of the solution, we turn to higher-order methods—specifically, explicit Runge–Kutta schemes—to enable higher-order convergence of the limited values. We employ explicit time-stepping schemes to resolve the stress and shock waves accurately in time. To quantify the tradeoff between stability and computational cost in time integration, we adopt the efficiency rating of Runge–Kutta methods as defined in [33, Def. 1.1], which characterizes the effective timestep per unit of computational effort.

Definition 9 (efficiency ratio). *Let τ^* be the maximal time step that makes the forward Euler method IDP. Consider some s -stage ERK method and let $\tilde{\tau}$ be the maximal time step that makes this method IDP as well. We call the efficiency ratio of the s -stage ERK method the ratio $c_{\text{eff}} := \frac{\tilde{\tau}}{s\tau^*}$.*

4.2.1 General Explicit Runge-Kutta

Runge-Kutta methods approximate solutions to a differential equation of the form

$$\frac{\partial u}{\partial t} = f(t, u).$$

The most general form of an ERK rule with s stages is

$$u^{n+1} = u^n + h \sum_{i=1}^s b_i k_{n,i},$$

where

$$k_{n,i} = f \left(t^n + c_i h, u^n + h \sum_{j=1}^{i-1} a_{i,j} k_{n,j} \right),$$

with corresponding Butcher tableau

	0				
c_2		$a_{2,1}$			
c_3		$a_{3,1}$	$a_{3,2}$		
\vdots		\vdots		\ddots	
c_s		$a_{s,1}$	$a_{s,2}$	\cdots	$a_{s,s-1}$
		b_1	b_2	\cdots	b_{s-1}
					b_s

Since we only consider explicit methods in this thesis, we have that $a_{j,j} = 0$ for all $j \in \{1 : s\}$. Notice also that in order for our ERK method to be consistent, we require $\sum_{j \in \{1:s\}} b_j = 1$. Following Butcher's row-sum condition, we also assume $\sum_{\ell=1}^j a_{j,\ell} = c_j$, $j = 1, \dots, s$, which implies $c_1 = 0$. For conventional simplicity, we maintain the formalism used in previous stages and write the final stage of the ERK scheme involving the b_j 's using the convention $a_{s+1,k} = b_k$ for all $k \in \{1 : s\}$ and $c_{s+1} := 1$, and hence $t^{n,s+1} = t^{n+1}$.

4.2.2 Example Runge-Kutta Methods

The numerical experiments presented in Chapter 6 employ a selection of explicit Runge–Kutta (ERK) time integration methods, which are detailed below. We adopt the notation $\text{RK}(s, p; c_{\text{eff}})$, where s denotes the number of stages, p the formal order of accuracy, and c_{eff} the efficiency ratio, as defined in Definition 9. A summary of representative schemes, along with their associated Butcher

tableaux, is provided below.

- Forward Euler

$$u^{n+1} = u^n + hk_{n,1} \quad \text{where} \quad k_{n,1} = f(t^n, u^n),$$

with corresponding Butcher tableau

0	0
1	

- RK (2, 2; 1). The update at time t^{n+1} is given by

$$u^{n+1} = u^n + h \left[\left(1 - \frac{1}{2a}\right) k_{n,1} + \frac{1}{2a} k_{n,2} \right]$$

where

$$k_{n,1} = f(t^n, u^n)$$

$$k_{n,2} = f(t^n + ah, u^n + ahk_{n,1}),$$

with corresponding Butcher tableau

0	0	
a	a	0
1	$1 - \frac{1}{2a}$	$\frac{1}{2a}$

In practice, we take $a = \frac{1}{2}$ which is the so-called *midpoint rule*, but one can choose any $a \in (0, 1]$.

- RK (3, 3; 1). The update at time t^{n+1} is given by

$$u^{n+1} = u^n + h \left[\frac{1}{4} k_{n,1} + \frac{3}{4} k_{n,3} \right]$$

where

$$\begin{aligned} k_{n,1} &= f(t^n, u^n) \\ k_{n,2} &= f\left(t^n + \frac{h}{3}, u^n + \frac{h}{3}k_{n,1}\right), \\ k_{n,3} &= f\left(t^n + \frac{2h}{3}, u^n + \frac{2h}{3}k_{n,2}\right) \end{aligned}$$

with corresponding Butcher tableau

	0	0	
$\frac{1}{3}$	$\frac{1}{3}$	0	
$\frac{2}{3}$	0	$\frac{2}{3}$	0
1	$\frac{1}{4}$	0	$\frac{3}{4}$

This method is generally referred to as *Heun's method*.

- RK $(4, 4; \frac{1}{2})$. While the ERK methods previously described have had optimally equidistributed substeps and maintain convergence order equal to the number of substeps, it is shown in passing that this can not exist for four-stage ERK methods in [33, Sec. 4.1]. For this reason, the fourth-order time stepping scheme we describe will not have a maximal efficiency ratio (see Def. 9). The update at time t^{n+1} is given by

$$u^{n+1} = u^n + \frac{h}{6} (k_{n,1} + 2k_{n,2} + 2k_{n,3} + k_{n,4})$$

where

$$\begin{aligned} k_{n,1} &= f(t^n, u^n) \\ k_{n,2} &= f(t^n + 0.5h, u^n + 0.5hk_{n,1}) \\ k_{n,3} &= f(t^n + 0.5h, u^n + 0.5hk_{n,2}) \\ k_{n,2} &= f(t^n + h, u^n + hk_{n,3}), \end{aligned}$$

with corresponding Butcher tableau

		0		
$\frac{1}{2}$		$\frac{1}{2}$	0	
$\frac{1}{2}$		0	$\frac{1}{2}$	0
1		0	0	1 0
		$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$ $\frac{1}{6}$

- RK $(4, 4; \frac{3}{4})$. The second fourth order method we highlight is often called the $\frac{3}{8}$ rule. It has equidistributed substeps, though still does not have maximal efficiency ratio. The Butcher tableau is given by

		0		
$\frac{1}{3}$		$\frac{1}{3}$	0	
$\frac{2}{3}$		$-\frac{1}{3}$	1 0	
1		1 -1 1 0		
		$\frac{1}{8}$	$\frac{3}{8}$	$\frac{3}{8}$ $\frac{1}{8}$

(4.2.1)

This is the timestepping scheme that we employ for all our high order results presented in section 6.3.

As is noted in [32, Lem. 78.17], RK2 methods are conditionally stable. In nonlinear problems or shock-dominated problems, RK2 methods may be unstable unless the CFL condition is severely

restricted. As such we will generally avoid RK2 methods in favor of RK3 and RK4 methods for their stability results [8, Lem. 4.3].

4.3 Summary of the IDP-ERK Framework

The invariant-domain-preserving (IDP) explicit Runge–Kutta (ERK) framework, introduced in [33], provides a systematic approach for combining a high-order discretization with a provably invariant-domain-preserving low-order method. The framework relies on two key ideas: (i) at each stage $l \in \{2 : s+1\}$ one constructs a low order update $\mathbf{U}^{L,l}$ and a high order update $\mathbf{U}^{H,l}$ which are constructed from the same starting state, (ii) rewrite the l -th stage of the ERK method in incremental form. We now detail how these ideas are realized in the stage construction of the IDP–ERK method.

At stage l , instead of starting from the initial state U_n , we use as reference the most recent IDP state $U_{n,l'}$, where $l' < l$ is chosen such that $c_l - c_{l'} \geq 0$ is minimal. This choice minimizes the CFL restriction and remains meaningful even for confluent ERK methods where multiple c_j coincide. Defining

$$\Delta c_{\max} := \max_{2 \leq l \leq s+1} (c_l - c_{l'}),$$

we note that $\Delta c_{\max} \geq 1/s$, with equality when the c_j are uniformly distributed.

Given $U_{n,l'}$, we construct two provisional updates to approximate the solution at $t_{n,l}$: a low-order forward Euler step

$$\mathbb{M}^L \mathbf{U}^{L,l} = \mathbb{M}^L \mathbf{U}^{n,l'} + \tau(c_l - c_{l'}) \mathbf{F}^L(\mathbf{U}^{n,l'}),$$

and a high-order update written in incremental form,

$$\mathbb{M}^H \mathbf{U}^{H,l} = \mathbb{M}^H \mathbf{U}^{n,l'} + \tau \sum_{k=1}^{l-1} (a_{l,k} - a_{l',k}) \mathbf{F}^H(\mathbf{U}^{n,k}).$$

Both $\mathbf{U}^{L,l}$ and $\mathbf{U}^{H,l}$ use the IDP state $\mathbf{U}^{n,l'}$ as a starting point, which allows a consistent compari-

son. The final stage state is then obtained by applying the limiting operator,

$$\mathbf{U}^{n,l} := \ell \left(\mathbf{U}^{n,l'}, \Phi^{L,l}, \Phi^{H,l} \right), \quad (4.3.1)$$

with fluxes

$$\Phi^{L,l} := (c_l - c_{l'}) \mathbf{F}^L(\mathbf{U}^{n,l'}), \quad \Phi^{H,l} := \sum_{k=1}^{l-1} (a_{l,k} - a_{l',k}) \mathbf{F}^H(\mathbf{U}^{n,k}). \quad (4.3.2)$$

In summary, the s -stage IDP-ERK method is described in Algorithm 1. The reader is referred to [33, Lem. 2.2] on a proof that the update obtained in this manner is IDP.

4.4 Review on Limiters

In the literature, several limiting techniques have been proposed that either guarantee some sort of positivity-preserving qualities or stability preserving quantities.

- In [16, Eqns. 3.6-8], the authors propose a high order finite difference scheme by using the essentially non-oscillatory (ENO) reconstruction, the strong stability preserving (SSP) high order time discretizations and the positivity-preserving scaling limiter which they prove maintains conservation. The limiting is done by defining a modified reconstruction polynomial

$$\tilde{\mathbf{U}}_i(x) = \theta_i (\mathbf{U}_i - \bar{\mathbf{U}}_i) + \bar{\mathbf{U}}_i$$

where $\bar{\mathbf{U}}_i$ is defined to be the average over the cell. Beginning with the assumption that the cellwise average $\bar{\mathbf{U}}_i$ is in the positivity preserving set G at time t_n , this value is used to find the limited update at the quadrature points at time t^{n+1} which are also in G . Their process of finding these limited values consists of first finding a limiter for the density and then second finding a separate limiter for the internal energy.

- In [83] the authors implement a limiting procedure which relies on the characteristic variables, an approach first introduced in [22]. The authors claim that when applying a limiting

procedure directly on the physical variables it is impossible to enforce totally the monotonicity of solutions.

- In [87], the authors introduce both a synchronized limiter and a linearized limiter. The synchronized limiter adopts the same scalar limiting ratio and is chosen over the whole mesh element to ensure density, velocity, and specific internal energy remain in bounds, but falls back to first order on smooth flows. The linearized limiter limits each of the conserved variables separately and is shown to obtain second order accuracy. Neither limiter guarantees that an invariant-domain is preserved for the solution process.

For the purpose of this thesis, we focus on limiting in the density field alone. Limiting in the energy is a current work in progress. Our limiter in the density does guarantee that the high order update remains invariant-domain preserving for the density. The locally mass conservative limiting procedure we choose to employ is described in [42, Ch. 2], of which we give a brief summary here. We later turn our focus to a convergence analysis on a selection of smooth flow test cases in section 6.3.

4.4.1 The Setting

We consider a set of degrees of freedom $\{u_i\}_{i \in \mathcal{V}}$ with associated nonnegative lumped masses $m_i \geq 0$. For each degree of freedom $i \in \mathcal{V}$, we define a stencil $\mathcal{K}(i) \subset \mathcal{V}$ representing the neighboring degrees of freedom with which i can exchange mass, and we denote $\mathcal{K}(i)^* := \mathcal{K}(i) \setminus \{i\}$. In addition, we introduce the set $\mathcal{V}(i) \subset \mathcal{V}$, which collects the degrees of freedom of the low-order method that are used to determine the local maximum and minimum values required for the limiting procedure. Our objective is to enforce these local bounds while preserving local mass.

Remark 4.4.1 (Various stencils). *Three distinct stencils are used throughout this thesis. \mathcal{I} denotes the stencil on which the low order finite element update relies, $\mathcal{K}(i)$ identifies the degrees of freedom that belong to the same high-order cell as i , and $\mathcal{V}(i)$ specifies the low-order degrees of freedom that assist in establishing local bounds. These stencils are visually represented in Figure 4.1.*

4.4.2 Obtaining the Local Bounds

To define the stencil $\mathcal{V}(i)$ for the limiting procedure, we first describe the setting of the low-order method and its relation to the high-order finite element discretization.

Suppose the high-order method employs piecewise polynomial elements of degree $r \geq 1$. In d dimensions, each high-order mesh element K therefore contains $(r + 1)^d$ degrees of freedom (DoFs). The low-order method, by contrast, is taken to be piecewise constant ($r = 0$) on a refined mesh obtained by subdividing each high-order element into $(r + 1)^d$ subcells, as illustrated in Figure 4.2. This construction guarantees that each subcell contains exactly one high-order DoF, establishing a one-to-one correspondence between the high- and low-order discretizations. This correspondence allows the low-order values to serve as local bounds for the high-order solution.

Remark 4.4.2. *This construction ensures that the low-order stencil $\mathcal{V}(i)$ is naturally aligned with the high-order FE mesh. It also guarantees that each high-order degree of freedom is represented in the low-order mesh, which is crucial for preserving the invariant domain property during limiting.*

For each high-order degree of freedom i , we define the stencil $\mathcal{V}(i)$ as follows. Let $K(i)$ denote the cell containing DoF i . Then

$$\mathcal{V}(i) := K(i) \cup \{\text{dofs in any adjacent cell that shares the same coordinate location as } i\}. \quad (4.4.1)$$

The stencil includes the high-order degrees of freedom within the element as well as selected neighboring degrees of freedom that share the same boundary vertex. The reduced stencil excluding the dof i itself is

$$\mathcal{V}(i)^* := \mathcal{V}(i) \setminus \{i\}. \quad (4.4.2)$$

The admissible range for the high-order update is then determined by the minimum and maximum of the density values over this stencil. In addition, one may optionally enforce global maximum and minimum values, further constraining the solution if desired.

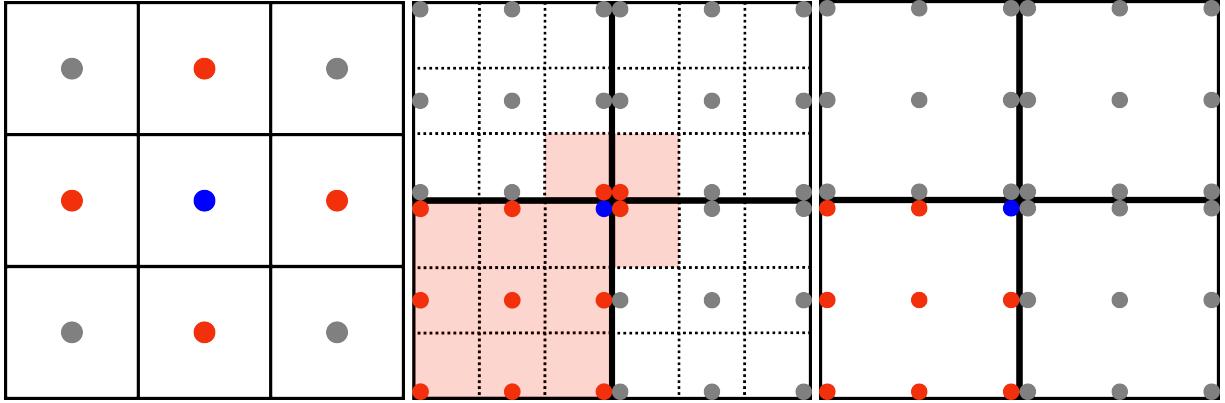


Figure 4.1: Illustration of the various stencils used in the high order implementation. Red DoFs represent the node indices that are part of the stencil. In the case of the max/min stencil, red submesh cells indicate the contained DoFs. Blue nodes indicate node i . Left: low order state update stencil $\mathcal{J}^*(i)$. Middle: max/min stencil $\mathcal{V}^*(i)$ defined on a Q_2 approximation space. Right: mass conservation stencil $\mathcal{K}^*(i)$ defined on a Q_2 approximation space.

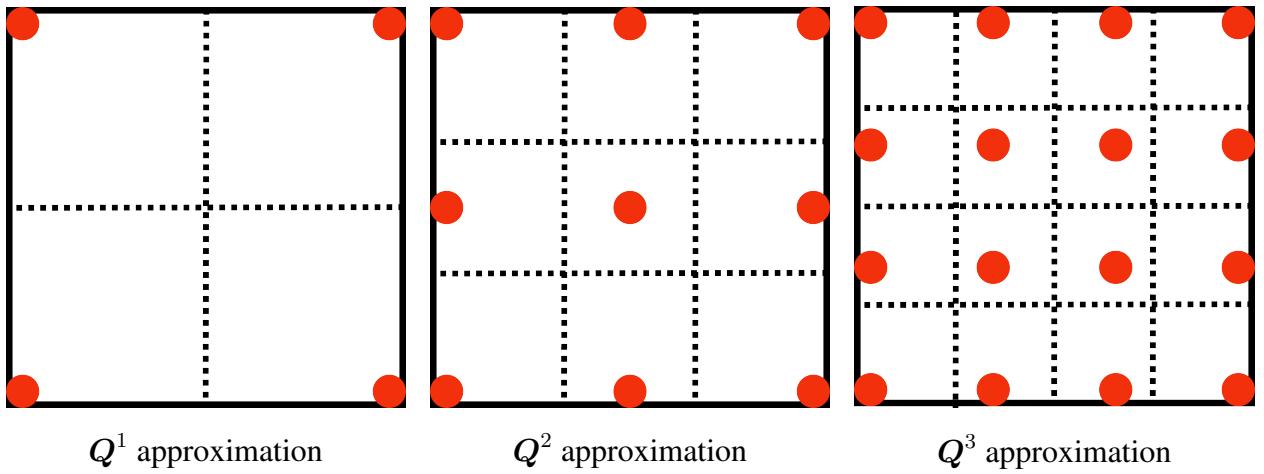


Figure 4.2: Examples of high order finite element mesh elements and the associated low order mesh elements. The low order mesh element boundaries are given by the dashed lines. Each dashed cell corresponds to a single low order degree of freedom.

4.4.3 Relaxation

To prevent order reduction, it is necessary to apply a second-order relaxation of the local bounds, as is standard in local limiting techniques (see [44, Sec. 4.7] and for a detailed discussion). In the finite volume literature, similar ideas are implemented through relaxed slope reconstructions [48, 75]. In this work, we adopt the methodology of [42, Sec. 3.2] and [44, Sec. 4.7], which estimates local curvature of the solution at Lagrange nodes using stiffness coefficients of the Laplace operator

$$\Delta_i^2 := \frac{\sum_{j \in \mathcal{I}(i)^*} \beta_{ij} (u_i - u_j)}{\sum_{j \in \mathcal{I}(i)^*} \beta_{ij}}, \text{ and then set } \overline{\Delta_i^2} := \text{minmod} \left\{ \Delta_j^2 \right\}_{j \in \mathcal{I}(i)}. \quad (4.4.3)$$

Notice that Δ_i^2 is well defined as $\sum_{j \in \mathcal{I}(i)^*} \beta_{ij} = \beta_{ii} = - \int_D (\nabla \phi_i)^2 d\mathbf{x} \neq 0$ by the partition of unity property. The resulting curvature estimate $\overline{\Delta_i^2}$ is obtained by combining with a minmod operation. The minmod function of a finite set is defined to be zero if there are two numbers of a different sign in this set, and it is set to the number whose absolute value is the smallest otherwise. These relaxed bounds expand the admissible range of the solution while preserving global extrema, thereby preventing excessive clipping and ensuring that second-order accuracy can be maintained. The pseudo-code for this relaxation technique is given in Algorithm 3 in Appendix C.

4.4.4 Locally Conservative Limiting

The procedure operates on a set of degrees of freedom $\{u_i\}_{i \in \mathcal{V}}$ with associated nonnegative lumped masses $m_i \geq 0$ and local bounds u_i^{\min} and u_i^{\max} . For each degree of freedom $i \in \mathcal{V}$, we define a stencil $\mathcal{V}(i) \subset \mathcal{V}$ representing the neighboring degrees of freedom with which i can exchange mass, and let $\mathcal{V}(i)^* := \mathcal{V}(i) \setminus \{i\}$.

The goal is to enforce the local bounds while preserving mass locally. For each degree of freedom i , if $m_i = 0$, the value is simply projected onto the admissible range:

$$y_i := \min(u_i^{\max}, \max(u_i^{\min}, u_i)),$$

which does not affect mass.

If $m_i > 0$, the algorithm enforces the maximum principle by first computing

$$a_i^+ := \sum_{j \in \mathcal{V}(i)^*} m_j \max(0, u_j^{\max} - u_j), \quad b_i^+ := \max\left(u_i - \frac{a_i^+}{m_i}, u_i^{\max}\right),$$

and a limiting coefficient

$$\ell_i^+ := \begin{cases} 0 & \text{if } a_i^+ = 0, \\ \frac{m_i(u_i - b_i^+)}{a_i^+} & \text{otherwise.} \end{cases}$$

The updated degrees of freedom are then

$$y_j := u_j + \ell_i^+ \max(0, u_j^{\max} - u_j), \quad \forall j \in \mathcal{V}(i)^*, \quad y_i := b_i^+,$$

ensuring local mass conservation and that $u_j \leq y_j \leq \max(u_j, u_j^{\max})$ for neighboring nodes j , and $u_i^{\max} \leq y_i \leq u_i$. In Figure 4.3 an example of a HO DoF whose density violates the local maximum value is presented where excess density is distributed to the neighboring DoFs of the cell.

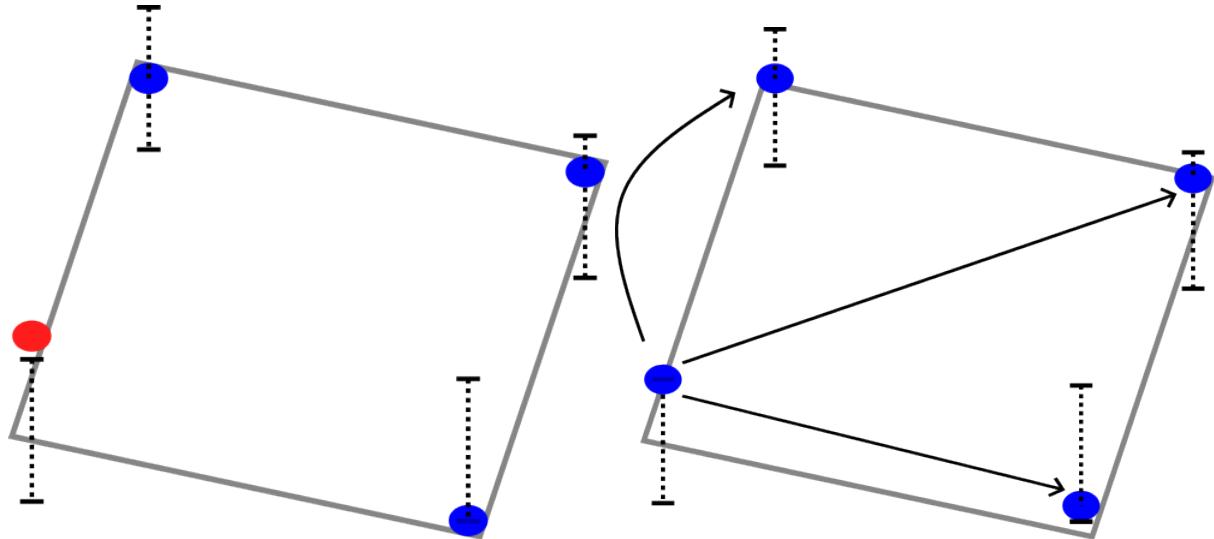


Figure 4.3: Nodal limiting on a Q_1 element. One node (red) exceeds its local upper bound and the excess mass is redistributed within the element to neighboring DoFs (blue) that have remaining capacity per their bounds. Dashed lines indicate local admissible range for each DoF.

Analogously, the minimum principle is enforced by computing

$$a_i^- := \sum_{j \in \mathcal{V}(i)^*} m_j \max(0, u_j - u_j^{\min}), \quad b_i^- := \min \left(u_i + \frac{a_i^-}{m_i}, u_i^{\min} \right),$$

with

$$\ell_i^- := \begin{cases} 0 & \text{if } a_i^- = 0, \\ \frac{m_i(b_i^- - u_i)}{a_i^-} & \text{otherwise,} \end{cases}$$

and the corresponding updates

$$y_j := u_j - \ell_i^- \max(0, u_j - u_j^{\min}), \quad \forall j \in \mathcal{V}(i)^*, \quad y_i := b_i^-.$$

This procedure is applied iteratively to all degrees of freedom. The resulting updates preserve mass locally, enforce the local minimum and maximum bounds, and maintain neighboring values within their admissible ranges. Guermond and Wang [42] note that numerical experiments indicate that applying this limiting procedure twice is typically sufficient for convergence when starting from a reasonable approximation, such as a linearly stabilized Galerkin solution. The pseudo-code for this limiting technique is given in Algorithm 2 in Appendix C.

4.5 Full Workflow

The purpose of this section is to clearly define the modifications that are made to Laghos to make the resulting high order approximation to the density IDP. We want to be clear that for the present moment our method limits the high order density alone. It is a current work in progress to limit in the specific internal energy. The full IDP limiting process for a single stage of an IDP ERK timestep is described in Algorithm 4 in Appendix C.

4.5.1 Summary of Modifications to Laghos

The modifications made to Laghos can be summarized as:

- **General EOS support.** Refactored material-property evaluation to allow arbitrary equa-

tions of state (beyond ideal gas), updating the pressure and sound-speed computations used throughout the scheme.

- **Low-order (LO) baseline.** Introduced
 - the LO submesh (refinement illustrated in Fig. 4.2), and
 - LO finite element spaces defining the cell-centered state $U = (\tau, \mathbf{v}, E)$ on submesh zones.
- **HO→LO transfers and mesh sync.** Form the dG0 (cellwise-constant) velocity by interpolating the continuous HO velocity onto the dG0 space. For mesh motion, use a one-to-one vertex mapping and copy HO nodal velocities to the corresponding LO vertices so the LO coordinates advance identically and LO/HO geometries coincide at every stage (see §4.5.3). Scalars are transferred via conservative cell averages (see §4.5.2).
- **High-order density evaluation.** Evaluate density via strong mass conservation at quadrature and recover a finite-element representation by elementwise L^2 projection (see §4.5.4).
- **IDP–ERK limiting.** Implemented the invariant-domain-preserving explicit Runge–Kutta framework with local bounds, locally conservative limiting, and relaxation (see §4.3).
- **Stagewise mass matrices.** Reassemble the kinematic and thermodynamic mass matrices at each RK stage using the current, limited density $\rho_h^{H,n,l}$, replacing the original cached-density approach in Laghos (see §4.5.5; cf. Remark 4.5.3).
- **Operator refactor.** The update procedure was divided into two stages—one performing the unconstrained advance, and another applying limiter constraints based on low-order reference bounds.

4.5.2 Interpolation from HO to LO

Remark 4.5.1 (High order notation). *Laghos defines the following finite element spaces for their approximation*

- a kinematic space $\mathcal{V}^H \subset [H^1(D)]^d$ with a basis $\{w_j^H\}_{j=1}^{N_V^H}$,
- a thermodynamic space $\mathcal{E}^H \subset L_2(D)$ with basis $\{\phi_j^H\}_{j=1}^{N_E^H}$.

Note that we augment the definition of the spaces, shape functions, and total number of basis functions with the superscript H to clearly indicate that they describe the high order space.

Recall that our low order method follows the cell-centered Lagrangian framework, in contrast to the staggered-grid approach that is employed in the Laghos framework. Since the IDP-ERK timestepping (see Sec. 4.3) framework we adopt requires the high order and low order update to be computed from the same initial state, we must project the high order state back onto the low order state. This is done purely by interpolation on the degrees of freedom. Specifically, given the high order nodal coefficients of the velocity $\{V_i^{H,n}\}$, we have the finite element approximation

$$\mathbf{v}_h^H(\mathbf{x}) = \sum_i V_i^{H,n} w_i^H(x, t).$$

We then define the low order dG coefficients by interpolation

$$\mathbf{V}_c^{L,n}(\mathbf{x}) := v_h^H(\mathbf{x}), \quad (4.5.1)$$

where \mathbf{x} is the coordinate location of the low order dG node.

4.5.3 High Order Mesh Velocity

In our high order limiting strategy our previous discussion on mesh velocity in the low order case in Section 3.4 is unused. In our workflow, the low-order mesh has the same Lagrange vertex DOFs as the high-order mesh, so we keep the two configurations identical by moving the LO nodes with the HO mesh velocity. At each stage, we copy the HO nodal velocities onto the corresponding LO nodes and update the LO coordinates using those velocities; equivalently, we can simply overwrite the LO coordinates with the HO coordinates after each stage.

Remark 4.5.2 (Interpolation of Curvilinear Mesh Interfaces). *In Laghos, the high-order mesh*

representation includes curved elements while the auxiliary low-order mesh is defined with linear interfaces. As a result, directly copying vertex coordinates from the high-order mesh does not perfectly reproduce the curved geometry but incurs some interpolation error. As is noted in [54], curvilinear meshes are essential to observing greater than second order accuracy. Since the low order mesh is solely used to establish bounds for the limiting of the high order solution, we don't believe that this linear interpolation of mesh boundaries will result in order reduction. We recall as well that relaxation of the bounds according to section 4.4.3 helps to mitigate order reduction that may be introduced through the limiting process.

4.5.4 High-Order Density Evaluation (Pre-Limiting)

In the Laghos framework, the density is not solved at the level of the PDE but is rather represented by an equation the authors call the *strong mass conservation principle* (see [28, Eq. 4.8])

$$\rho(t) |\mathbb{J}_c(t))| = \rho(t_0) |\mathbb{J}_c(t_0))|, \quad (4.5.2)$$

where \mathbb{J} represents the Jacobian of the Lagrange mapping, defined in (1.4.4). Notice that this definition of the density is not a finite element at all, but rather is a way to define the density at any given point in any cell $c \in \eta^{\text{Cel}}$. The function $\rho(t)$ implied by (4.5.2) is generally not a polynomial on D^t , so we recover a finite-element representation $\rho_h(t) \in \mathcal{E}^H$ by an elementwise L^2 projection.

4.5.5 Stagewise Update of Thermodynamic and Kinematic Mass Matrices

In Laghos, the thermodynamic and kinematic mass matrices are weighted by the strong mass density (4.5.2) (effectively using the cached $\rho_0 |\mathbb{J}_0|$ at quadrature). In our high-order method with limiting, the density ρ_h^H is smoothed/limited at every Runge–Kutta stage. To keep the operators consistent with the state actually used by the limiter and the EOS, we reassemble the mass matrices at each stage using the current, limited density.

$$\mathbf{M}_{\mathcal{V}}^{H,n,l} = \int_{D^t} \rho_h^{H,n,l} \mathbf{w} \mathbf{w}^\top dx, \quad \mathbf{M}_{\mathcal{E}}^{H,n,l} = \int_{D^t} \rho_h^{H,n,l} \phi \phi^\top dx \quad (4.5.3)$$

Here w and ϕ are the kinematic and thermodynamic basis functions evaluated on the current configuration D^t , and $\rho_h^{H,n,l}$ is the density after the stage's evaluation and limiting (see §4.4.4). Notice also that the notation from [28] has changed to include superscripts, which indicate time dependence.

Remark 4.5.3. *The proof that the semidiscrete method in Laghos is conservative in both total mass and total energy relies on both the kinematic and thermodynamic mass matrices being symmetric and independent of time. To show this the authors use the Reynold's Transport Theorem 1.4.4 and the Lagrangian tenet that mass in a control volume remains unchanged in time. Since neither of these principles are violated during the limiting process we claim that the updated limited scheme also remains exactly conservative in the mass and energy.*

5. NONLINEAR HYPERELASTICITY

5.1 Introduction

The simulation of elastic and elasto-plastic behavior of solid materials is of central importance in applications such as metal forming, high-rate impact, and biological material modeling. These problems may be studied in either an Eulerian or Lagrangian framework, each offering distinct benefits and limitations. In the Eulerian framework, interface capturing becomes increasingly complex, typically requiring specialized methods such as volume-of-fluid [70, 50], level set [36, 71], or diffuse-interface approaches [68, 46]. We would like accurately capture material interfaces while avoiding use of such techniques and thus would like to adapt our Lagrangian framework introduced in this thesis to the elastic regime.

Within either framework, two main classes of models have been proposed for the description of finite elasticity: hypoelastic and hyperelastic formulations. Hypoelastic models, originally introduced in [86], prescribe an evolution law for the deviatoric stress. While widely used in engineering applications and hydrocodes, these models suffer from inherent drawbacks. They can be ill-posed, may violate the second law of thermodynamics, and depend on the choice of objective stress rates, which introduces non-conservative and sometimes ambiguous terms.

Hyperelastic models, by contrast, are derived from a stored energy potential and possess stronger theoretical underpinnings. Dating back to the work of Cauchy, Piola, and Kirchhoff, and later extended to dynamic settings, hyperelasticity provides a thermodynamically consistent and frame-invariant description of large deformations. In this framework, the Cauchy stress tensor naturally decomposes into spherical and deviatoric contributions: the spherical part reduces to the hydrodynamic pressure described by an equation of state, while the deviatoric part represents elastic shear. This formulation introduces an additional variable associated with the shear energy, together with a corresponding evolution equation required for closure.

In this thesis, we adopt the Lagrangian perspective to formulate the governing equations of

nonlinear hyperelasticity. This system may be viewed as a direct extension of the Euler equations in the Lagrangian frame (2.0.1), previously introduced in section 1.4.1, in which the isotropic pressure is generalized to the full stress tensor and the closure is provided through the shear energy equation. Building on the robust invariant-domain preserving Lagrangian framework developed earlier for compressible flows, we extend the formulation to nonlinear hyperelasticity, thereby unifying the treatment of compressible fluids and elastic solids within a consistent model.

5.2 Equations of Hyperelasticity Formulated in Lagrangian Coordinates

Having introduced the Euler equations in a purely Lagrangian description, we now turn to the extension required for modeling nonlinear hyperelastic materials. The transition from a compressible fluid to a hyperelastic solid is conceptually straightforward: the isotropic pressure term in the fluid stress is replaced by a more general Cauchy stress tensor that captures elastic response to deformation. The momentum balance equation maintains the same Lagrangian form as in fluid dynamics, but the constitutive relation is modified to account for the strain energy stored in the solid.

Unlike fluids, where the pressure can be expressed as a function of the density and internal energy through an equation of state, hyperelastic solids require the introduction of a stored energy function that depends on the deformation gradient. The resulting stress tensor depends nonlinearly on the deformation, and its evolution introduces an additional degree of freedom associated with shear energy. To obtain a closed system of equations, we therefore supplement the mass, momentum, and energy balances with an additional evolution equation for the shear energy. This ensures that both volumetric and deviatoric (shear) contributions to the material response are consistently represented.

We formulate the equations of nonlinear hyperelasticity in Lagrangian coordinates. As in the compressible Euler case, the dependent variable is

$$\boldsymbol{u} = (\tau, \boldsymbol{v}, E)^T \in \mathbb{R}^{d+2},$$

where τ denotes the specific volume, \mathbf{v} the velocity, and E the specific total energy. The motion of the material is described by the mapping $\Phi(\xi, t)$ from the reference configuration D^0 to the deformed configuration D^t , with

$$\partial_t \Phi(\xi, t) = \mathbf{v}(\Phi(\xi, t), t),$$

and $\rho = 1/\tau$ the density.

The balance laws take the same form as in the compressible Euler system,

$$\frac{1}{\tau(\Phi(\xi, t), t)} \partial_t (\mathbf{u}(\Phi(\xi, t), t)) + (\nabla \cdot \mathbf{f}(\mathbf{u}))(\Phi(\xi, t), t) = \mathbf{0},$$

but the flux is now modified to account for elastic response

$$\mathbf{f}(\tau, \mathbf{v}, E) = (-\mathbf{v}, \boldsymbol{\sigma}(\mathbf{u}), \boldsymbol{\sigma}\mathbf{v})^\top, \quad (5.2.1)$$

where $\boldsymbol{\sigma}(\mathbf{u})$ is the Cauchy stress tensor. In the purely fluid case, $\boldsymbol{\sigma} = p\mathbb{I}_d$ recovers the standard Euler flux; in the hyperelastic setting, $\boldsymbol{\sigma}$ incorporates both spherical (pressure) and deviatoric (shear) contributions derived from a stored energy potential.

This generalization introduces an additional internal variable associated with the shear energy, together with a corresponding evolution equation required for closure of the system. In what follows, we describe the specific form of the hyperelastic stress and the closure relation used in our formulation.

5.3 Closure of the System

To close the system a constitutive relation must be established between the stress tensor $\boldsymbol{\sigma}$ and the kinematic variables. We begin by recalling the deformation gradient, which we originally introduced in (1.4.4)

$$\mathbb{J} = \frac{\partial \Phi(\xi, t)}{\partial \xi},$$

which is the gradient of the Lagrange map, which maps the reference configuration to the current configuration. From \mathbb{J} , one defines the right Cauchy–Green strain tensor

$$\mathbf{C} := \mathbb{J}^T \mathbb{J}. \quad (5.3.1)$$

Although the left Cauchy–Green tensor $\mathbf{B} = \mathbb{J} \mathbb{J}^T$ is frequently used in the hydrocode literature, in the isotropic setting this distinction is immaterial. In this work we adopt \mathbf{C} , which is more naturally extended to problems involving anisotropy and plasticity.

Remark 5.3.1 (Nonlinearity). *Equation (5.3.1) highlights why the hyperelastic models considered in this work are inherently nonlinear: the strain tensor depends on \mathbf{C} , which itself depends quadratically on the deformation gradient \mathbb{J} .*

It is natural to assume that the specific total energy admits a separable decomposition,

$$E(\mathbf{c}, \eta) = \frac{1}{2} |\mathbf{v}|^2 + e^h(\rho, \eta) + e^s(\mathbf{c}), \quad (5.3.2)$$

where the first term represents the kinetic energy, $e^h(\rho, \eta)$ denotes the volumetric (hydrodynamic) contribution to the internal energy and is given by the equation of state (see section 1.5.1), and $e^s(\mathbf{c})$ accounts for the shear contribution. The shear energy depends on the reduced Cauchy–Green strain tensor,

$$\mathbf{c} := \frac{\mathbf{C}}{(\det \mathbf{C})^{1/3}}, \quad (5.3.3)$$

which isolates the deviatoric part of the deformation. Since the tensor \mathbf{c} has unit determinant, we note that it is unaffected by volume change. In the following models, the specific internal energy is taken to be a function of ρ , \mathbf{c} , and s . This model was first introduced in [41]

We introduce the *Cauchy stress tensor*

$$\boldsymbol{\sigma} := 2\rho \mathbb{J} \frac{\partial E}{\partial \mathbf{C}} \mathbb{J}^T, \quad (5.3.4)$$

which is symmetric if E is an isotropic function.

5.3.1 Isotropic Closure

We begin by considering media which have the same material properties in all directions (isotropic media). In this case, the shear energy depends only on invariants of the reduced strain tensor \mathbf{c}

$$j_k := \text{Tr}(\mathbf{c}^k), \quad k \in \{1, 2\}, \quad (5.3.5)$$

With those defined, the shear energy is then expressed as

$$e^s = e^s(j_1, j_2). \quad (5.3.6)$$

It is shown in [41, Eq. 10] that

$$\boldsymbol{\sigma} = -\rho^2 \frac{\partial e^h}{\partial \rho} \mathbb{I} + 2\rho \mathbb{J} \frac{\partial e^s}{\partial \mathbf{C}} \mathbb{J}^\top, \quad (5.3.7)$$

where \mathbb{I} is the $d \times d$ identity matrix. Following a discussion presented in section D.3, one arrives at the following representation

$$\boldsymbol{\sigma} = 2\rho \left[\frac{\partial e^s}{\partial j_1} \left(\mathbf{b} - \frac{1}{3} j_1 \mathbb{I} \right) + 2 \frac{\partial e^s}{\partial j_2} \left(\mathbf{b}^2 - \frac{1}{3} j_2 \mathbb{I} \right) \right] - \rho^2 \frac{\partial e^h}{\partial \rho} \mathbb{I}, \quad (5.3.8)$$

where

$$\mathbf{b} = \frac{\mathbf{B}}{(\det \mathbf{C})^{1/3}}. \quad (5.3.9)$$

The first two terms of 5.3.8 represent the deviatoric (shear) response, while the last term corresponds to the volumetric pressure.

A simple choice for the shear energy is the Neo-Hookean model [74] for incompressible solids,

$$e^s = \frac{\mu}{2\rho_0} (j_1 - 3), \quad (5.3.10)$$

where μ is the shear modulus. A more sophisticated alternative is the Mooney–Rivlin type model [67, 74], which depends nonlinearly on the invariants. In the form used for comparison with [35], the energy reads

$$e^s = \frac{\mu}{32\rho_0} (j_1^4 - 2j_2j_1^2 + j_2^2 - 8j_1 - 12), \quad (5.3.11)$$

These examples illustrate how different constitutive choices for e^s produce different shear responses, while the general structure (5.3.8) remains unchanged.

5.3.2 Anisotropic Closure

Many biological and composite materials exhibit anisotropy due to fibers or structural reinforcements that align along preferred directions—for example, the grain in wood, collagen/elastin fiber families in arterial walls and myocardium, tendons and ligaments, and engineered carbon-fiber laminates. Following the invariant-based framework of Schröder and Neff [76], such anisotropy can be incorporated by introducing unit vectors

$$\mathbf{m}_i := (\cos \theta_i, \sin \theta_i, 0), \quad i = 1, \dots, N, \quad (5.3.12)$$

in the reference configuration, each oriented at an angle $\theta_i \in [0, \pi)$ corresponding to the fiber direction. See figure 5.1 for various examples of fiber orientations employed in this work. From each unit vector, we define structural tensors

$$\mathbf{G}_i := \mathbf{m}_i \otimes \mathbf{m}_i, \quad (5.3.13)$$

which encode the contribution of the i th fiber orientation. This approach ensures that the constitutive equations respect material frame indifference and the underlying material symmetry group while allowing for polyconvex free energy functions that guarantee well-posedness of the variational problem.

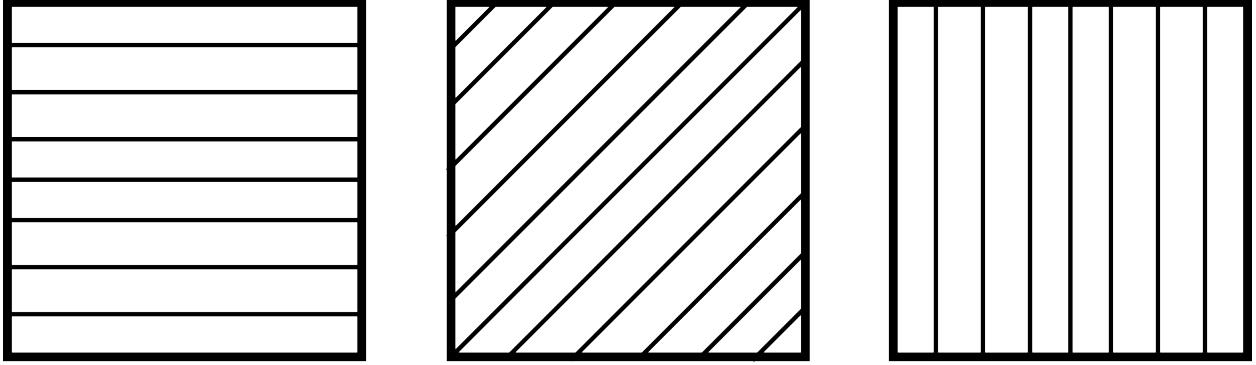


Figure 5.1: Examples of various fiber alignments. Left: $\theta = 0$, middle: $\theta = \frac{\pi}{4}$, right: $\theta = \frac{\pi}{2}$.

The structural tensors enter the stored energy through additional mixed invariants,

$$j_{4,i} = \text{Tr}(\mathbf{c}\mathbf{G}_i) \quad \text{and} \quad j_{5,i} = \text{Tr}(\mathbf{c}^2\mathbf{G}_i), \quad (5.3.14)$$

which augment the isotropic invariants of \mathbf{c} and capture the influence of fiber families. By superposing the contributions of multiple \mathbf{G}_i , the framework naturally accommodates materials with an arbitrary number of fiber orientations, though in this thesis we present results using a single fiber orientation. The corresponding Cauchy stress tensor can be expressed as

$$\begin{aligned} \boldsymbol{\sigma} = & 2\rho \left[\frac{\partial e^s}{\partial j_1} \left(\mathbf{b} - \frac{1}{3}j_1\mathbb{I} \right) + 2 \frac{\partial e^s}{\partial j_2} \left(\mathbf{b}^2 - \frac{1}{3}j_2\mathbb{I} \right) \right] - \rho^2 \frac{\partial e^h}{\partial \rho} \mathbb{I} \\ & + 2\rho \left[\frac{\partial e^s}{\partial j_{4,i}} \left(\frac{\mathbb{J}\mathbf{G}_i\mathbb{J}^T}{|\mathbf{C}|^{1/3}} - \frac{1}{3}j_{4,i}\mathbb{I} \right) + 2 \frac{\partial e^s}{\partial j_{5,i}} \left(\frac{\mathbb{J}\mathbf{C}\mathbf{G}_i\mathbb{J}^T}{|\mathbf{C}|^{2/3}} - \frac{1}{3}j_{5,i}\mathbb{I} \right) \right]. \end{aligned} \quad (5.3.15)$$

Building on this general framework, we adopt the phenomenological model proposed by Chaimoon and Chindaprasirt [15], developed for the coronary arteries. Their model decomposes the shear energy into isotropic and anisotropic parts and takes the form

$$\begin{aligned} e^s = & (1 - 2w_1^1) \left(\frac{A_1}{2}(j_1 - 3) + \frac{B_1}{4}(j_1^2 - j_2 - 6) \right) \\ & + 2w_1^1 D_1 \left(\exp\{j_{4,i} - 1\} + \exp \left\{ j_{5,i} - j_1 j_{4,i} + \frac{j_1^2 - j_2}{2} - 1 \right\} - 2 \right) \end{aligned} \quad (5.3.16)$$

where the anisotropic constants are chosen as

$$A_1 = 4.815 \times 10^5, \quad B_1 = 4.815 \times 10^5, \quad D_1 = 7.2225 \times 10^5, \quad w_1^1 = 0.49.$$

The first term in (5.3.16) represents the isotropic contribution and the exponential terms model the nonlinear stiffening due to fiber reinforcement. This form captures the nonlinear stiffening of collagen fibers at large strains while remaining consistent with polyconvexity and natural state conditions, ensuring material stability even in compression.

The closure of the hyperelastic system is achieved by prescribing a stored energy function, which defines the constitutive model and from which the stress tensor is derived. The shear contribution to the energy e^s depends on invariants of the strain tensor (and, in more sophisticated models, additional material parameters such as fiber orientations). A summary of the elastic equations of state employed in this work is provided in Table 5.1. The examples presented above illustrate both isotropic and anisotropic responses; in each case, the volumetric and shear contributions combine to yield a thermodynamically consistent model that extends the Euler equations to the nonlinear elastic regime.

Table 5.1: Reference equations of state for shear energy.

Elastic energy equations of state	
Neo Hookean	$e^s = \frac{\mu}{2\rho_0} (Tr(\mathbf{c}) - 3)$
Mooney Rivlin	$e^s = \frac{\mu}{32\rho_0} (i_1^4 - 2i_2i_1^2 + i_2^2 - 8i_1 - 12)$
Aortic	$e^s = (1 - 2w_1^1) \left(\frac{A_1}{2}(j_1 - 3) + \frac{B_1}{4} (j_1^2 - j_2 - 6) \right) + 2w_1^1 D_1 [\exp\{j_{4,i} - 1\} + \exp \left\{ j_{5,i} - j_1 j_{4,i} + \frac{j_1^2 - j_2}{2} - 1 \right\} - 2]$

5.4 Extension of the Robust Low-Order Framework to Nonlinear Elasticity

The robust low-order invariant-domain preserving framework introduced in Chapter 3 was originally developed for the compressible Euler equations in a purely hydrodynamic setting. To extend this framework to nonlinear hyperelasticity, only modest modifications are required. The general algorithmic structure—including the treatment of mass, momentum, and energy transport, as well as the invariant-domain preserving artificial viscosity defined based on the local riemann problem—remains unchanged. The primary augmentation lies in the replacement of the isotropic pressure term with the full Cauchy stress tensor, together with the inclusion of the additional variables and evolution equations needed to represent shear contributions.

Since the low-order scheme operates with cell-averaged quantities, an additional consideration arises in how to treat the kinematic variables that define the constitutive response. One may either compute cell averages of the deformation gradient \mathbb{J} , or alternatively compute averages of the Cauchy–Green strain tensor \mathbf{C} . In this work, we choose to average \mathbb{J} in order to accommodate both isotropic and anisotropic closures, as the anisotropic model depends explicitly on \mathbb{J} . We note, however, that for isotropic models discussed in section 5.3.1, no discernible difference has been observed in our implementation between averaging \mathbb{J} or \mathbf{C} . The complete process for updating the hydrodynamics of our system is outlined in the pseudocode in Algorithm 5.

This extension preserves the robustness and stability properties of the original scheme while enabling the accurate simulation of both compressible fluids and elastic solids within a unified computational framework.

Remark 5.4.1 (Computation of λ^{\max}). *In the case of incompressible solids, as is the subject of this chapter, the computation of the maximum wave speed requires further development and is part of ongoing work. For the moment, we adopt a modified definition based on the fluid formulation,*

$$\lambda_{L,R}^{\max} = \sqrt{\hat{\lambda}_{L,R}^2 + \frac{4}{3} \max\{\rho_L, \rho_R\} \mu}, \quad (5.4.1)$$

where $\hat{\lambda}_{L,R}$ denotes the maximum wave speed associated with the local Riemann problem between

the left and right states \mathbf{U}_L and \mathbf{U}_R , and μ is the shear modulus. This modification serves as a provisional model pending a more rigorous derivation.

6. NUMERICAL RESULTS

6.1 Technical Details

The time step at each time instant t^n is defined by

$$\Delta t^n := \frac{\text{CFL}}{2} \min_{c \in \mathcal{N}^{\text{Cel}}} \frac{m_c}{\sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n}, \quad (6.1.1)$$

where $\text{CFL} \in [0, 1]$ is the Courant-Friedrichs-Levy number chosen by the user. When the exact solution of a specific test case is known, we measure the relative errors by computing

$$\delta^1(t) := \frac{\|\tau_h(t) - \tau(t)\|_{L^1(D)}}{\|\tau(t)\|_{L^1(D)}} + \frac{\|\mathbf{v}_h(t) - \mathbf{v}(t)\|_{L^1(D)}}{\|\mathbf{v}(t)\|_{L^1(D)}} + \frac{\|E_h(t) - E(t)\|_{L^1(D)}}{\|E(t)\|_{L^1(D)}}. \quad (6.1.2)$$

In addition, for the results presented in the nonlinear hyperelasticity regime in section 6.4, we measure the error in the Cauchy stress tensor as

$$\delta_\sigma^1(t) := \frac{\sum_i |\sigma_{i,i}^h(t) - \sigma_{i,i}(t)|}{\sum_i |\sigma_{i,i}(t)|}. \quad (6.1.3)$$

The code is based on the miniapp Laghos (LAGrangian High-Order Solver) developed at LLNL. The Laghos miniapp is based on [28] and is part of the Center for Efficient Exascale Discretizations software suite <http://github.com/ceed>. We use Q_0 approximation for the state variable and, unless stated otherwise, we use Q_1 continuous finite elements for the mesh velocity. Unless stated otherwise, we systematically use the three-step method described §3.5.1. The method is exactly mass conservative. The L^1 -norm of the indicator $\delta^1(t)$ is estimated on quadrangles by using a Gaussian quadrature rule of order 3 with 2×2 Gauss points.

See our paper [45] for the results and implementation using triangles.

6.2 Low Order Results*

*A majority of this section is a modification of the work in [45] and is reprinted with permission from the authors.

We illustrate the performance of the proposed method in two space dimensions using a quadrangle based code. All the simulations are done on sequential computers. It is not the purpose of this thesis to demonstrate scalability.

6.2.1 Sod Shocktube

In this section we start evaluating the method on the Sod shocktube. We solve the problem in two dimensions using the computational domain $D^0 = (0, 1) \times (0, 0.1)$. Using the notation $\mathbf{x} := (x_1, x_2)$, the initial condition is

$$\mathbf{u}_0(\mathbf{x}) = (\tau, \mathbf{v}, E)^\top = \begin{cases} (1, \mathbf{0}, 2.5)^\top & \text{if } 0 < x_1 \leq 0.5, \\ (8, \mathbf{0}, 2)^\top & \text{if } 0.5 < x_1 < 1. \end{cases}$$

Dirichlet boundary conditions are applied to the left and right boundaries. Slip boundary conditions are enforced on the upper and lower boundaries. We perform four simulations: one on the initial mesh consisting of 200×10 squares, and three additional simulations on subdivided refined meshes. The simulations run up to $t = 0.225$ with $\text{CFL} = 0.5$.

Table 6.1: Convergence tests for the Sod shocktube problem. Reprinted with permission from [45].

# cells	$\delta^1(0.225)$	order
2000	3.747E-2	—
8000	2.378E-2	0.66
32000	1.443E-2	0.72
128000	8.839E-3	0.71

Table 6.1 displays the error indicator $\delta^1(0.225)$ and corresponding convergence orders. The method converges as expected.

6.2.2 Distorted Sod Shocktube

We now assess the capacity of the method to preserve one-dimensional waves on distorted meshes. We also compare the accuracy and efficiency of the method ensuring mass conservation

by optimization with the method ensuring mass conservation by post-processing. We solve again the Sod problem on $D^0 = (0, 1) \times (0, 0.1)$, but this time the initial uniform Cartesian mesh is distorted by using the following mapping: $(x, y) \mapsto (x, y + x(1 - x)\sin(4\pi x)/20)$.

Table 6.2: Convergence tests for the distorted Sod shocktube. Reprinted with permission from [45].

# dof	$\delta^1(0.225)$	order
500	5.985E-2	–
2000	3.786E-2	0.66
8000	2.418E-2	0.65
32000	1.470E-2	0.72

In Table 6.2 we show the errors and the convergence orders on the distorted meshes. We observe that despite distorting the mesh, the error and convergence orders remain the same as in Table 6.1. See Table B.1 in Appendix B for a comparison of the alternate mesh velocity approaches. In addition, see Figure B.2 in Section B.4 for an illustration of premature mesh collapse caused by the face-correction mass-conservation method.

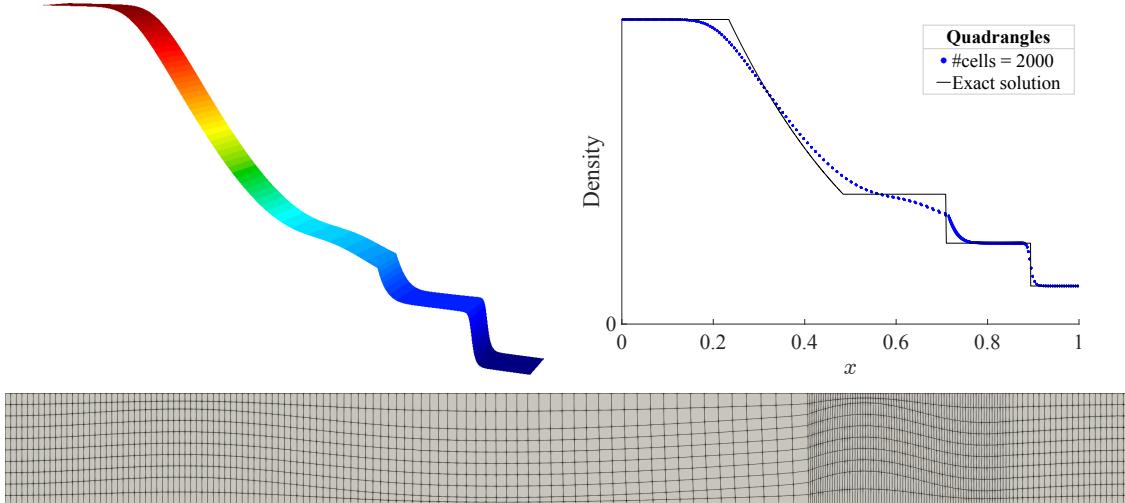


Figure 6.1: Sod shocktube on an unstructured mesh at $t = 0.225$. Top left: 3D view of the density field. Top right: scatter plot of the density. Bottom: final mesh configuration. Reprinted with permission from [45].

In Figure 6.1, we show the mesh and the graph of the density at $t = 0.225$. This test demonstrates well that the proposed method effectively preserves the one-dimensional nature of the solution.

6.2.3 Isentropic Vortex

To access the accuracy of the method when applied to a moving mesh with large vortex-induced deformations, we test it on the isentropic vortex problem. The exact solution for this problem is given by

$$\rho(\mathbf{x}, t) = (T_\infty + \delta T)^{1/(\gamma-1)}, \quad \mathbf{v}(\mathbf{x}, t) = \mathbf{v}_\infty + \delta \mathbf{v}, \quad p(\mathbf{x}, t) = \rho^\gamma, \quad (6.2.1)$$

with the free-stream conditions $\rho_\infty = 1$, $p_\infty = 1$, $T_\infty = 1$, $\mathbf{v}_\infty = \mathbf{0}$, and the perturbations

$$\delta \mathbf{v}(\mathbf{x}, t) = \frac{\beta}{2\pi} e^{\frac{1-r^2}{2}} (-\bar{x}_2, \bar{x}_1), \quad \delta T(\mathbf{x}, t) = -\frac{(\gamma-1)\beta^2}{8\gamma\pi^2} e^{1-r^2}, \quad (6.2.2)$$

where $\bar{\mathbf{x}} := \mathbf{x} - \mathbf{x}_0 - \mathbf{v}_\infty t$, and $r := \|\bar{\mathbf{x}}\|_{\ell^2}$ is the Euclidean distance from the vortex center. All the numerical simulations are done with $\beta = 5$, $\mathbf{x}_0 = (0, 0)$ and $\gamma = \frac{7}{5}$. The initial computational domain is $D^0 = (-5, 5)^2$. The density, velocity, and total specific energy are maintained at the free-stream values on the boundary of the computational domain at all times.

The initial mesh is composed of 16×16 squares. Six simulations are done: one on the initial mesh and five on meshes obtained by uniform refinement. The CFL number is set to 0.5, and the final time is $t = 1$.

The convergence results are summarized in Table 6.3. We observe that the L^1 -error indicator converges with a rate close to 1.

6.2.3.1 Long Time Integration

We now run the vortex simulation until $t = 15$ to assess the ability of the scheme to move the mesh without tangling. Figure 6.2 presents the density field and the final mesh for a mesh composed of 65536 cells. This test demonstrates that the low-order Lagrangian numerical scheme

Table 6.3: Composite L^1 -error for the isentropic vortex problem at $t = 1$ with $CFL = 0.5$. Reprinted with permission from [45].

# cells	$\delta^1(1)$	order
256	1.894E-1	–
1024	1.205E-1	0.65
4096	6.930E-2	0.80
16384	3.744E-2	0.89
65536	1.950E-2	0.94
262144	9.958E-3	0.97

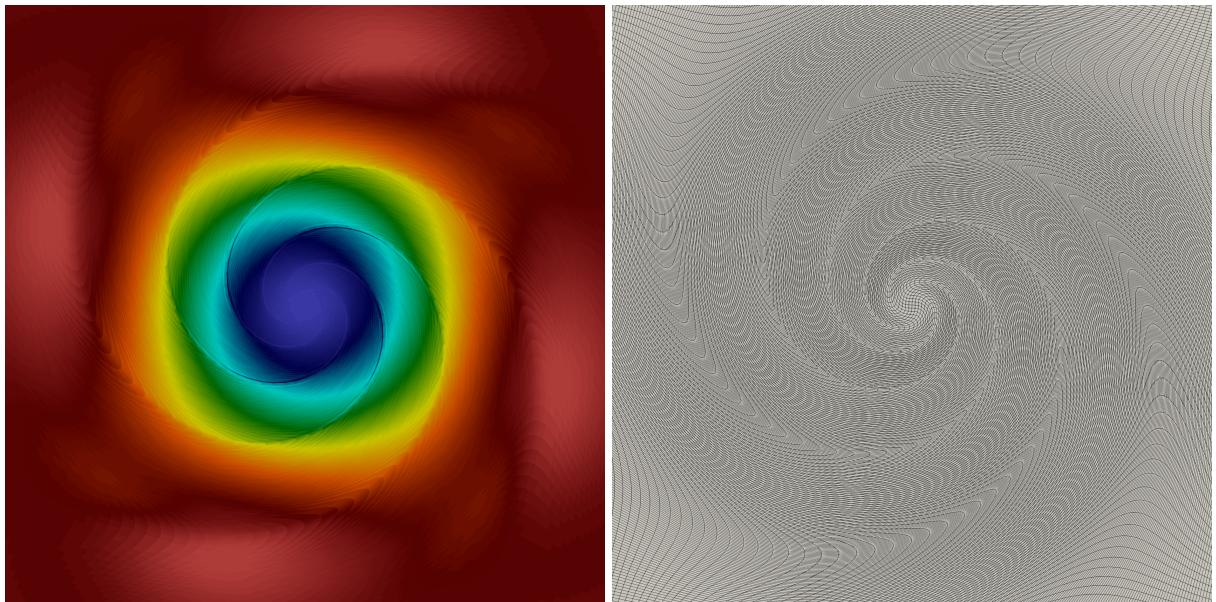


Figure 6.2: Density field and mesh configuration at $t = 15$ for the isentropic vortex on a mesh composed of 65536 cells. Reprinted with permission from [45].

effectively adapts the mesh to the vortex flow structure while maintaining its overall integrity. Note however that this is not the case for all mesh velocity computations that we explored. See Figure B.2 in Section B.4 for an illustration of premature mesh collapse caused by the face-correction mass-conservation method on the isentropic vortex.

6.2.4 Taylor-Green Vortex

The goal of this example is to verify convergence in space and time on a moving Lagrangian mesh for a smooth (shock-free) problem. We consider the classical Taylor–Green vortex, a steady solution of the incompressible Navier-Stokes equations on the unit square. We define

$$\mathbf{v}(x, y) = (\sin(\pi x) \cos(\pi y), -\cos(\pi x) \sin(\pi y)), \quad p(x, y) = \frac{\rho}{4} (\cos(2\pi x) + \cos(2\pi y)) + 1,$$

where the velocity is defined to be divergence free $\nabla \cdot \mathbf{v} = 0$ and the pressure is defined to satisfy the momentum equation $\rho d_t \mathbf{v} + \nabla p = 0$.

To use this field with the compressible Euler equations and an ideal-gas EOS with adiabatic index $\gamma = 5/3$, we adopt a manufactured-solution approach. Since $\nabla \cdot \mathbf{v} = 0$, the density remains constant in both space and time and we use $\rho = 1$. Since the pressure was defined to satisfy the momentum equation exactly, no external body force source term is necessary. All that is required is a source term for the equation of total energy balance. Writing $e = p/((\gamma - 1)\rho)$, the needed source is

$$S_E(x, y) = \rho \frac{de}{dt} + p \nabla \cdot \mathbf{v} = \rho \mathbf{v} \cdot \nabla e = \frac{1}{\gamma - 1} \mathbf{v} \cdot \nabla p.$$

For $\gamma = 5/3$ and $\rho = 1$, using a standard trigonometric identity this reduces to

$$\begin{aligned} S_E(x, y) &= \frac{3}{2} \begin{pmatrix} \sin(\pi x) \cos(\pi y) \\ -\cos(\pi x) \sin(\pi y) \end{pmatrix} \cdot \begin{pmatrix} -\frac{\pi}{2} \sin(2\pi x) \\ -\frac{\pi}{2} \sin(2\pi y) \end{pmatrix} \\ &= -\frac{3\pi}{4} (\sin(\pi x) \sin(2\pi x) \cos(\pi y) - \cos(\pi x) \sin(\pi y) \sin(2\pi y)) \\ &= \frac{3\pi}{8} (\cos(3\pi x) \cos(\pi y) - \cos(\pi x) \cos(3\pi y)). \end{aligned}$$

The computational domain is $[0, 1] \times [0, 1]$ with slip-wall boundary conditions $\mathbf{v} \cdot \mathbf{n} = 0$ on all sides. Although the exact fields are steady, they vary along particle trajectories, so the moving mesh undergoes nontrivial deformation in time. Errors and observed convergence rates at the final time $t_f = 0.5$ with CFL number 0.5 are reported in Table 6.4. The velocity at initial and final time

along with the mesh is shown in Figure 6.3.

Table 6.4: Error and convergence rates for the Taylor-Green Vortex problem at final time $t_f = 0.5$. Reprinted with permission from [45].

# cells	Q_1 mesh velocity	
	$\delta^1(0.5)$	order
256	1.509E-1	–
1024	1.050E-1	0.52
4096	6.585E-2	0.67
16384	3.774E-2	0.80
65536	2.040E-2	0.89
262144	1.068E-2	0.93

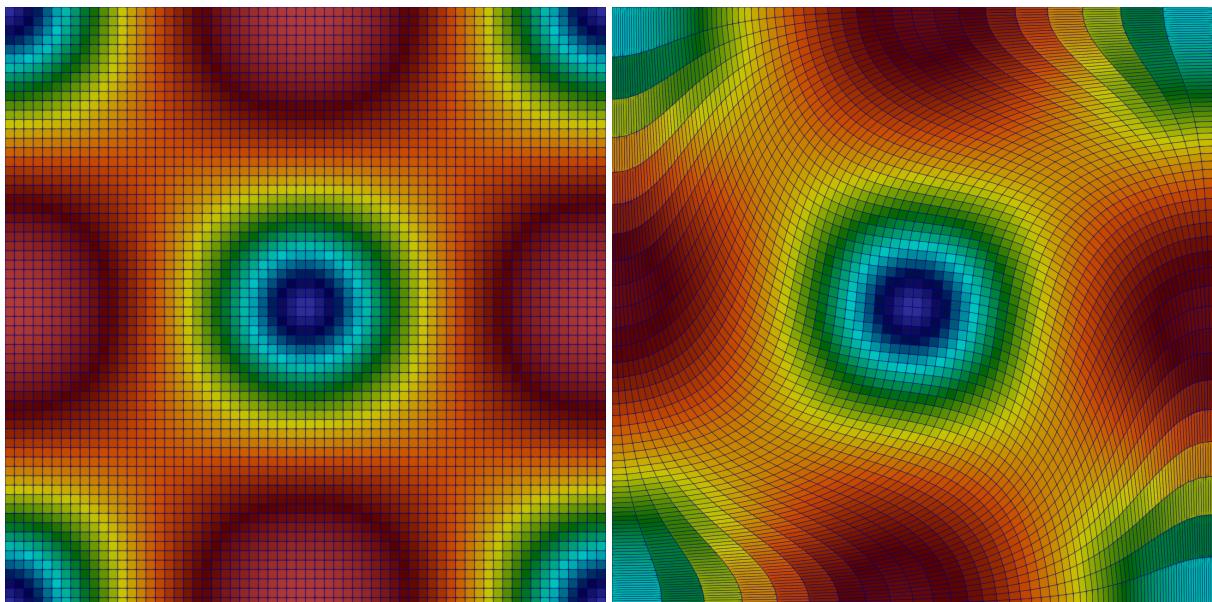


Figure 6.3: Velocity field and mesh at $t = 0$ and $t_f = 0.5$ for the Taylor-Green Vortex. Reprinted with permission from [45].

6.2.5 Kidder Problem

The Kidder problem examines the isentropic compression of a spherical shell filled with a perfect gas and is a simplified Intertial Confinement Fusion (ICF) like test. The initial conditions are specifically designed to ensure smooth flow dynamics. It is particularly suited for evaluating the accuracy, stability, and convergence of methods under idealized, symmetric conditions. The initial computational domain is a shell bounded by $r_i \leq r(\mathbf{x}) \leq r_e$, where r_i and r_e are the internal and external radii at $t = 0$ and $r(\mathbf{x}) := \sqrt{x_1^2 + x_2^2}$.

The initial density and pressure distributions are defined by

$$\rho^0(\mathbf{x}) = \left(\frac{(r_e)^2 - (r(\mathbf{x}))^2}{(r_e)^2 - (r_i)^2} \rho_i^{\gamma-1} + \frac{(r(\mathbf{x}))^2 - (r_i)^2}{(r_e)^2 - (r_i)^2} \rho_e^{\gamma-1} \right)^{\frac{1}{\gamma-1}}, \quad p^0(\mathbf{x}) = \rho^0(\mathbf{x})^\gamma. \quad (6.2.3)$$

The initial radial velocity is $u^0 = 0$, and the specific entropy is $s^0 = 1$. At time t , the initial domain D^0 is compressed into $D_t = \Phi(D^0, t)$, where $\Phi(\xi, t) := \xi h(t)$, $h(t) = \sqrt{1 - \frac{t^2}{\tau^2}}$, and τ is given by

$$\tau = \sqrt{\frac{(\gamma - 1)(r_e^2 - r_i^2)}{2(c_e^2 - c_i^2)}}, \quad \text{with } c = \sqrt{\gamma p / \rho}. \quad (6.2.4)$$

The analytical solution for the density, pressure and radial velocity at $\mathbf{x} \in D_t$, is

$$\rho(\mathbf{x}, t) = \rho^0(\Phi_t^{-1}(\mathbf{x})) h(t)^{\frac{-2}{\gamma-1}}, \quad p(\mathbf{x}, t) = p^0(\Phi_t^{-1}(\mathbf{x})) h(t)^{\frac{-2\gamma}{\gamma-1}}, \quad u(\mathbf{x}, t) = \Phi_t^{-1}(\mathbf{x}) \frac{dh(t)}{dt}. \quad (6.2.5)$$

In the simulations, we use $r_i = 0.9$, $r_e = 1$, $\rho_i = 1$ and $\rho_e = 2$. The gas follows a polytropic equation of state with $\gamma = 2$. The boundary conditions are enforced by modifying the numerical flux in (3.1.5) at the boundary faces. Let F be a boundary face, let K_c be the cell that contains F , and let \mathbf{x}_F be the barycenter of F . Instead of setting $\hat{\mathbf{f}}_F = \mathbf{f}(\mathbf{U}_c^n)$ in (3.1.5), the numerical flux at F is defined by

$$\hat{\mathbf{f}}_F := \begin{pmatrix} -\mathbf{V}_c \\ p(\mathbf{x}_F, t^n) \\ p(\mathbf{x}_F, t^n) \mathbf{V}_c \end{pmatrix}. \quad (6.2.6)$$

If F is part of the external shell we have $p(\mathbf{x}_F, t^n) = \rho_e^\gamma = 4$, and if F is part of the internal shell we have $p(\mathbf{x}_F, t^n) = \rho_i^\gamma = 1$. The simulations are run up to $t_f = \frac{\sqrt{3}}{2}\tau$, at which time the shell is expected to be the ring $\{\mathbf{x} \in \mathbb{R}^2 \mid 0.45 \leq r(\mathbf{x}) \leq 0.5\}$. The computational domain is discretized with mapped Cartesian meshes composed of $N_1 \times (20N_1)$ squares.

Convergence tests are done on four meshes with $N_1 \in \{5, 10, 20, 40\}$. Table 6.5 shows the convergence rates and the $\delta^1(t_f)$ relative errors. We observe that the method converges with the expected order.

Table 6.5: Kidder problem. Error indicator $\delta^1(t_f)$ and convergence order obtained on different meshes. Reprinted with permission from [45].

# cells	$\delta^1(t_f)$	order
500	8.364E−2	—
2000	4.546E−2	0.88
8000	2.359E−2	0.94
32000	1.209E−2	0.96

Figure 6.4 shows the initial and final meshes, along with the corresponding density distribution, illustrating that the mesh boundary approximation does not introduce a significant deterioration in solution quality. Figure 6.5 shows the evolution of the mean values of the external and internal numerical radii, computed with the two finest meshes, compared with their exact location given by $r_e^0 h(t)$ and $r_i^0 h(t)$, respectively. Our results are consistent with those reported in the literature, particularly in the work of [64]. A discrepancy is also observed between the numerical and analytical density profiles at the final time, which decreases with mesh refinement, indicating it is likely due to numerical diffusion.

6.2.6 Saltzman Problem

We now consider the so-called Saltzman test (see [29, §2.2]). It is a shocktube problem in a two-dimensional rectangular box, $(0, 1) \times (0, 0.1)$ with $\gamma = \frac{5}{3}$. It is a benchmark for evaluating the robustness of Lagrangian schemes against hourglass instabilities. It consists of simulating only

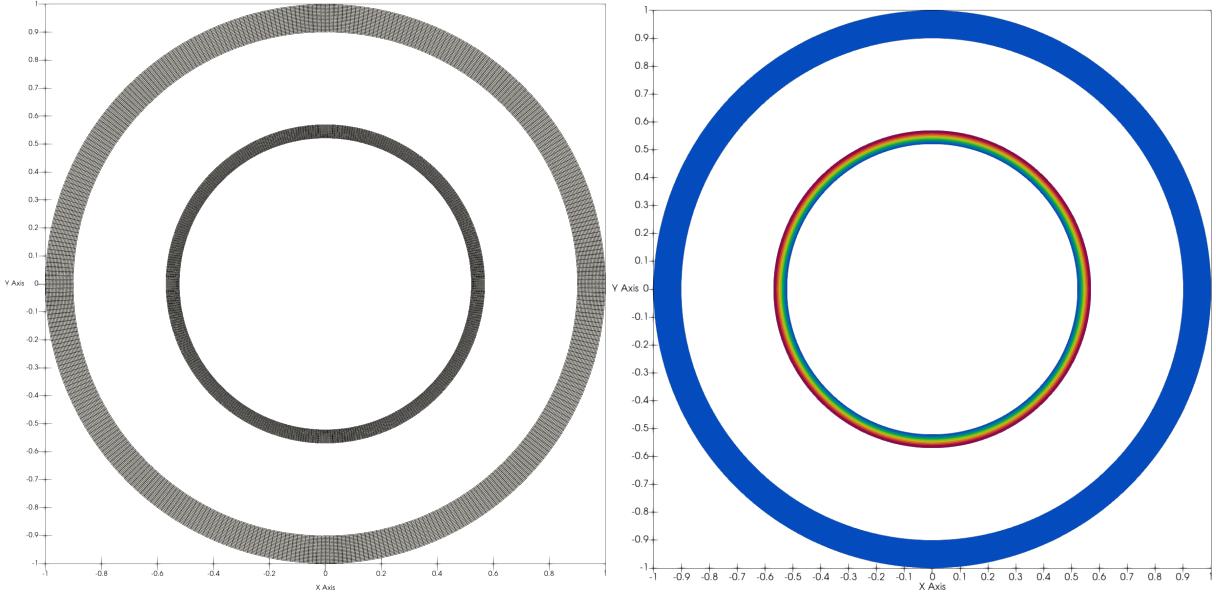


Figure 6.4: Density field and mesh configuration at $t = 0$ and $t = t_f$ for the Kidder problem on a mesh composed of 2000 cells. Reprinted with permission from [45].

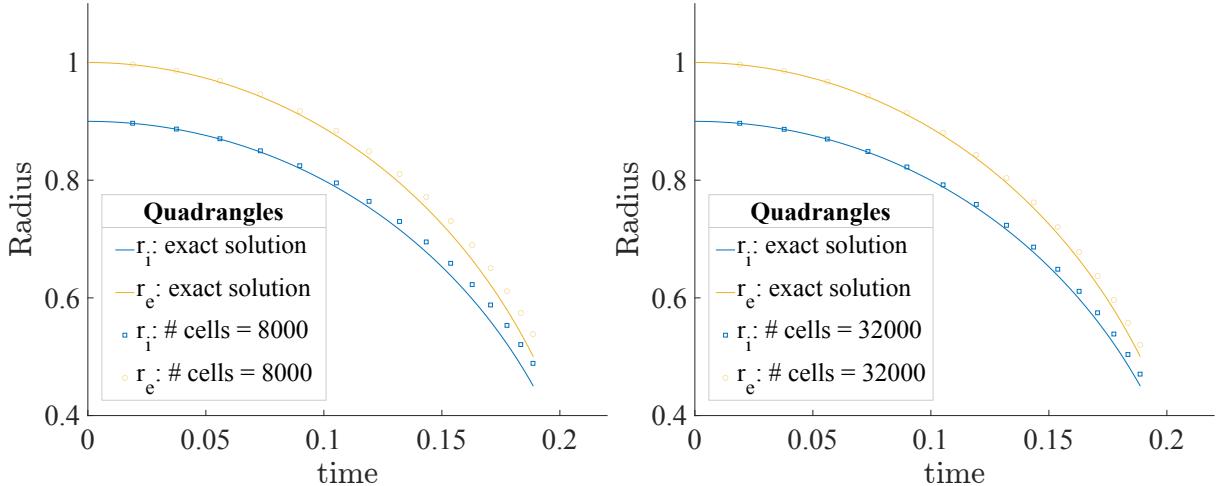


Figure 6.5: Kidder problem. Time evolution of the external and internal radius of the shell versus the analytical solution. Reprinted with permission from [45].

half of a Riemann problem by replacing the contact wave with a wall moving at speed 1. The wall is initially located at $x = 0$. The initial data is $\mathbf{u}_0 = (1, \mathbf{0}, 10^{-5})^\top$, and the boundary condition is $\mathbf{v} = \mathbf{e}_x$ at the boundary $x = t$ and $\mathbf{v} \cdot \mathbf{n} = 0$ otherwise.

To assess the robustness of the method we run two simulations. One is done using an initial

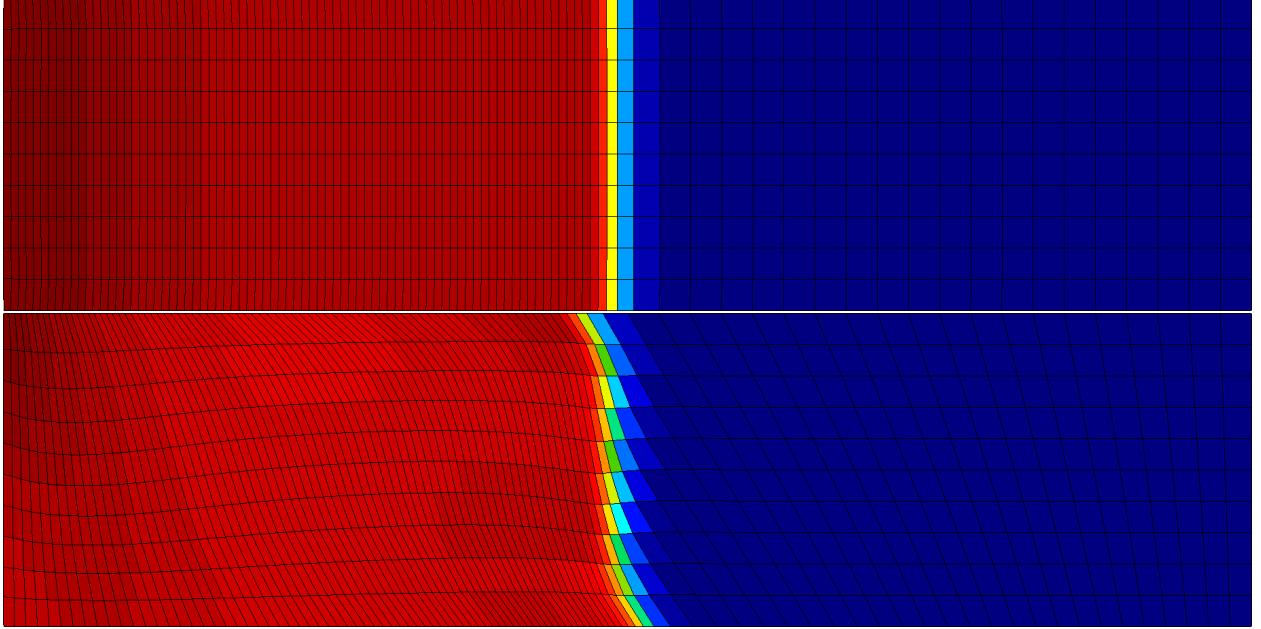


Figure 6.6: Saltzman problem. Density field and mesh configuration at $t = 0.6$. Top: Solution obtained on the uniform mesh. Bottom: Solution obtained on the distorted meshes. Reprinted with permission from [45].

mesh that is Cartesian and composed of 100×10 uniformly distributed square cells. The second one uses the Cartesian mesh transformed by applying the following mapping to all the nodes of the mesh: $(x_1, x_2) \rightarrow (x_1 + (0.1 - x_2) \sin(\pi x_1), x_2)$. This configuration increases the complexity of the Lagrangian simulation, as the shock front intersects element interfaces at oblique angles. As was done in Boscheri et al. [5], we avoid the generation of invalid elements along the left boundary where the piston is located by using $\text{CFL} = 0.01$ up until time $t = 0.01$, and then changing to $\text{CFL} = 0.5$ after that time.

Figure 6.6 shows a comparison of the density and mesh configurations at $t = 0.6$ using both the Cartesian mesh and the distorted mesh. In each case we observe that the mesh aligns with the flow to best capture the zones with high gradients.

We show in Figure 6.7 the density and y -component of the velocity as functions of the x coordinate on the perturbed mesh. The scheme is robust with respect to the bias introduced by the mesh distortion.

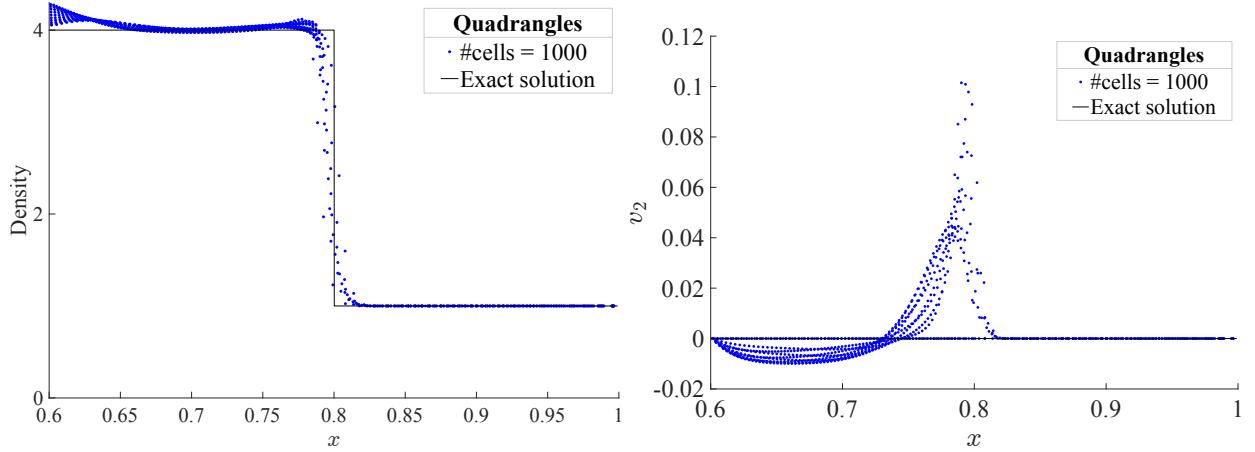


Figure 6.7: Saltzman problem on the distorted mesh. Scatter plots of the density and y -component of the velocity as functions of the x coordinate (blue dots) compared to the exact solution (black line). Left: density. Right: y -component of the velocity. Reprinted with permission from [45].

6.2.7 Van der Waals Equation of State

We consider now a Riemann problem using the van der Waals equation of state (1.5.9) as the pressure oracle. The purpose of this test is to verify that the scheme performs well when a maximal compressibility condition is enforced by the equation of state, see Remark 1.5.1. The van der Waals equation of state is defined as

$$p(\rho, e) := (\gamma - 1) \frac{\rho e + a\rho^2}{1 - b\rho} - a\rho^2,$$

where γ , a and b are constants. We set $a = 1$, $b = 1$, and $\gamma = 1.02$, and we solve the Riemann problem in a two dimensional domain $D^0 = (-1.7, 1) \times (0, 0.1)$ with the following initial condition:

$$(\rho_0, \mathbf{v}_0, p_0)^T = \begin{cases} (0.9932, 3, 2)^T & \text{if } x_1 \leq 0, \\ (0.95, -3, 2)^T & \text{if } x_1 > 0. \end{cases} \quad (6.2.7)$$

The density is chosen to be close to the maximal compressibility condition $\rho < 1 = \frac{1}{b}$. We do the computations up to $t = 0.005$. This test case is proposed in [19, Eq. 6.6]. We use $CFL = 1$. We show in Figure 6.8 the density, pressure, and sound speed for various meshes using the described

method. The results converge to the exact solution. We have observed on this series of tests that the post-processing never violates the invariant domain property.

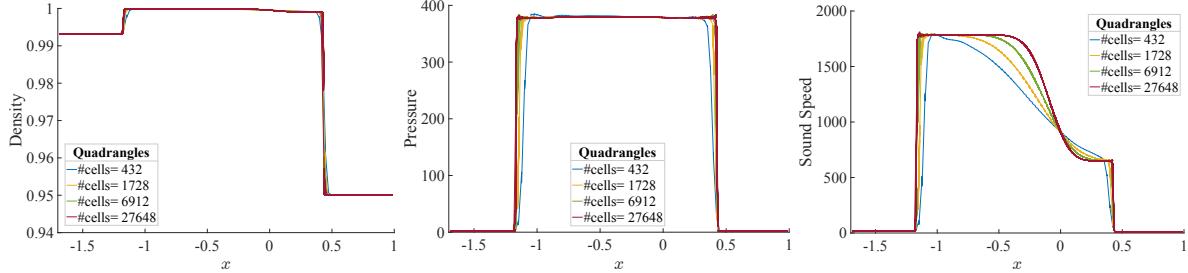


Figure 6.8: Van der Waals equation of state. Solution to the Riemann problem (6.2.7) at $t = 0.005$ with $\text{CFL} = 1$. From left to right: density; pressure; sound speed. Reprinted with permission from [45].

Remark 6.2.1 (Traditional Lagrangian hydrodynamics). *We conjecture that the test presented in this section might be challenging to traditional Lagrangian codes using the strong form of the mass update for there is no mechanism in the expression $\rho_h(\Phi_h(\xi_l, t)) = \rho_0(\xi_l) \frac{1}{\det(\nabla \Phi_h(\xi_l, t))}$ to enforce the maximal compressibility on ρ_h , see Remark 3.4.1. Similarly, traditional Lagrangian codes may have difficulties enforcing lower bounds on the internal energy like $p_\infty(\tau - b) < e(\mathbf{u}) - q$, see (1.5.1). We explore this test case under our limited high order framework in section 6.3.3.*

6.2.8 Radial Sod Problem

The radial Sod problem is a cylindrical adaptation of the classic 1D Sod shocktube setup. It involves a central high-pressure region expanding outward into a surrounding lower-pressure area. Initial conditions are radially symmetric and defined by

$$\mathbf{u}(\mathbf{x}, t) = \begin{cases} (1, 0, 0, 2.5) & \text{if } \|\mathbf{x}\| \leq 0.5, \\ (8, 0, 0, 2) & \text{if } \|\mathbf{x}\| > 0.5. \end{cases}$$

The simulations are done only in the upper right quadrant as the solution is axisymmetric. The computational domain is $D^0 = (0, 1) \times (0, 1)$ with symmetry boundary conditions enforced on

the intervals $\{(0, y) \mid 0 < y < 1\}$ and $\{(x, 0) \mid 0 < x < 1\}$. An ideal-gas equation of state with adiabatic index $\gamma = 1.4$ is considered. Calculations are performed on a sequence of uniform Cartesian meshes generated from $N \times N$ square grids, with $N \in \{60, 120, 240, 480\}$. The final simulation time is $t = 0.25$ with $\text{CFL} = 1$. The results are shown in Figure 6.9. The 3D continuous rendering of density shows the overall flow structure on the finest mesh. A comparison with the 1D reference solution from [30] is also shown. Scatter plots comparing the numerical solution with the reference solution demonstrate that the essential characteristics of the flow are well preserved.

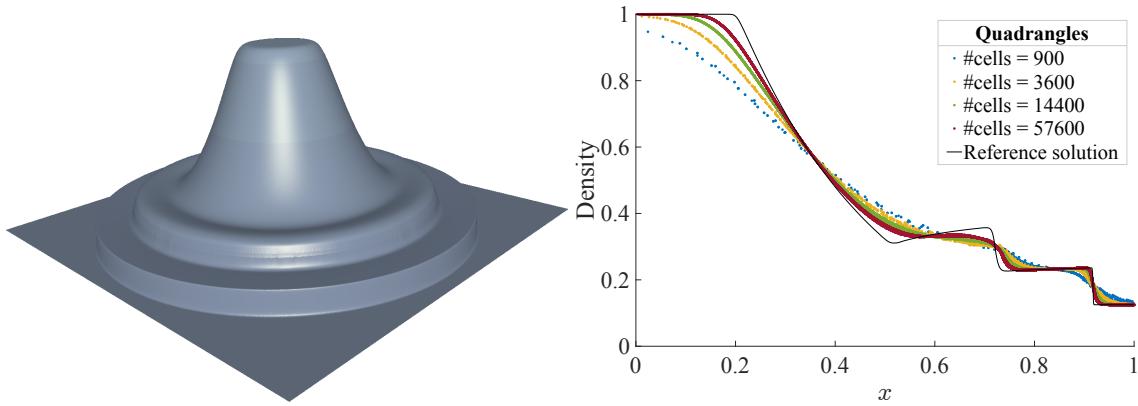


Figure 6.9: Radial Sod problem in 2D at $t = 0.25$ on a mesh composed of 57600 cells. Left: 3D rendering of the density field. Right: Scatter plot of density and 1D reference solution from [30]. Reprinted with permission from [45].

6.2.9 Noh Problem

The Noh problem is a classic test case with an exact solution, involving a shock wave that propagates radially outward at constant speed. The exact solution is given by

$$\mathbf{u}(\mathbf{x}, t) = \begin{cases} (1/16, \mathbf{0}, 1/2)^T & \text{if } \|\mathbf{x}\| \leq \frac{t}{3}, \\ ((1 + \frac{t}{\|\mathbf{x}\|})^{-1}, -\frac{\mathbf{x}}{\|\mathbf{x}\|}, \frac{1}{2})^T & \text{if } \|\mathbf{x}\| > \frac{t}{3}. \end{cases}$$

This test case is often used to assess the robustness of algorithms in handling strong discontinuities.

The initial computational domain is $D^0 = (-1, 1)^2$. We use an ideal gas equation of state with the adiabatic index $\gamma = \frac{5}{3}$. We do the computations up to $t = 0.6$ with $\text{CFL} = 0.5$. We use a sequence of uniform meshes generated from grids of size $N \times N$, where $N \in \{30, 60, 120, 240, 480\}$. Convergence rates and $\delta^1(0.6)$ relative errors are presented in Table 6.6.

Table 6.6: Noh problem. Error indicator $\delta^1(0.6)$ and convergence order obtained on different meshes. Reprinted with permission from [45].

# cells	$\delta^1(0.6)$	order
3600	1.277E-1	—
14400	5.837E-2	1.13
57600	2.801E-2	1.06
230400	1.370E-2	1.03

Figure 6.10 presents the simulation results on the finest meshes, showing the computed density fields and final mesh configurations at $t = 0.6$. The images demonstrate how the final mesh adapts to the shock structure.

Figure 6.11 shows scatter plots of the density distribution for the Noh problem at $t = 0.6$ for the two finest meshes $N \in \{120, 240\}$, comparing the numerical solutions (blue dots) with the exact solution (black solid line).

We finish this section solving the Noh problem on a non-uniform mesh constructed as follows. The initial square D_0 is divided into four quadrants. The bottom left quadrant uses a mesh composed of 32×32 square cells, the top left quadrant uses a mesh composed of 32×64 rectangular cells, the top right quadrant uses a mesh composed of 64×64 square cells, and the bottom right quadrant uses a mesh composed of 64×32 rectangular cells. The goal of this test is to assess the ability of the algorithm to maintain radial symmetry. We show in Figure 6.12 the density field at $t = 0.6$ and the final mesh. There is a slight loss of symmetry, but the solution does not exhibit the Rayleigh-Taylor-type instabilities that often appear in other Lagrangian codes.

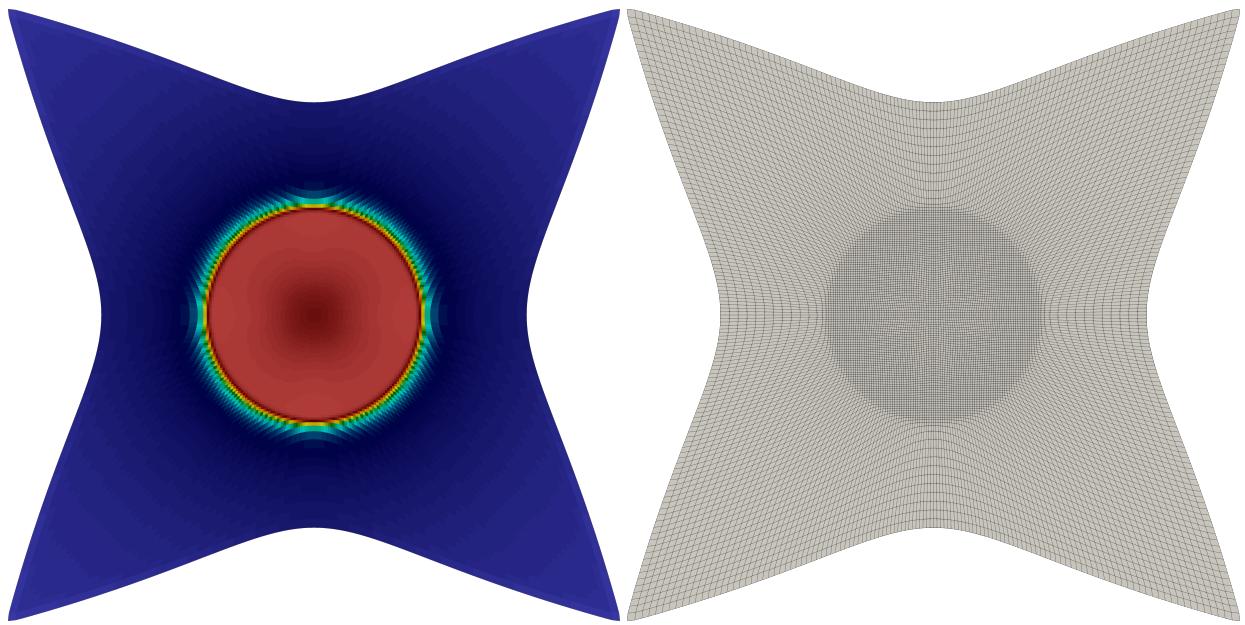


Figure 6.10: Density field and mesh from Noh problem on uniform meshes at $t = 0.6$. Reprinted with permission from [45].

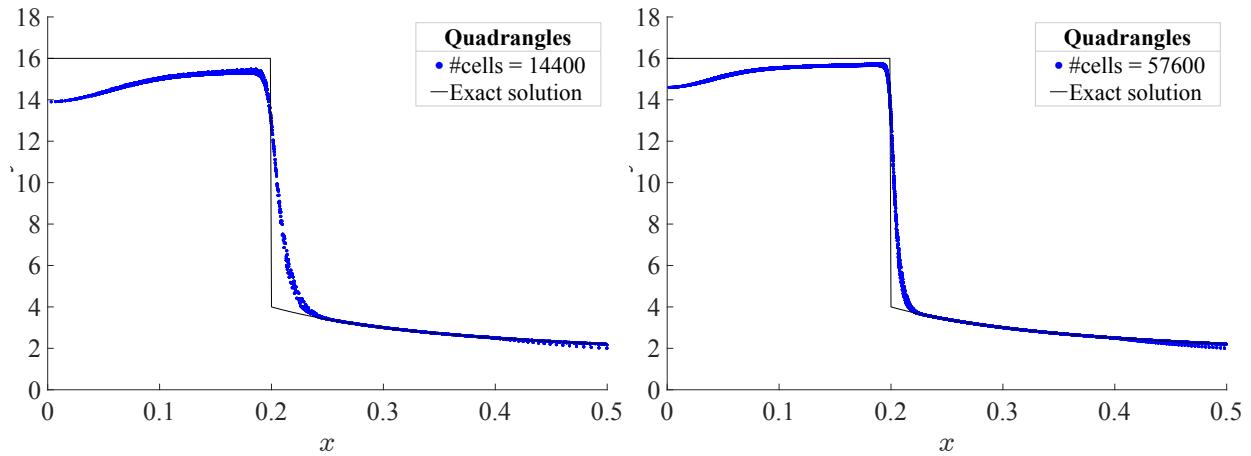


Figure 6.11: Noh problem. Scatter plots of the density at $t = 0.6$ comparing the approximation (blue dots) with the exact solution (black solid line). Left: mesh with 14400 cells. Right: mesh with 57600 cells. Reprinted with permission from [45].

6.2.10 Sedov Explosion

The Sedov explosion problem is a test used to evaluate the ability of Lagrangian codes to handle strong shocks and radial symmetry. We use the computational domain $D^0 = (0, 1)^2$ with reflective

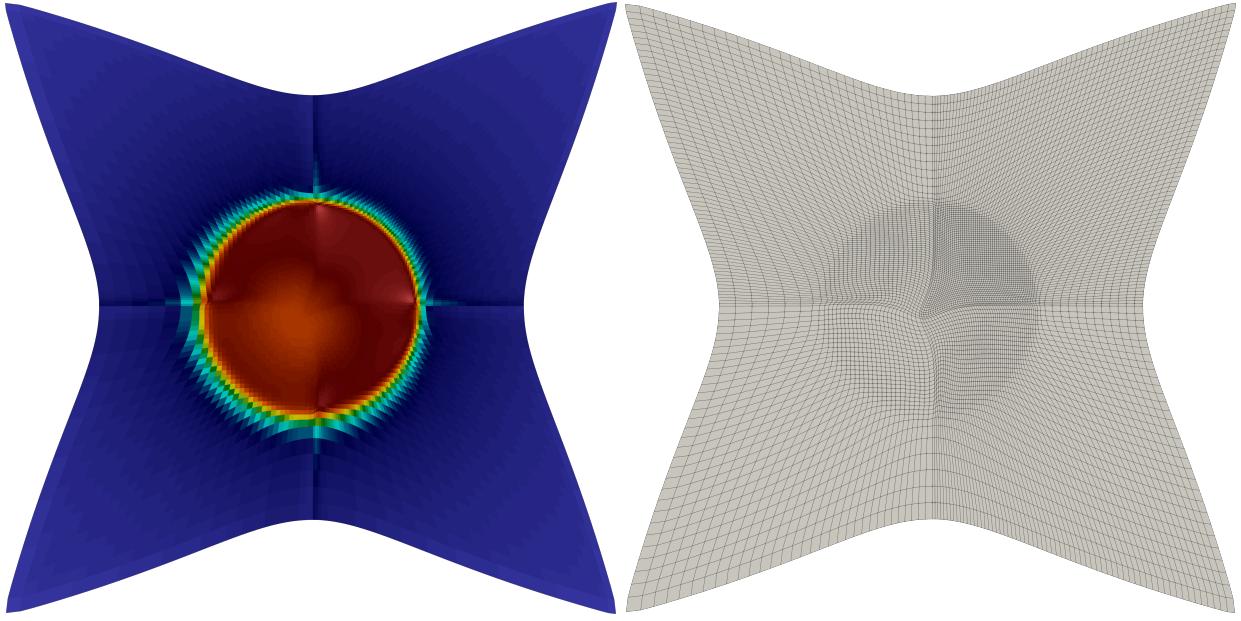


Figure 6.12: Noh problem on non-uniform mesh. Density and mesh at final time. Reprinted with permission from [45].

boundary conditions. The initial conditions are given by $(\rho_0, P_0, \mathbf{u}) = (1, 10^{-6}, 0)$, for an ideal gas with polytropic index set to $\gamma = \frac{7}{5}$. We approximate an initial Dirac-delta energy source by prescribing a quantity of internal energy ε into a small region defined by $D_{r_0} = \{\mathbf{x} \in D^0; \|\mathbf{x}\| \leq r_0\}$ with $r_0 := 0.05$. Denoting by $|D_{r_0}|$ the area of D_{r_0} , the pressure inside the disk D_{r_0} is given by $p_0 = \frac{(\gamma-1)\varepsilon}{|D_{r_0}|}$ where the energy, $\varepsilon = 0.979264$, is computed so that the radius of the shock is equal to 1 at $t = 1$. The simulations are run until $t = 0.9$ with $\text{CFL} = 1$.

Table 6.7: Modified composite L^1 -error for the Sedov problem at $t = 0.9$ using $\text{CFL} = 1$. Convergence is observed in the density, velocity, and total energy. Reprinted with permission from [45].

# cells	$\delta^1(0.9)$	order
3600	1.370E-1	–
14400	9.089E-2	0.59
57600	6.222E-2	0.55
230400	3.518E-2	0.82

Table 6.7 shows the convergence rates for the modified $\delta^1(0.9)$ error indicator, where the spe-

cific volume has been replaced by the density and the specific total energy has been replaced by the total energy to accommodate for the Dirac-delta initial condition. The calculations are performed on a sequence of uniform meshes generated from $N \times N$ square grids, $N \in \{60, 120, 240, 480\}$. Although the wave is radial, these simulations have been done on purpose with initial grids that are Cartesian to demonstrate the robustness of the proposed method with respect to the radial symmetry.

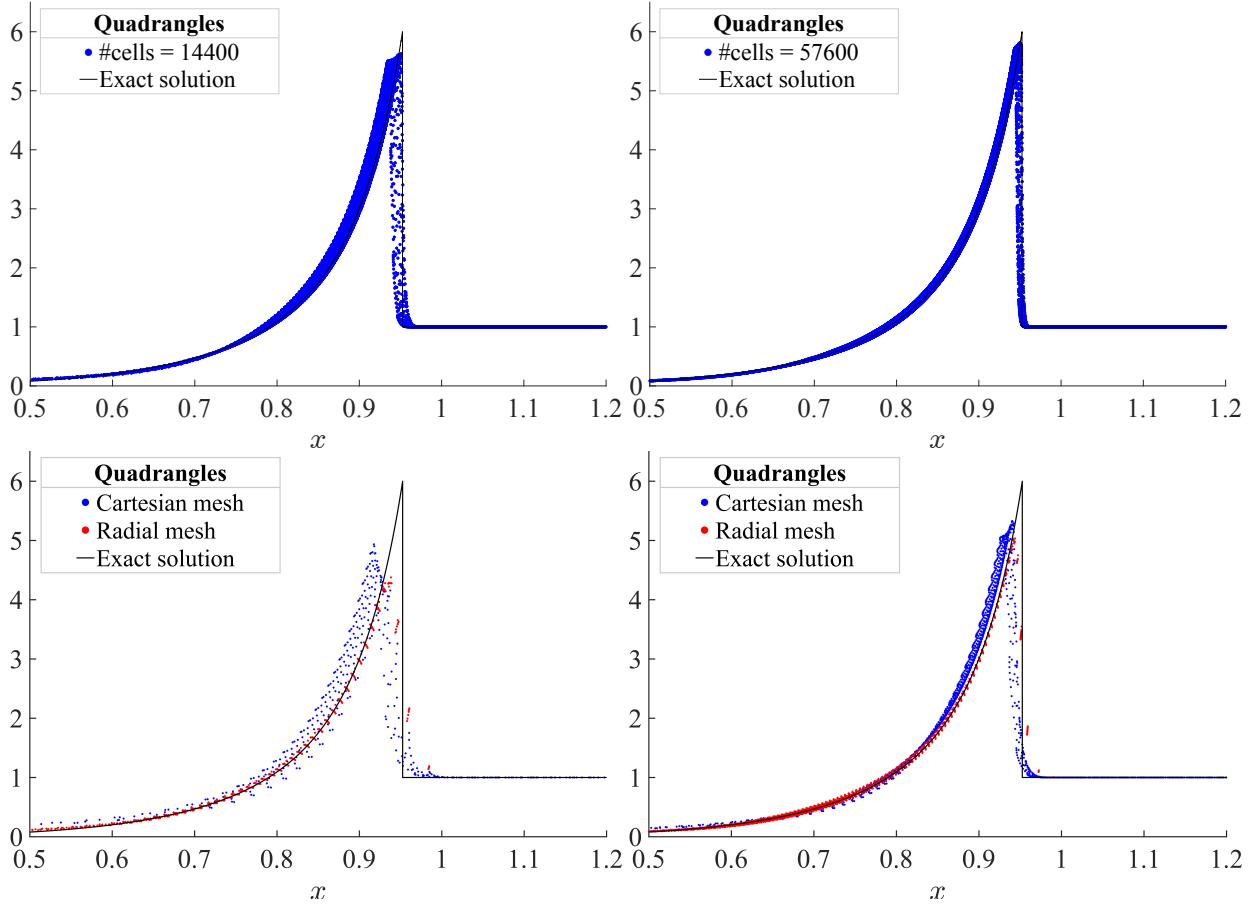


Figure 6.13: Density scatter plots at $t = 0.9$. Top row: Cartesian meshes. From left to right: quadrangular meshes with 14400 cells and 57600 cells. Bottom row: Results comparing cartesian vs. radial coarse meshes. Left: Cartesian mesh (900 cells) vs. radial mesh (896 cells). Right: Cartesian mesh (3600 cells) vs. radial mesh (3584 cells). Reprinted with permission from [45].

Figure 6.13 displays scatter plots of the density distribution. The numerical solutions (blue

dots) are compared with the analytical solution (black solid line). The results shown in the top row have been obtained with initial grids that are Cartesian. In the second row, we compare results obtained using Cartesian grids with results obtained using radial grids (red dots) with approximately the same number of cells. We use coarse meshes on purpose to emphasize the differences. As expected, the radial grids are slightly more accurate, but the point of this series of tests is to demonstrate the robustness of the method when using initial grids that do not have the symmetry of the solution. Convergence is observed for both Cartesian and radial meshes.

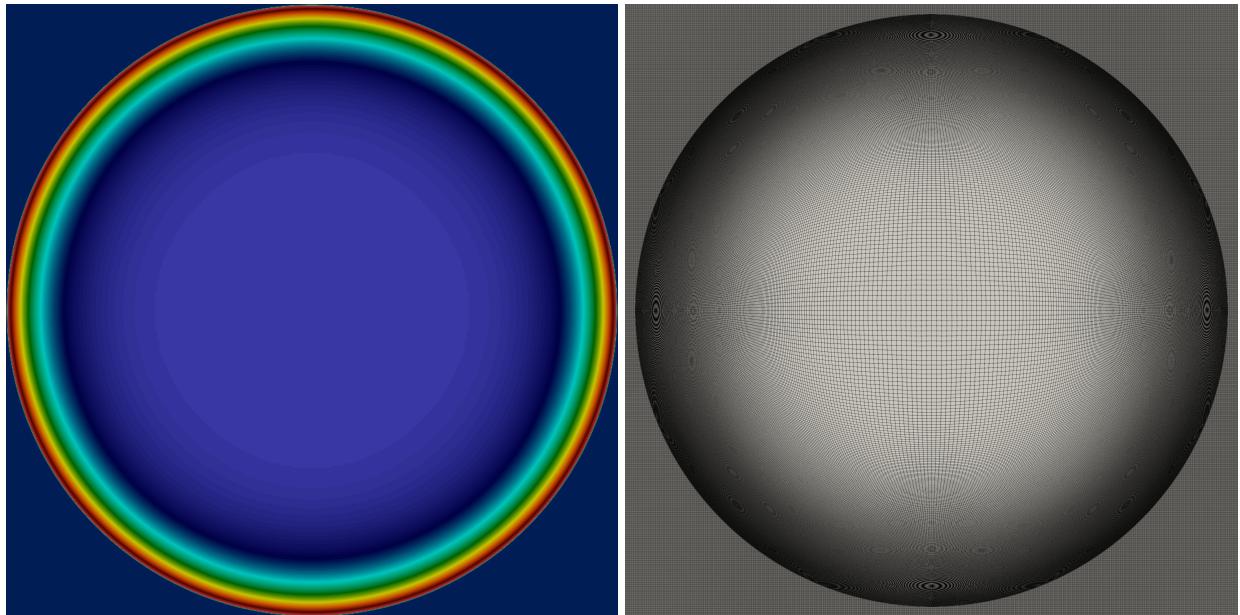


Figure 6.14: Sedov problem. Density and mesh composed of 230400 cells at $t = 0.9$. Reprinted with permission from [45].

Finally, Figure 6.14 presents the density field and mesh obtained on the finest mesh. The initial grids are on purpose chosen to be Cartesian to demonstrate robustness. We observe that the radial symmetry of the problem is well maintained in both cases, and the mesh adapts to resolve the region where the shock wave is located.

6.2.11 Multimaterial Triple Point

We finish the low order numerical results section with simulations of the triple point problem. This problem is set in the rectangular domain $D^0 = [0, 7] \times [0, 3]$. The domain D^0 is divided into three distinct regions: $D_1^0 = [0, 1] \times [0, 3]$, $D_2^0 = [1, 7] \times [0, 1.5]$, and $D_3^0 = [1, 7] \times [1.5, 3]$ as depicted in Figure 6.15. The initial condition for the is a follows:

$$\mathbf{u}(\mathbf{x}, 0) = \begin{cases} (1, \mathbf{0}, 2) & \text{if } \mathbf{x} \in D_1, \\ (1, \mathbf{0}, 0.25) & \text{if } \mathbf{x} \in D_2, \\ (8, \mathbf{0}, 1.6) & \text{if } \mathbf{x} \in D_3. \end{cases} \quad (6.2.8)$$

The regions D_1 and D_3 are filled with the same gas characterized by $\gamma = 1.5$, whereas the region D_2 is filled with a different material with $\gamma = 1.4$. The target final time is $t = 5$.

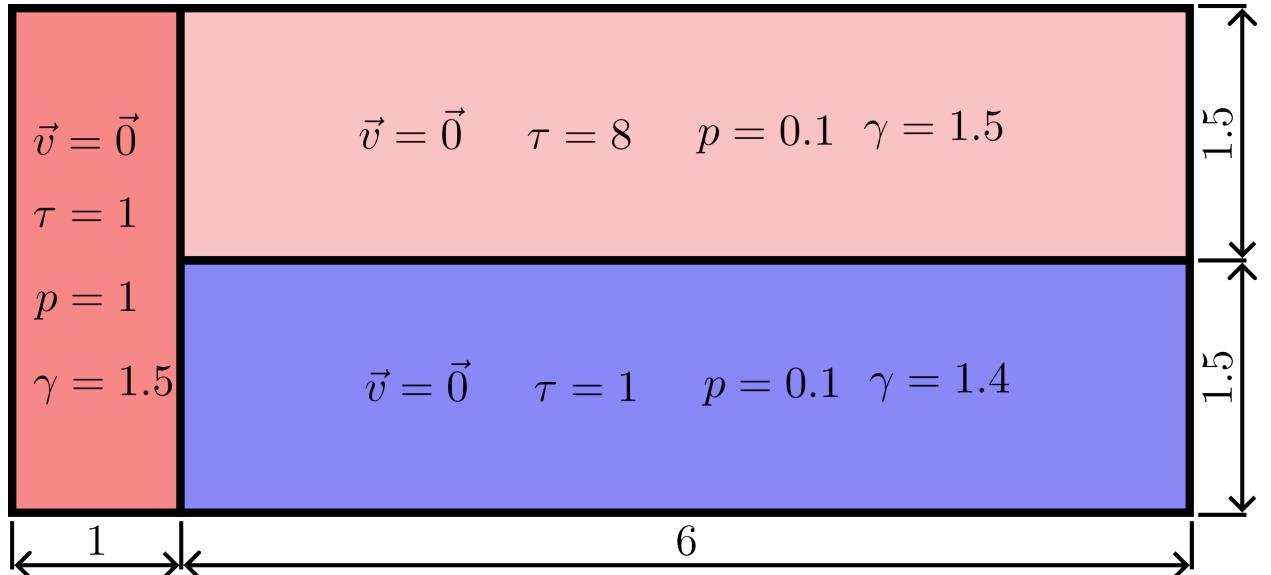


Figure 6.15: Initial configuration for the triple point problem.

It is known that most Lagrangian algorithms fail solving this problem until the final time $t = 5$. For most algorithms, the vortex induces premature mesh tangling . As stated in [61], “ $t \approx 1.67$ is

the time after which any Lagrangian scheme inexorably fails” due to the mesh tangling. Whichever mass conservation strategy is used, the approximation method using quadrilateral meshes cannot reach the final time $t = 5$ due to mesh tangling. We believe this is due to the rigidity of the low order mesh. As seen in figure 6.16, we observe that the mesh is resistant to rotation and that curvilinear mesh boundaries are essential. Our results applying the relaxed limiting framework described in chapter 4 can be found in section 6.3.6.

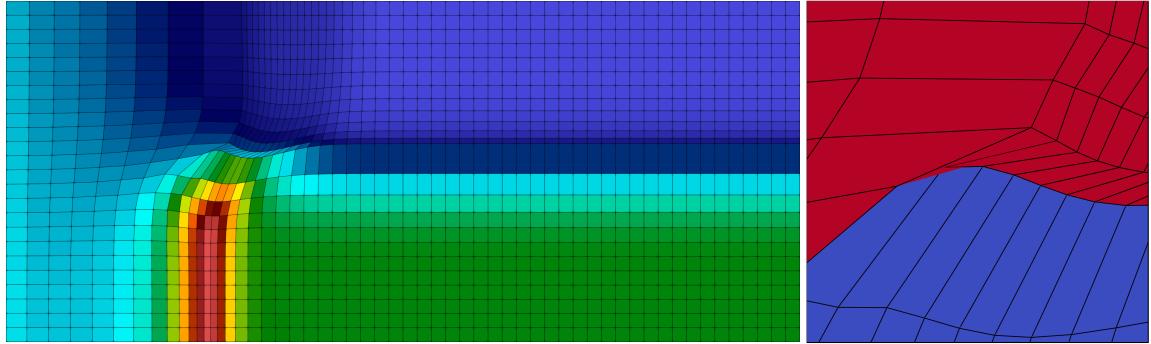


Figure 6.16: Left: Density field for the triple-point problem at time of collapse $t = 1.101$ on a coarse mesh composed of 1,344 cells. Right: zoom of the mesh at the material boundaries at the time of collapse.

6.3 High Order Results

The purpose of this section is to validate our limiting procedure described in Chapter 4. As previously mentioned, the high order method that we limit is Laghos, developed at Lawrence Livermore National Laboratory [28]. The primary goal is to verify that the application of the limiter does not degrade the expected high-order convergence rates when we look at smooth test problems.

For all the tests presented here, we utilize the RK4 timestepping scheme given by the Butcher Tableau in (4.2.1). The time step at each time instant t^n is defined to be the minimum from the automatic time step control estimate outlined in [28, § 7.3] and the timestep given according to the CFL restriction on the low order method, given in (6.1.1). When an exact solution is available, we

report both the error and the observed order of convergence. Results are presented using a three-block column structure: columns 1–2 correspond to the unmodified high-order method, columns 3–4 to the limiting algorithm without relaxation of the bounds, and columns 5–6 to the full limiting procedure with relaxation.

6.3.1 Approximation Space Notation

In our computations, we follow the approximation space choices adopted in [28, § 4.7]. While their general framework permits independent selection of kinematic and thermodynamic spaces, the authors note that stability and accuracy considerations impose a interdependence relationship between the approximation spaces. Specifically, we employ the so-called Q_{k+1} – Q_k pairing: the velocity space

$$V = (Q_{k+1})^d,$$

consisting of continuous finite elements of order $k + 1$ on quadrilateral meshes, and the thermodynamic space

$$E = Q_k,$$

consisting of discontinuous finite elements one degree lower. At lowest order ($k = 0$) one recovers the classical Q_1 – Q_0 discretization corresponding to straight-edged zones with piecewise constant thermodynamic variables which is similar to the approximation space we employ in our low order approximation described in Chapter 3. The key difference of course is that in the low order approximation outlined in Chapter 3, the thermodynamic velocity solved for in the Euler equations is represented using the Q_0 approximation space and only the mesh velocity is approximated in the continuous Q_1 space. Furthermore, we abbreviate this notation by referring to the Q_{k+1} – Q_k pairing as the Q_k approximation, where the velocity space being one order greater is implied.

6.3.2 Smooth 1D

We begin with a smooth wave case in 1D introduced in [44, Sec. 5.2]. The purpose of this test is to estimate the convergence rate of the method on a very smooth solution. We consider the

following exact solution to the Euler equations: $v(x, t) = 1$, $p(x, t) = 1$, and

$$\rho(x, t) = \begin{cases} 1 + \frac{64}{(x_1 - x_0)^6} (x - t - x_0)^3 (x_1 - x + t)^3, & \text{if } x_0 \leq x - t < x_1, \\ 1, & \text{otherwise.} \end{cases} \quad (6.3.1)$$

with $x_0 = 0.1$, $x_1 = 0.3$, and $\gamma = \frac{7}{5}$. The computational domain is $D = [0, 1]$ and error and convergence rates are evaluated at final time $t_f = 0.6$. We report our results using a Q_1 approximation in Table 6.8, using a Q_2 approximation in Table 6.9, and using a Q_3 approximation in Table 6.10. We observe that both the Q_1 and Q_2 approximations achieve the expected theoretical order of convergence, and that the application of the limiting procedures—with or without relaxation—does not degrade the convergence rate. In contrast, we do observe a reduction in the observed order of convergence for the Q_3 approximation.

Table 6.8: Convergence results for the smooth 1D test case with Q_1 approximation, demonstrating the expected order accuracy for all variants of the method.

# dofs	Q_1		Q_1 Limited		Q_1 Relaxed	
	$\delta^1(0.6)$	order	$\delta^1(0.6)$	order	$\delta^1(0.6)$	order
120	7.501E-4	—	9.195E-4	—	9.196E-4	—
240	1.639E-4	2.19	1.834E-4	2.33	1.724E-4	2.42
480	4.069E-5	2.01	4.408E-5	2.06	4.156E-5	2.05
960	1.015E-5	2.00	1.064E-5	2.05	1.022E-5	2.02
1920	2.531E-6	2.00	2.600E-6	2.03	2.536E-6	2.01
3840	6.327E-7	2.00	6.421E-7	2.02	6.331E-7	2.00

6.3.3 Van der Waals Equation of State

We now turn our attention to the one-dimensional van der Waals (VDW) test case, introduced in Section 6.2.7. In our robust low-order implementation, this problem runs stably with $CFL = 1$. By contrast, the unlimited high-order Laghos code fails immediately: the density leaves the admissible set defined by the VDW maximum compressibility condition, and the computation crashes within

Table 6.9: Convergence results for the smooth 1D test case with Q_2 approximation, demonstrating the expected order accuracy for all variants of the method.

# dofs	Q_2		Q_2 Limited		Q_2 Relaxed	
	$\delta^1(0.6)$	order	$\delta^1(0.6)$	order	$\delta^1(0.6)$	order
180	9.433E-5	—	2.806E-4	—	1.808E-4	—
360	1.507E-5	2.65	4.092E-5	2.78	2.410E-5	2.91
720	2.048E-6	2.88	4.432E-6	3.21	2.848E-6	3.08
1440	2.554E-7	3.00	5.219E-7	3.09	3.288E-7	3.11
2880	3.225E-8	2.99	6.330E-8	3.04	3.902E-8	3.07
5760	4.030E-9	3.00	7.711E-9	3.04	4.687E-9	3.06

Table 6.10: Convergence results for the smooth 1D test case with Q_3 approximation.

# dofs	Q_3		Q_3 Limited		Q_3 Relaxed	
	$\delta^1(0.6)$	order	$\delta^1(0.6)$	order	$\delta^1(0.6)$	order
240	2.464E-5	—	1.884E-4	—	3.096E-4	—
480	1.988E-6	3.63	1.065E-5	4.15	2.787E-5	3.47
960	1.162E-7	4.10	8.098E-7	3.72	2.880E-6	3.27
1920	7.448E-9	3.96	1.012E-7	3.00	3.280E-7	3.13
3840	4.649E-10	4.00	1.217E-8	3.06	3.876E-8	3.08
7680	3.103E-11	3.90	1.505E-9	3.02	4.804E-9	3.01

a single time step. Reducing the CFL to 0.5 allows the simulation to progress only to $t_f = 0.0001$ before breakdown, while further reduction to $\text{CFL} = 0.05$ results in an immediate crash of the timestep. These results confirm our intuition in Remark 6.2.1: without additional stabilization, traditional high-order Lagrangian methods are not robust for this class of problems.

To address this, we evaluate the problem with our limiting procedure applied to the density component. In this case, the limiter corrects invalid values in the density, enabling the simulation to advance further in time before encountering invalid values in the specific internal energy. Figure 6.17 shows density profiles at the last attainable time for several mesh resolutions. The unlimited high-order solution (blue) exhibits severe overshoots and violates the maximum compressibility condition, while the limited density (red) remains within the physically admissible range established by the low-order solution. These results illustrate that the density limiting procedure is

essential for extending the computational stability to high-order methods.

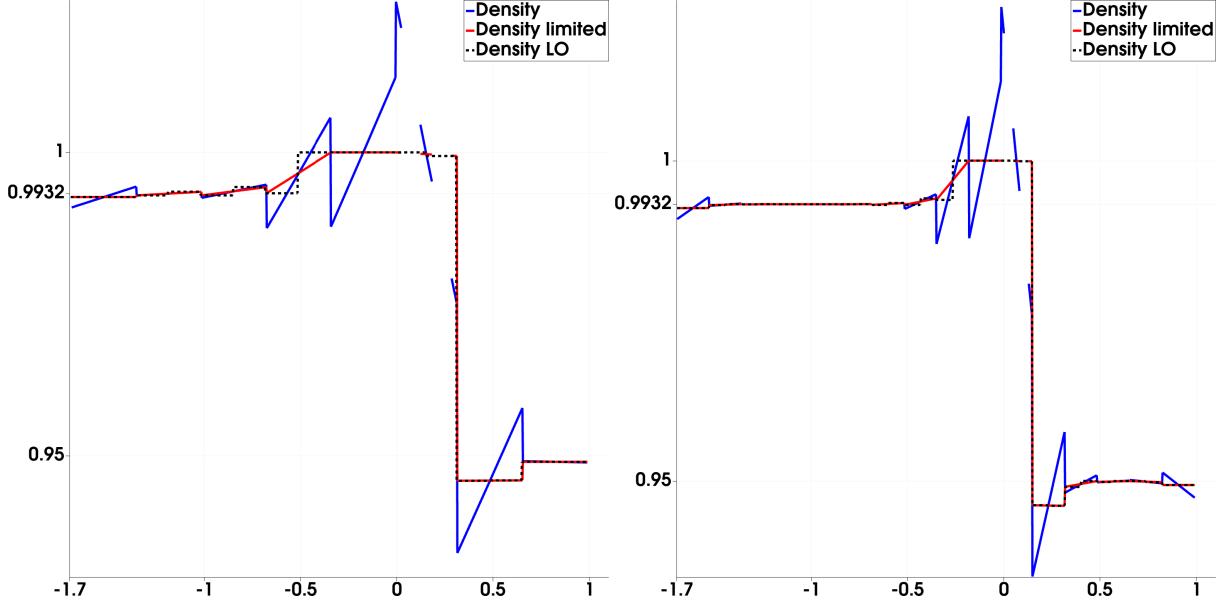


Figure 6.17: Density profiles for the van der Waals test (6.2.7) with a CFL number 1 using a Q_2-Q_1 approximation space. The high-order Laghos solution with a Q_1 approximation (blue) becomes unstable and crashes within a single timestep in the absence of limiting. In contrast, the low-order invariant domain preserving method (black dashed) remains robust, and the limited solution (red) shows strict adherence to the local bounds. Left: Mesh composed of 8 cells run until $t_f = 0.0031$. Right: Mesh composed of 16 cells run until $t_f = 0.0017$.

6.3.4 Isentropic Vortex

We now turn to smooth two-dimensional problems to assess the performance of the limiting procedure. We begin with the isentropic vortex test introduced in Section 6.2.3. All runs use $CFL = 0.1$ up to the final time $t_f = 0.25$. Convergence results with a Q_1 approximation are reported in Table 6.11, and with a Q_2 approximation in Table 6.12. For Q_2 , we observe a mild order reduction on the finest refinements, which we attribute to the temporal error dominating the error due to the spatial discretization. In Figure 6.18 we plot the relative amount of limited density with respect to time in both the Q_1 and Q_2 approximation spaces. We observe that both the maximum and average values across all DoFs does not grow in time but remains constant. This

indicates that our limiting procedure is not introducing any spurious oscillations in the high order solver in time.

Table 6.11: Convergence results for the isentropic vortex test case with Q_1 approximation, demonstrating the expected order accuracy for all variants of the method.

# dofs	Q_1		Q_1 Limited		Q_1 Relaxed	
	$\delta^1(0.25)$	order	$\delta^1(0.25)$	order	$\delta^1(0.25)$	order
1024	3.343E-3	—	7.866E-3	—	4.422E-3	—
4096	5.712E-4	2.55	1.040E-3	2.92	6.249E-4	2.82
16384	1.244E-4	2.20	1.958E-4	2.41	1.369E-4	2.19
65536	3.049E-5	2.03	3.926E-5	2.32	3.316E-5	2.05
262144	8.067E-6	1.92	8.622E-6	2.19	8.622E-6	1.94

Table 6.12: Convergence results for the isentropic vortex test case with Q_2 approximation, demonstrating the expected order accuracy for all variants of the method until the finer refinements where we believe the temporal error dominates the spatial error due to the CFL condition.

# dofs	Q_2		Q_2 Limited		Q_2 Relaxed (1)	
	$\delta^1(0.25)$	order	$\delta^1(0.25)$	order	$\delta^1(0.25)$	order
144	3.474E-2	—	5.751E-2	—	5.487E-2	—
576	4.585E-3	2.92	1.661E-2	1.79	1.359E-2	2.01
2304	2.984E-4	3.94	1.170E-3	3.83	5.512E-4	4.62
9216	2.429E-5	3.62	1.258E-4	3.22	3.578E-5	3.95
36864	2.973E-6	3.03	2.032E-5	2.63	6.269E-6	2.51

6.3.5 Taylor-Green Vortex

We now return to the Taylor–Green vortex problem, first introduced in the context of the low-order method in Section 6.2.4. This smooth, shock-free problem provides an effective benchmark for verifying that our high-order discretization achieves the expected rates of convergence on a moving mesh with nontrivial deformation. We run on the same domain as in the low order case with the same boundary conditions until final time $t_f = 0.1$ with $\text{CFL} = 0.1$. We present convergence

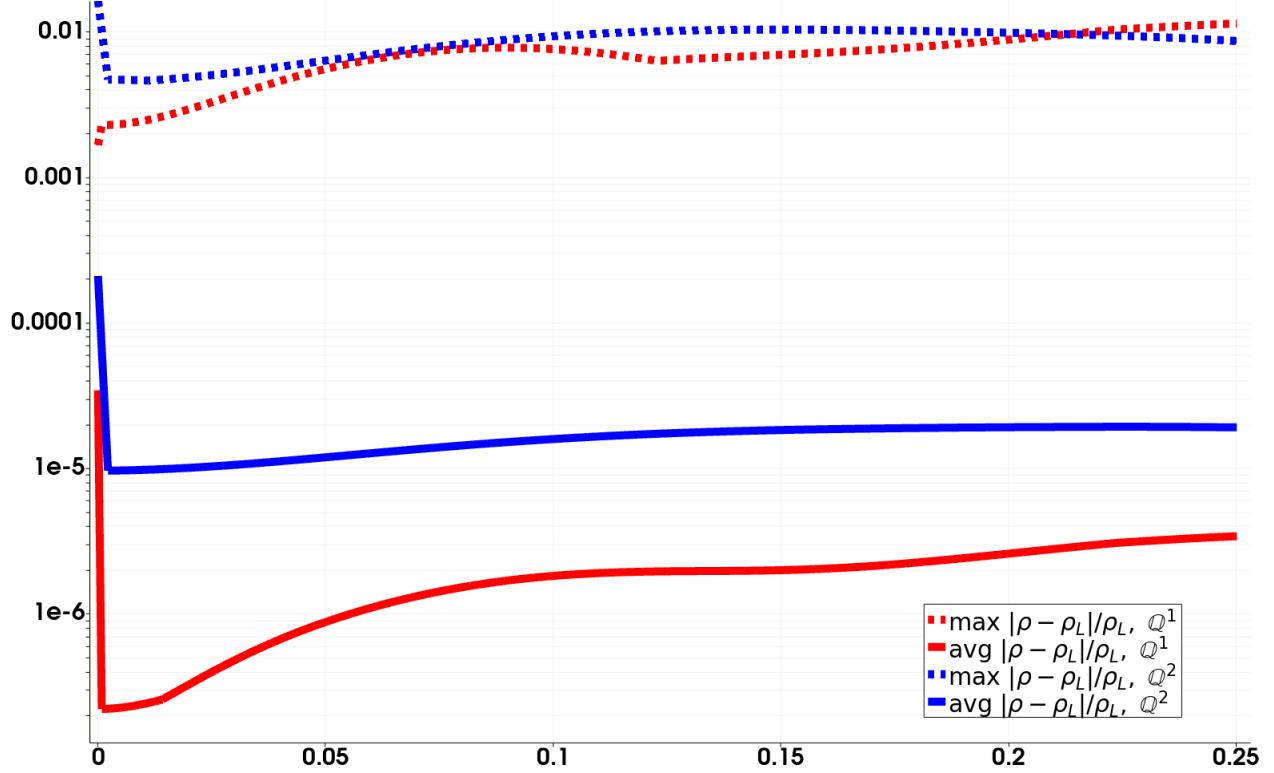


Figure 6.18: Time evolution of the relative amount of limited density constrained by the limiter for the isentropic vortex problem. Results are shown for both Q_1 and Q_2 approximations, with solid lines denoting average quantities across DoFs and dashed lines denoting maximum quantities.

results obtained with a Q_1 approximation in Table 6.13 and with a Q_2 approximation in Table 6.14. We again plot the relative amount of density that is limited with respect to time in the case of the Taylor-Green vortex in Figure 6.19. This further reinforces the point that for smooth 2D problems our limiting framework does not introduce additional dissipation or oscillations in the density: the fraction $|\rho - \rho_L|/\rho_L$ stays bounded and approximately constant in time, and the observed convergence rates are preserved.

6.3.6 Multimaterial Triple Point

We repeat the triple point problem outlined in section 6.2.11 in the limited Laghos framework. In contrast to the first-order scheme, the higher-order Q_k approximations in Laghos do not exhibit mesh tangling; accordingly, we do not expect entanglement when the limiter is applied. The tangling observed with the first-order method (see Fig. 6.16) is irrelevant to the results reported

Table 6.13: Convergence results for the Taylor-Green vortex test case with Q_1 approximation, demonstrating the expected order accuracy for all variants of the method.

# dofs	Q_1		Q_1 Limited		Q_1 Relaxed	
	$\delta^1(0.1)$	order	$\delta^1(0.1)$	order	$\delta^1(0.1)$	order
16	3.275E-2	—	3.259E-2	—	3.259E-2	—
64	1.090E-2	1.59	1.120E-2	1.54	1.120E-2	1.54
256	2.416E-3	2.17	2.426E-3	2.21	2.431E-3	2.20
1024	5.878E-4	2.04	6.021E-4	2.01	6.032E-4	2.01
4096	1.456E-4	2.01	1.519E-4	1.99	1.520E-4	1.99
16384	3.618E-5	2.01	3.880E-5	1.97	3.881E-5	1.97

Table 6.14: Convergence results for the Taylor-Green vortex test case with Q_2 approximation, demonstrating the expected order accuracy for all variants of the method until the finer refinements where we believe again the temporal error dominates the spatial error due to the CFL condition.

# dofs	Q_2		Q_2 Limited		Q_2 Relaxed	
	$\delta^1(0.1)$	order	$\delta^1(0.1)$	order	$\delta^1(0.1)$	order
36	1.538E-2	—	1.524E-2	—	1.533E-2	—
144	1.787E-3	3.11	1.839E-3	3.05	1.824E-3	3.07
576	1.787E-3	3.11	1.905E-4	3.27	1.910E-4	3.26
2304	2.126E-5	3.07	2.801E-5	2.77	2.814E-5	2.76
9216	2.582E-6	3.04	7.179E-6	1.96	7.195E-6	1.97

here, since the low-order mesh velocity update is not used in the high-order runs. Our objective is to quantify the activity of the density limiter and verify that only minimal limiting is required to keep the high-order density within the admissible bounds defined by the low-order state. We observe that at final time we are limiting at most 20% of the density

6.4 Nonlinear Hyperelasticity Results

Unless otherwise stated, in the following test problems the material is considered elastic with a shear modulus $\mu = 2.6 \times 10^{10}$ Pa. To compute specific internal energy we use the Nobel-Abel Stiffened-Gas equation of state

$$e = \frac{p + \gamma p_\infty}{(\gamma - 1) \rho} \quad (6.4.1)$$

with $p_\infty = 2.15 \times 10^{10}$, where $b = 0$ and $q = 0$.

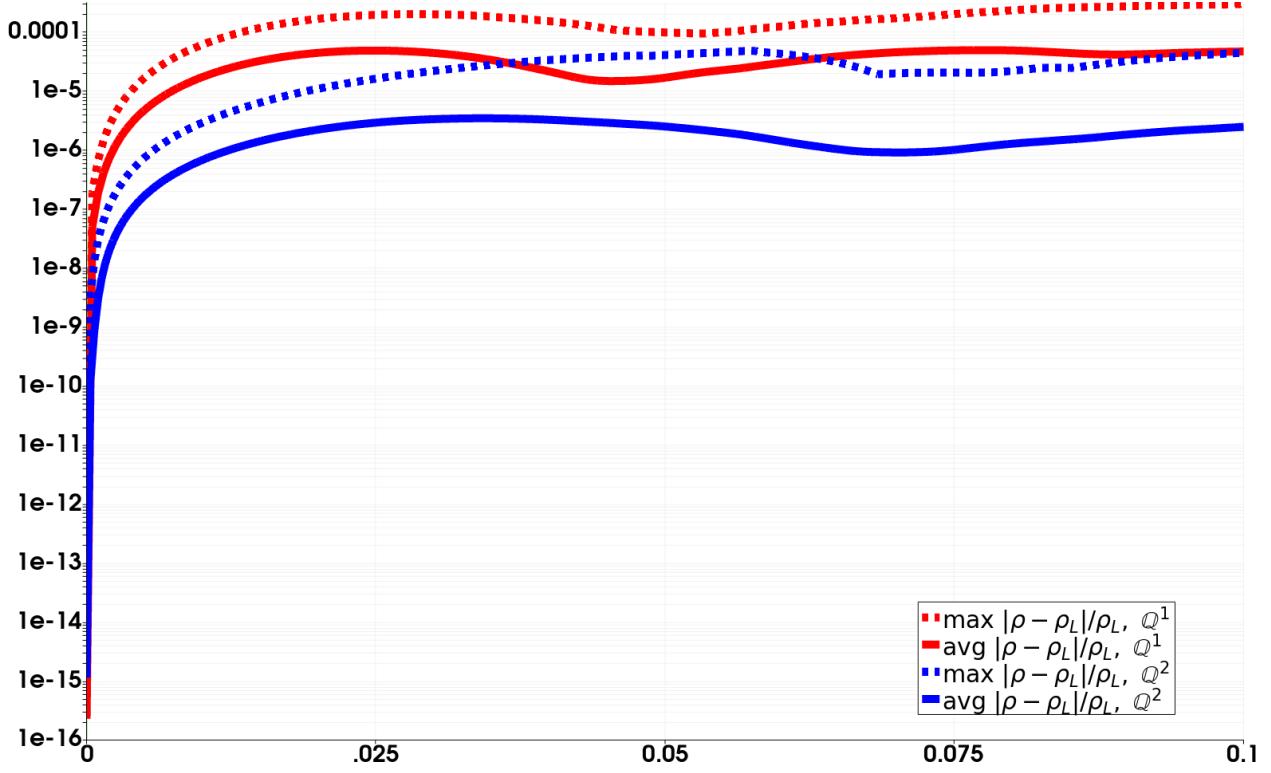


Figure 6.19: Time evolution of the relative amount of limited density constrained by the limiter for the Taylor-Green vortex problem. Results are shown for both Q_1 and Q_2 approximations, with solid lines denoting average quantities across DoFs and dashed lines denoting maximum quantities.

6.4.1 Shocktube

In this section we start by evaluating the method on a simple 1D shocktube. We take as our domain $D^0 = [0, 1]$. The initial conditions are set as

$$\mathbf{u}_0(\mathbf{x}) = (\rho, v, p)^\top = \begin{cases} (2.7 \times 10^3, 0, 1 \times 10^7)^\top & \text{if } 0 < x_1 \leq 0.5, \\ (2.7 \times 10^3, 0, 1 \times 10^5)^\top & \text{if } 0.5 < x_1 < 1. \end{cases} \quad (6.4.2)$$

We do not allow our mesh boundary to move by imposing $v \cdot n = 0$ on our mesh velocity. We perform five simulations: one on the initial mesh consisting of 256 segments, and four additional simulations on subdivided refined meshes. The simulations are run up to $t_f = 0.00005$ with CFL = 0.25. Table 6.15 shows convergence to the reference solution. The reference solution is computed

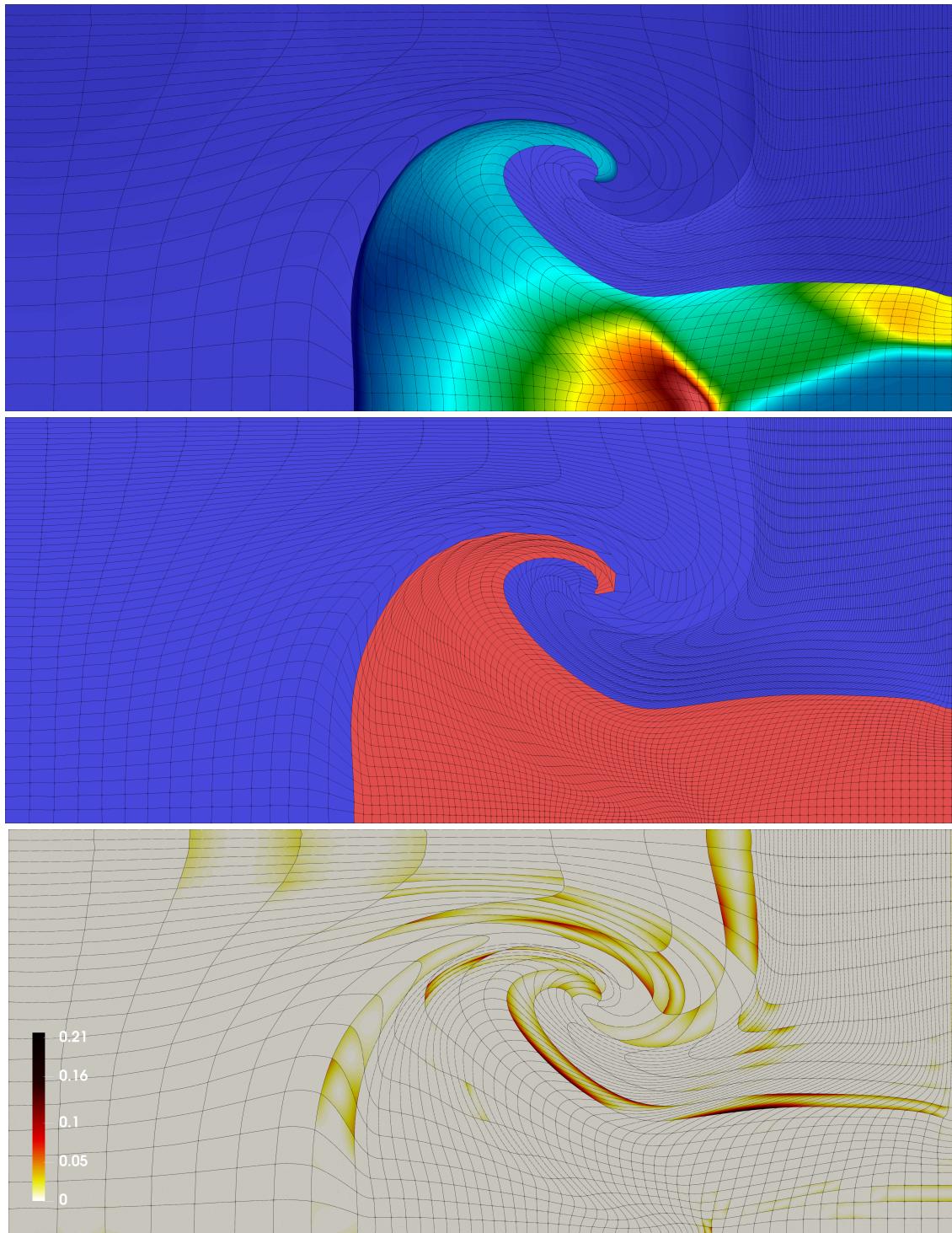


Figure 6.20: Triple-point problem at final time $t_f = 5$. Top: limited high-order density field (pair Q_2-Q_1). Middle: low-order material indicator on the submesh. Bottom: limiter activity for density—the relative correction applied at the Q_1 DoFs.

using the Eulerian hyperelasticity code developed in Ndanou et al. (2015) on a mesh composed of 600 cells. In Figure 6.21 we observe the expected order of convergence to the reference solution.

Table 6.15: Convergence test for the Riemann problem (6.4.2). Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.

# cells	$\delta^1(5 \times 10^{-5})$	order	$\delta_\sigma^1(5 \times 10^{-5})$	order
32	2.631E-1	—	9.836E-2	—
64	1.672E-1	0.65	6.145E-2	0.68
128	1.060E-1	0.66	3.818E-2	0.69
256	7.205E-2	0.56	2.564E-2	0.57
512	4.592E-2	0.65	1.592E-2	0.69
1024	2.763E-2	0.73	9.116E-3	0.80
2048	1.570E-2	0.81	4.764E-3	0.94

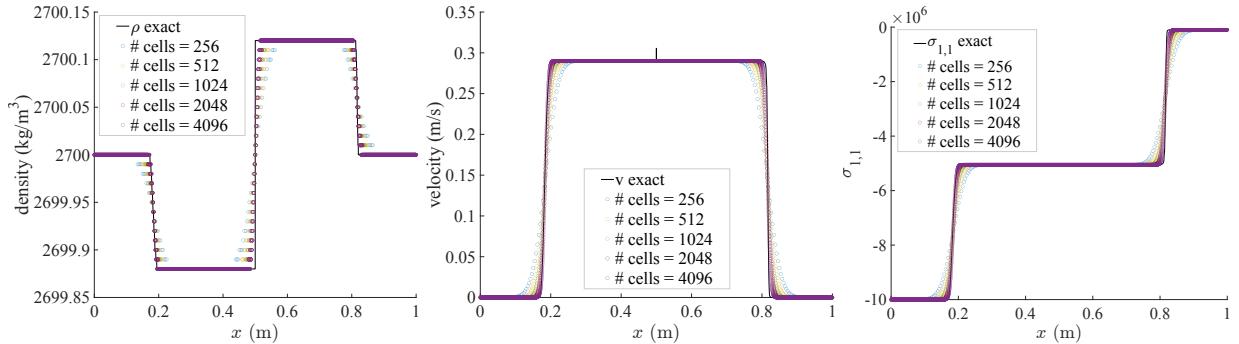


Figure 6.21: Solution to the Riemann problem (6.4.2) at $t = 0.00005$ with $CFL = .5$. From left to right: density, velocity, stress.

6.4.2 Double Impact Problem

The Riemann problem with two colliding materials is considered with an impact velocity difference of 200 m/s. We again take as our domain $D^0 = [0, 1]$. The materials have the same initial pressure $p = 10^5$ Pa and the same initial density $\rho = 1000$ kg / m³, with zero initial shear stress ($S = 0$).

The initial mesh for the method is composed of 256 segments. Five simulations are done: one on the initial mesh and four on meshes obtained by uniform refinement. We impose stationary mesh velocity boundary conditions by enforcing $v \cdot n = 0$ at the endpoints $x = 0$ and $x = 1$ of our mesh. We run the simulation until $t_f = 0.00005$ to avoid interaction of the shock waves with the boundary of our simulation, and set the CFL number to 0.25.

Table 6.16: Convergence test for the double impact problem. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.

# cells	$\delta^1(5 \times 10^{-5})$	order	$\delta_\sigma^1(5 \times 10^{-5})$	order
512	1.408E-1	—	5.057E-2	—
1024	9.353E-2	0.59	3.379E-2	0.58
2048	6.103E-2	0.62	2.223E-2	0.60
4096	3.791E-2	0.69	1.402E-2	0.66
8192	2.183E-2	0.80	8.318E-3	0.75
16384	1.091E-2	1.00	4.439E-3	0.91

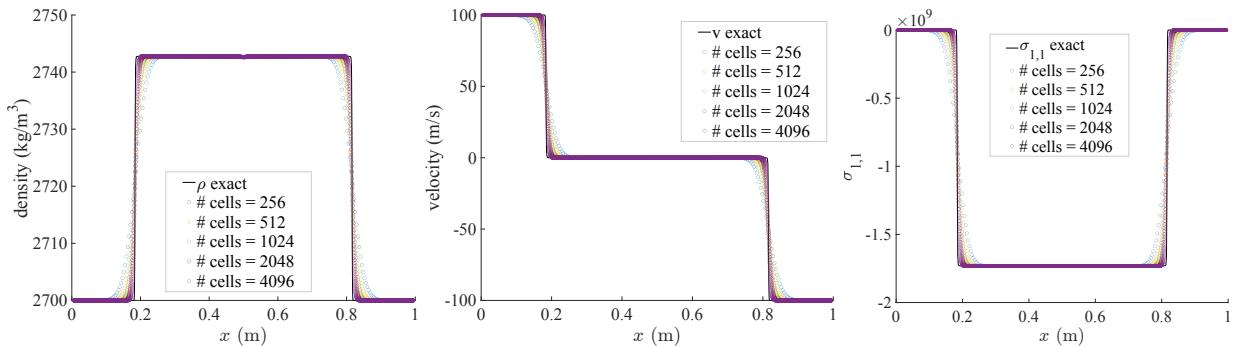


Figure 6.22: Solution to the double impact problem at $t = 0.00005$ with $\text{CFL} = .5$. From left to right: density, velocity, stress.

6.4.3 Shear Test

In this section, we consider a 2D shocktube with initial tangential velocity discontinuity. We take as our domain $D_0 = [0, 1] \times [0, h]$, discretized into $n_{\text{cells}} \times 1$ mesh, where $h = 1/n_{\text{cells}}$ so that

our 2D discretization of the mesh with an initial discontinuity at $x_0 = 0.5$. The initial conditions are set as

$$\mathbf{u}_0(\mathbf{x}) = (\rho, \mathbf{v}, p)^\top = \begin{cases} (2.7 \times 10^3, 0, 500, 1 \times 10^5)^\top & \text{if } 0 < x_1 \leq 0.5, \\ (2.7 \times 10^3, 0, -500, 1 \times 10^5)^\top & \text{if } 0.5 < x_1 < 1. \end{cases} \quad (6.4.3)$$

The results here are presented at final time $t_f = 0.00005$ using $\text{CFL} = 0.5$. For the shear energy, we use a Mooney-Rivlin type equation of state as given in equation 5.3.11.

In figure 6.23, we compare the results of our method to a reference solution from [35]. The reference solution is plotted as a solid black line.

Table 6.17: Convergence test for the 2D shear problem. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.

# cells	$\delta^1(5 \times 10^{-5})$	order	$\delta_\sigma^1(5 \times 10^{-5})$	order
3200	1.985E-1	—	9.186E-2	—
6400	1.411E-1	0.49	6.487E-2	0.50
12800	9.980E-2	0.50	4.557E-2	0.51
25600	7.013E-2	0.51	3.173E-2	0.52

6.4.3.1 Neo Hookean Results

We repeat the shear test, but this time with a Neo-Hookean equation of state for the shear energy, given in equation 5.3.10. We also consider slightly less shear velocity, with tangential velocity $v_y = -100$ m/s on the left and $v_y = 100$ m/s on the right. In figure 6.24, we compare the results of our method to a reference solution from [35]. The reference solution is plotted as a solid black line.

Remark 6.4.1 (Shear Convergence). *For the shear test, we observe a convergence rate for the Neo-Hookean material model of approximately 1 as seen in Table 6.18, whereas the Mooney-Rivlin model yields a lower rate of convergence around 0.5, as seen in Table 6.17. This discrepancy is*

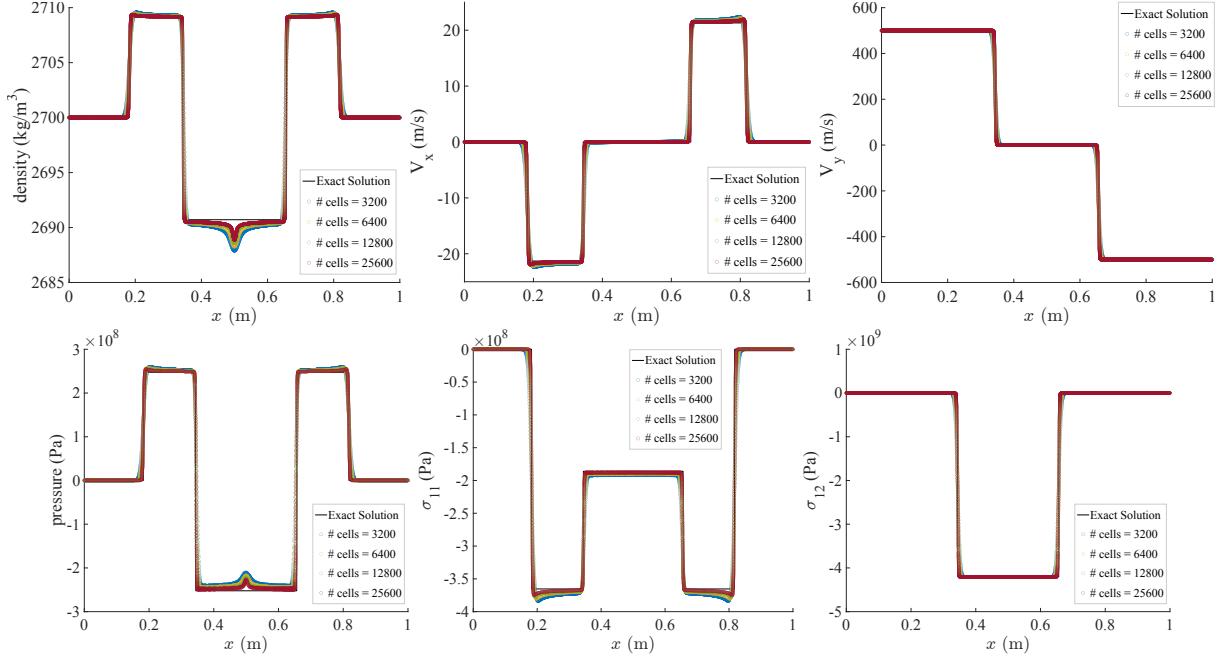


Figure 6.23: Solution to the Riemann problem (6.4.3) at $t = 0.00005$ with $\text{CFL} = .5$ using the Mooney-Rivlin closure model. From top left to bottom right: density, normal velocity, tangential velocity, pressure, normal stress, tangential stress.

Table 6.18: Convergence test for the 2D shear problem using the Neo-Hookean equation of state. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.

# cells	$\delta^1(5 \times 10^{-5})$	order	$\delta_\sigma^1(5 \times 10^{-5})$	order
3200	2.943E-1	—	1.265E-1	—
6400	1.818E-1	0.70	7.690E-2	0.72
12800	1.023E-1	0.83	4.222E-2	0.87
25600	4.920E-2	1.06	2.114E-2	1.00

not due to a deficiency in the numerical method, but rather stems from the construction of the exact solution used for error analysis. Specifically, for the Mooney-Rivlin case, the exact solution was reconstructed using digitized data from previously published graphical results. The limited resolution of the original plot introduces interpolation errors, thereby reducing the apparent convergence rate in the computed errors. This is true for the impact shear test as well, as observed in Table 6.19.

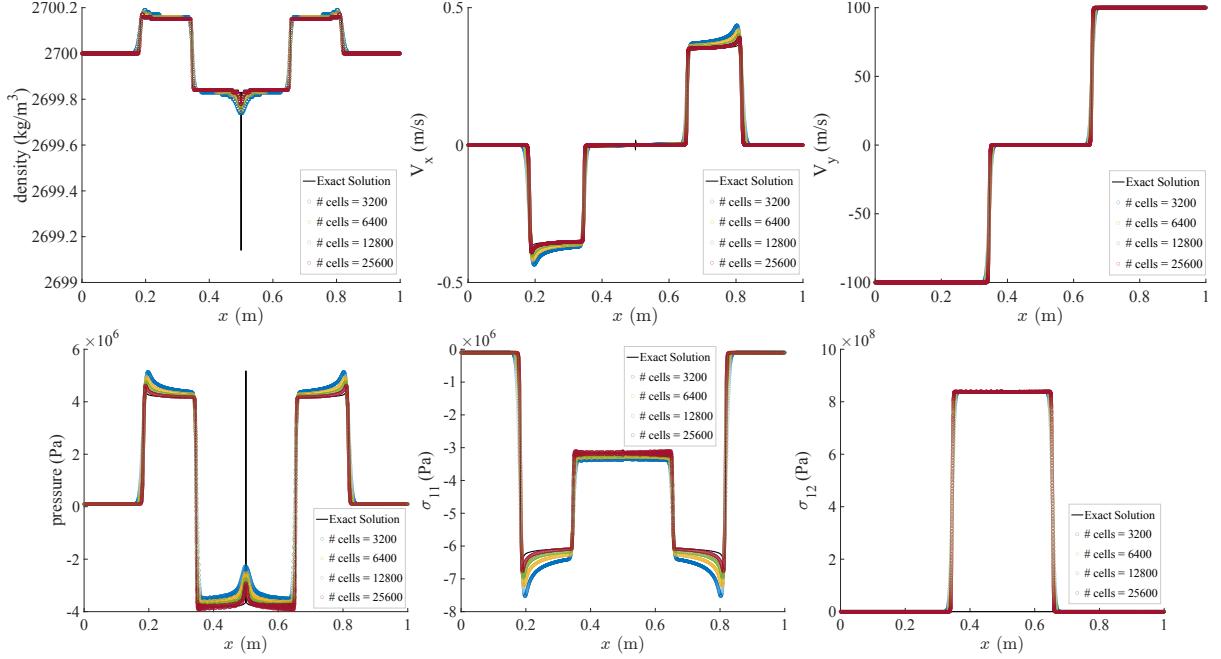


Figure 6.24: Solution to the Riemann problem (6.4.3) at $t = 0.00005$ with $\text{CFL} = .5$ using the Neo-Hookean closure model. From top left to bottom right: density, normal velocity, tangential velocity, pressure, normal stress, tangential stress.

6.4.4 Impact Shear Test

In this section we consider two colliding materials subject to strong shear velocity. For the initial conditions to the Riemann problem, we set

$$\mathbf{u}_0(\mathbf{x}) = (\rho, \mathbf{v}, p)^T = \begin{cases} (2.7 \times 10^3, 100, 500, 1 \times 10^5)^T & \text{if } 0 < x_1 \leq 0.5, \\ (2.7 \times 10^3, -100, -500, 1 \times 10^5)^T & \text{if } 0.5 < x_1 < 1. \end{cases} \quad (6.4.4)$$

The results here are presented at final time $t_f = 0.00005$ using $\text{CFL} = 0.5$. The domain $D_0 = [0, 1] \times [0, h]$ is discretized similar to as was done in the shear test. For the shear energy, we consider both a Mooney-Rivlin type equation of state 5.3.11 and an Aortic type equation of state 5.3.16.

In Fig.6.25, we present a comparison between the results obtained using our method and the reference solution reported in [35]. The comparison reveals strong agreement, with our computed profiles closely matching the reference across the domain. This consistency is further substantiated

in Table 6.19, where we report convergence data demonstrating that our approximation converges reliably and with good accuracy toward the reference solution.

Table 6.19: Convergence test for the 2D impact problem subject to shear velocity. Error indicators $\delta^1(t_f)$ and $\delta_\sigma^1(t_f)$ and convergence order obtained on different meshes.

# cells	$\delta^1(5 \times 10^{-5})$	order	$\delta_\sigma^1(5 \times 10^{-5})$	order
100	5.121E-1	—	3.114E-1	—
200	3.733E-1	0.46	2.204E-1	0.50
400	2.657E-1	0.49	1.575E-1	0.48
800	1.913E-1	0.47	1.124E-1	0.49
1600	1.367E-1	0.48	8.151E-2	0.46

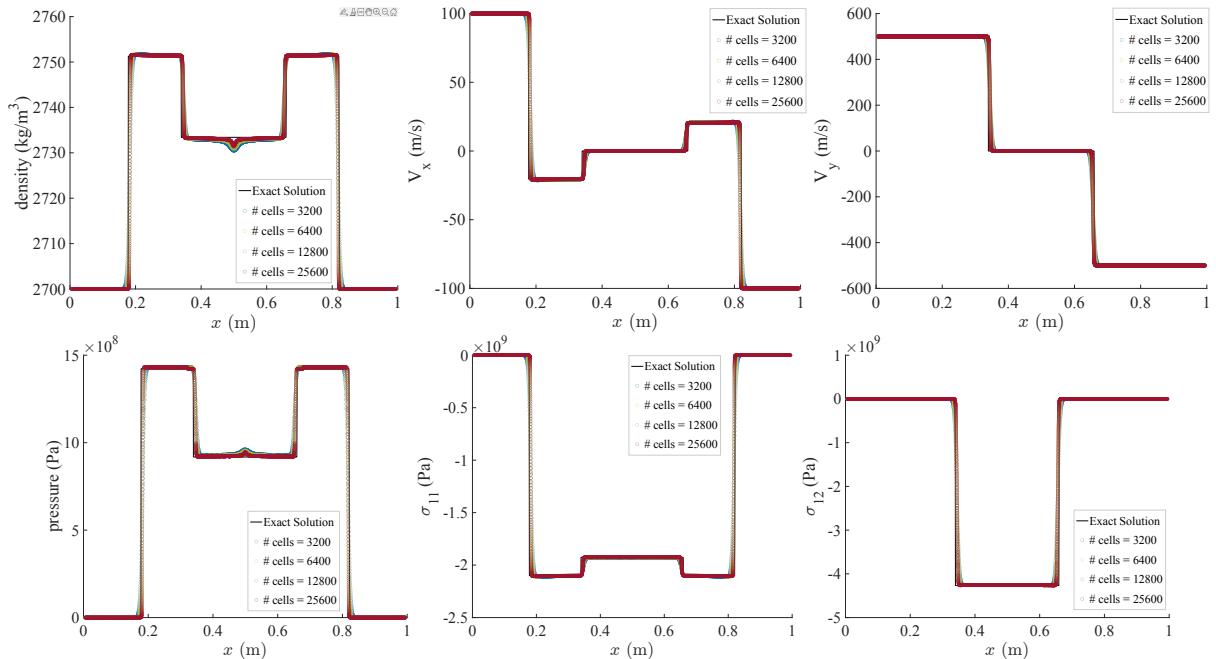


Figure 6.25: Solution to the Riemann problem (6.4.4) at $t = 0.00005$ with $CFL = .5$ and the Mooney-Rivlin equation of state using the Mooney-Rivlin closure model. From top left to bottom right: density, normal velocity, tangential velocity, pressure, normal stress, tangential stress.

6.4.5 Impact of a Projectile on a Solid Plate

We consider here the impact of a solid projectile on a solid plate, a problem initially introduced in [34]. We model a simplified version of the problem presented therein where we only consider the projectile and the plate, but not the surrounding air. The projectile has an initial velocity $v = 800$ m/s while the plate is initially at rest. The projectile is a square with side length 0.1 m and the plate is 0.5 m long and 0.1 m wide. Figure 6.26 shows the initial configuration.

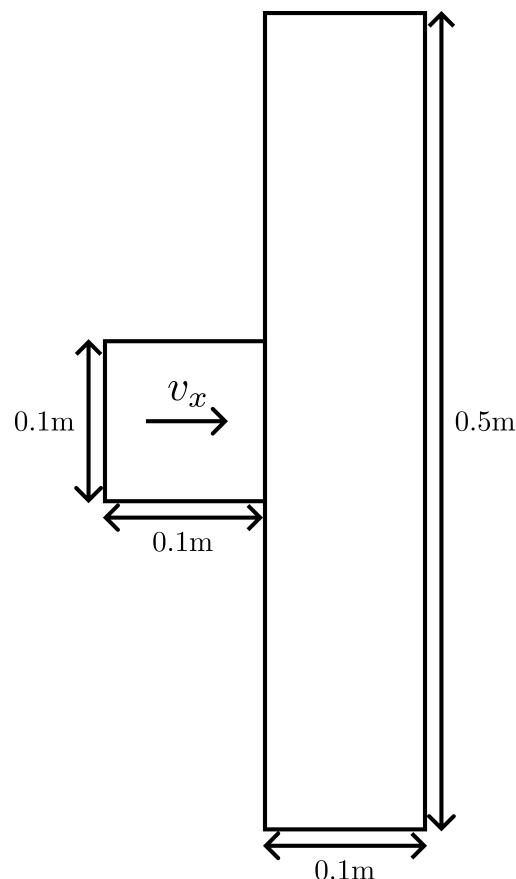


Figure 6.26: Initial configuration for the projectile-plate impact problem. The initial velocity of the projectile is 800 m/s while the plate is initially at rest.

To avoid simulation collapse due to mesh distortion and compression, the projectile and plate are in contact at initial time $t = 0$. The projectile contains 8×8 cells and the plate contains 8×40

cells. For this single material configuration, we use $\gamma = 4.22$, $p_\infty = 3.42 \times 10^{10}$ Pa, and $\rho_0 = 8900$ kg/m³.

6.4.5.1 Neo Hookean Results

We first begin by using the Neo-Hookean equation of state, given in Eqn. 5.3.10. We present our results using a variety of shear moduli: $\mu_s = 9.2 \times 10^{10}$ Pa corresponding to real copper at ambient temperature, $\mu_s = 9.2 \times 10^9$ corresponding to a softer material, and $\mu_s = 0$ corresponding to the fluid limit.

First we take $\mu_s = 9.2 \times 10^{10}$ which corresponds to the copper material. In Figure 6.27 we observe the density at various time snapshots, demonstrating nice bending and then elastic response. As would be expected for a Lagrangian method, when we consider the fluid case ($\mu_s =$

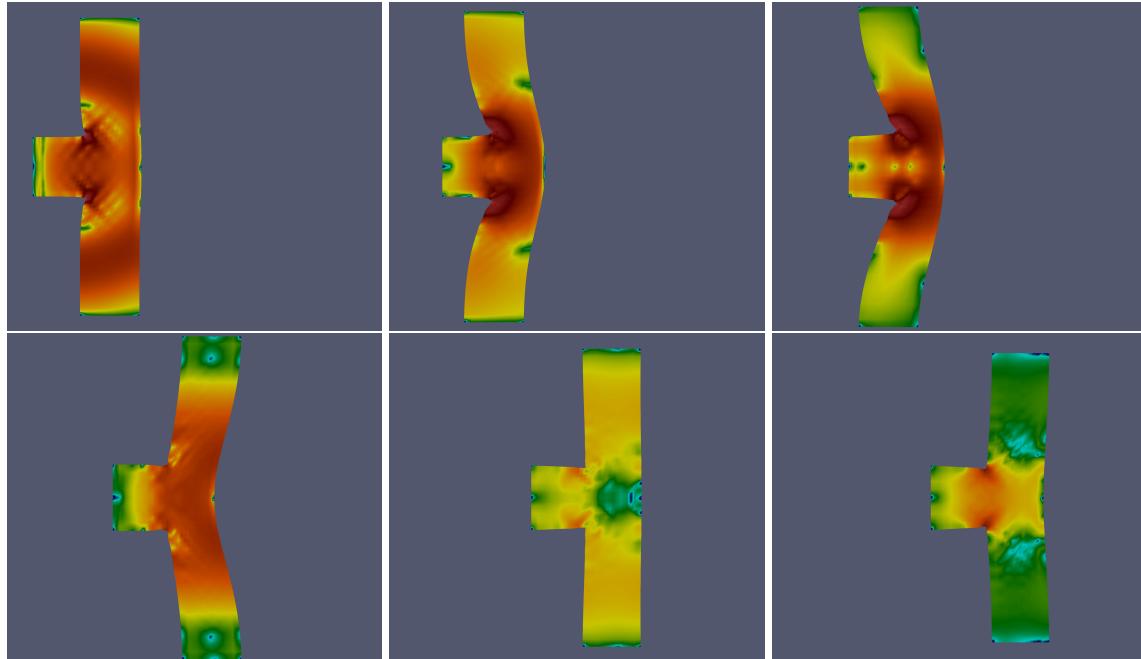


Figure 6.27: Projectile-plate impact problem with $\mu_s = 9.2 \times 10^9$. The schlieren image of the density is represented here at times $t = 3.2 \times 10^{-5}$, $t = 1.28 \times 10^{-4}$, $t = 2.27 \times 10^{-4}$, $t = 5.59 \times 10^{-4}$, $t = 9.16 \times 10^{-4}$, and $t = 1.417 \times 10^{-3}$. The Neo-Hookean equation of state is used for the sheer energy calculation.

0), we observe an early and quick deformation of the mesh as seen in Fig. 6.28.

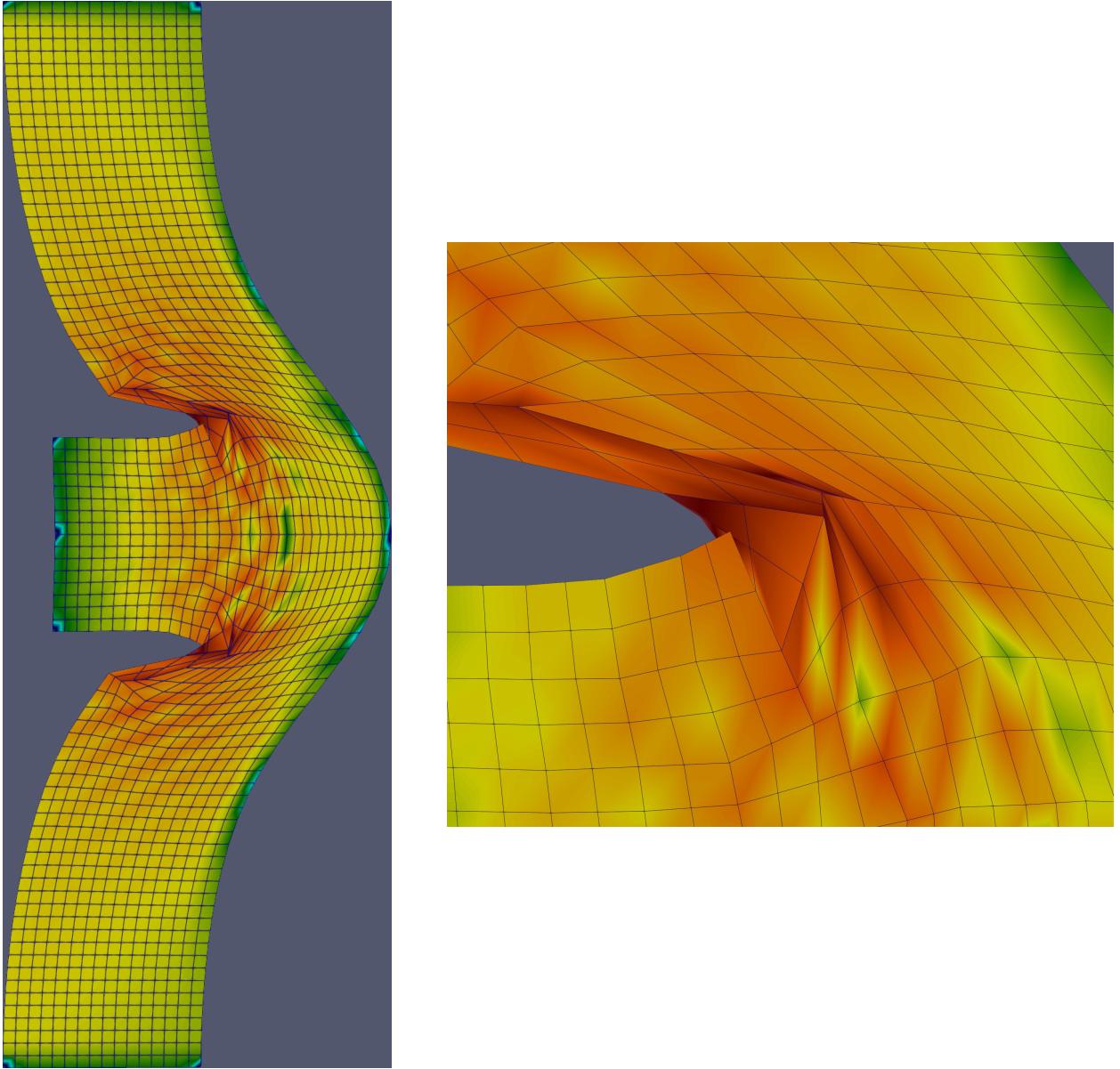


Figure 6.28: Projectile-plate impact problem with $\mu_s = 0$. The schlieren plot of the density is represented here at collapse time 2.12×10^{-4} , along with a zoom in on the cells causing the collapse. We observe large gradients in the density, which is the likely culprit to early termination of the simulation.

6.4.5.2 Aortic EOS

In this section, we present the first application of the anisotropic closure model to a fully two-dimensional impact scenario. Unlike the one-dimensional or quasi-one-dimensional cases studied

previously, the projectile–plate configuration provides a fully two-dimensional setting in which to examine the directional effects introduced by the anisotropic shear energy model. By prescribing different fiber orientations, we aim to highlight the emergence of anisotropic features in the velocity field, particularly in contrast to the symmetric behavior observed under the isotropic EOS discussed in the preceding section. This test case serves to validate the ability of the anisotropic model to capture directional effects in multidimensional impact dynamics.

In Fig. 6.29, we visualize the density field at three time snapshots for different fiber orientations $\theta = 0, \pi/4$, and $\pi/2$. When the fiber direction aligns with either the vertical or horizontal axis ($\theta = 0$ and $\theta = \pi/2$), we observe symmetric deformation patterns across the midline $y = 0$. This is consistent with the expected behavior of the anisotropic shear energy model when the fiber orientation is aligned with the primary axes of the domain geometry.

However, when the fiber angle is set to $\theta = \pi/4$, this symmetry is broken. The deformation and density patterns exhibit noticeable asymmetry, particularly near the impact interface, where directional stiffness introduced by the off-axis fiber alignment leads to uneven distribution of wave propagation and stress. These observations are further substantiated in Fig. 6.30, which tracks the tangential velocity component v_y at selected corner nodes. In this case, we see clear antisymmetric behavior between opposing sides of both the projectile and the plate.

Finally, Fig. 6.31 contrasts the normal and tangential velocity profiles between the anisotropic case ($\theta = \pi/4$) and the isotropic models. The comparison highlights how the isotropic cases maintain symmetry in the velocity field, while the anisotropic formulation introduces directional disparities. Together, these results illustrate the significant role that fiber orientation plays in governing the mechanical response under dynamic loading in this two-dimensional setting.

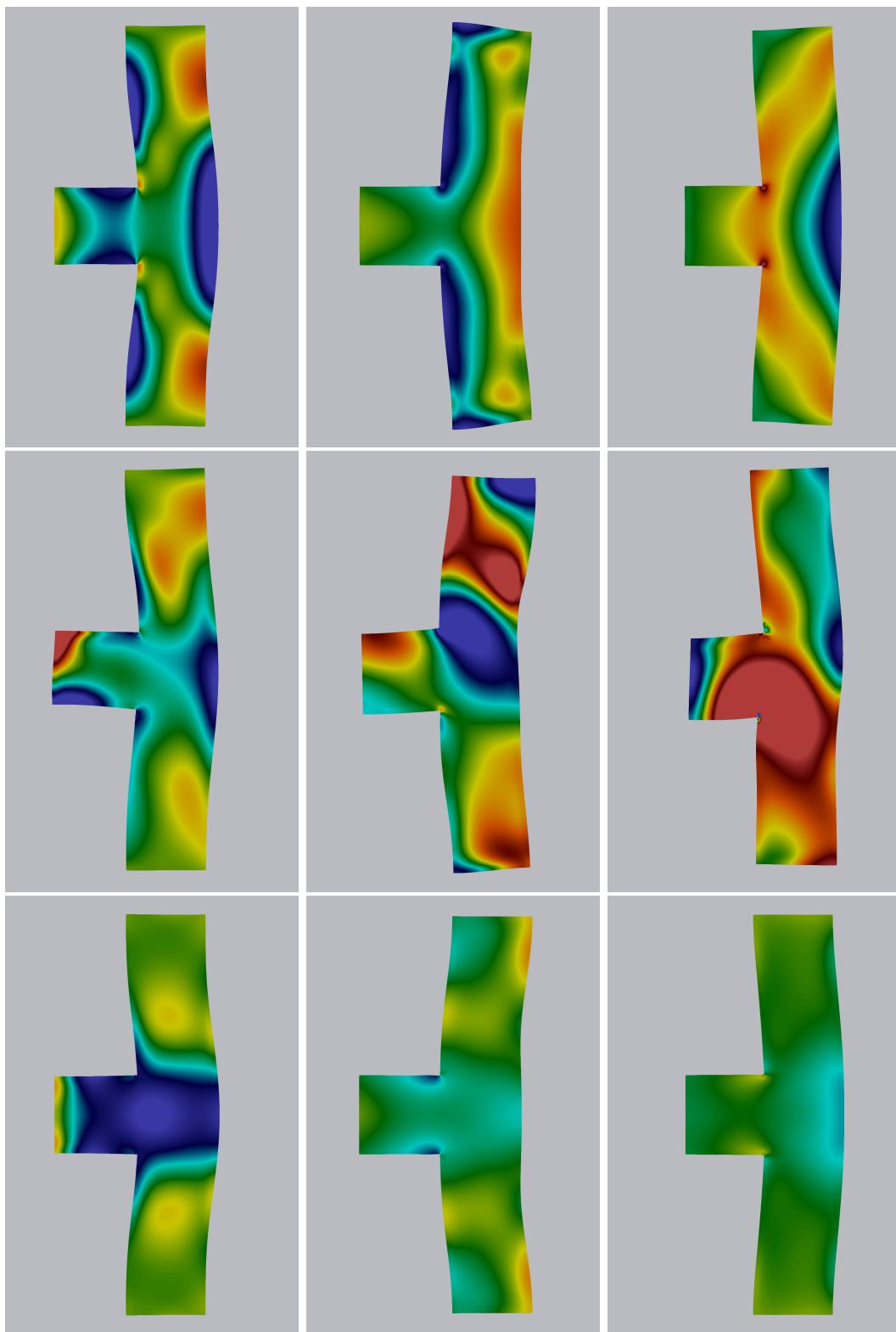


Figure 6.29: Evolution of the density field in the projectile–plate impact problem ($\mu_s = 9.2 \times 10^9$) using the anisotropic aortic equation of state. Time increases from left to right ($t = 6.5 \times 10^{-5}$, 1.62×10^{-4} , 2.93×10^{-4}), and fiber orientation varies from top to bottom ($\theta = 0, \pi/4, \pi/2$).

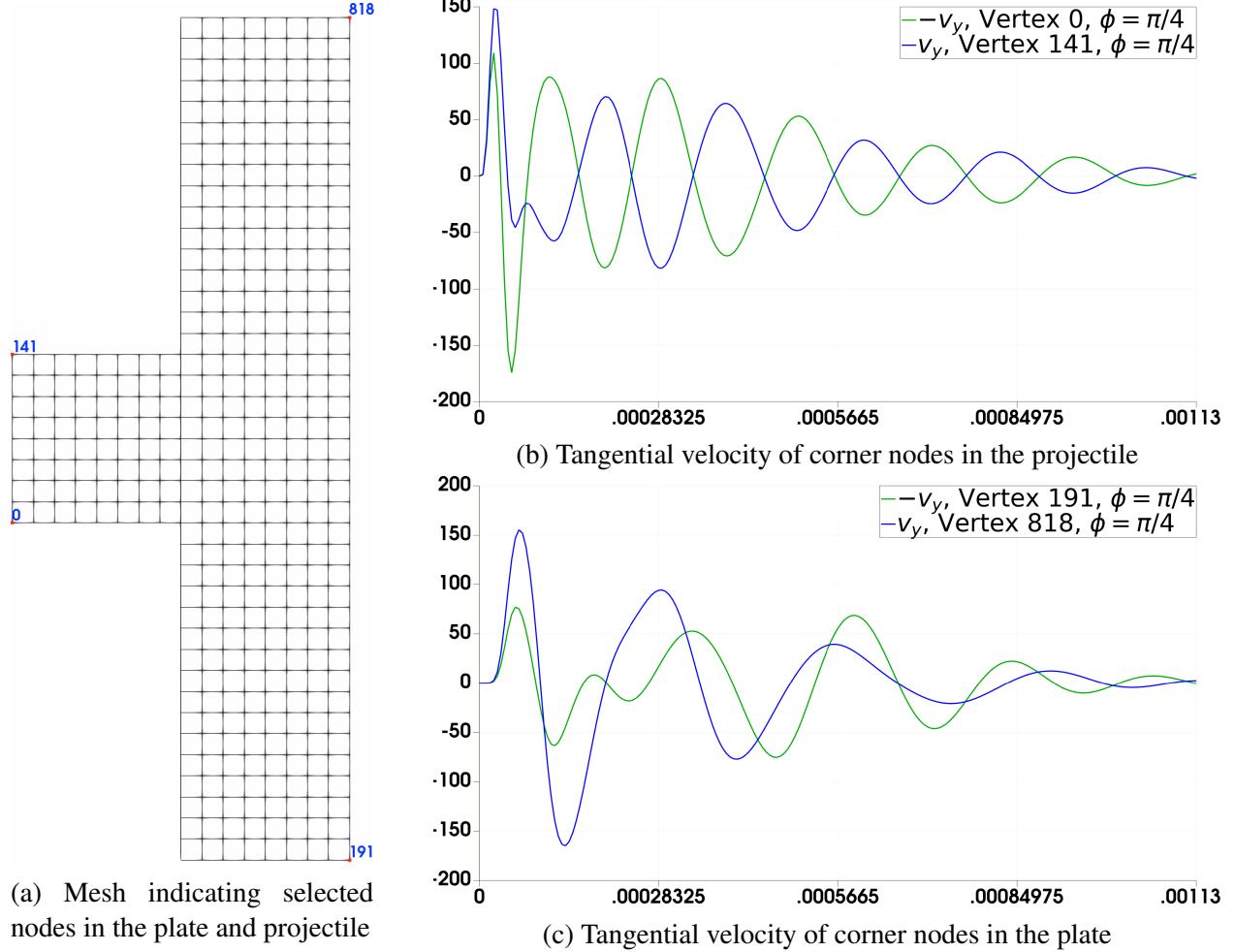


Figure 6.30: Projectile–plate impact problem using the anisotropic aortic equation of state defined in Eq. 5.3.16, with fiber orientation angle $\theta = \pi/4$. The plots compare the tangential velocity component v_y at selected corner nodes indicated in the mesh (a). Subfigures (b) and (c) illustrate the temporal evolution of v_y for opposing corners in the projectile and plate, respectively, highlighting the directional dependence introduced by the anisotropic fiber alignment.

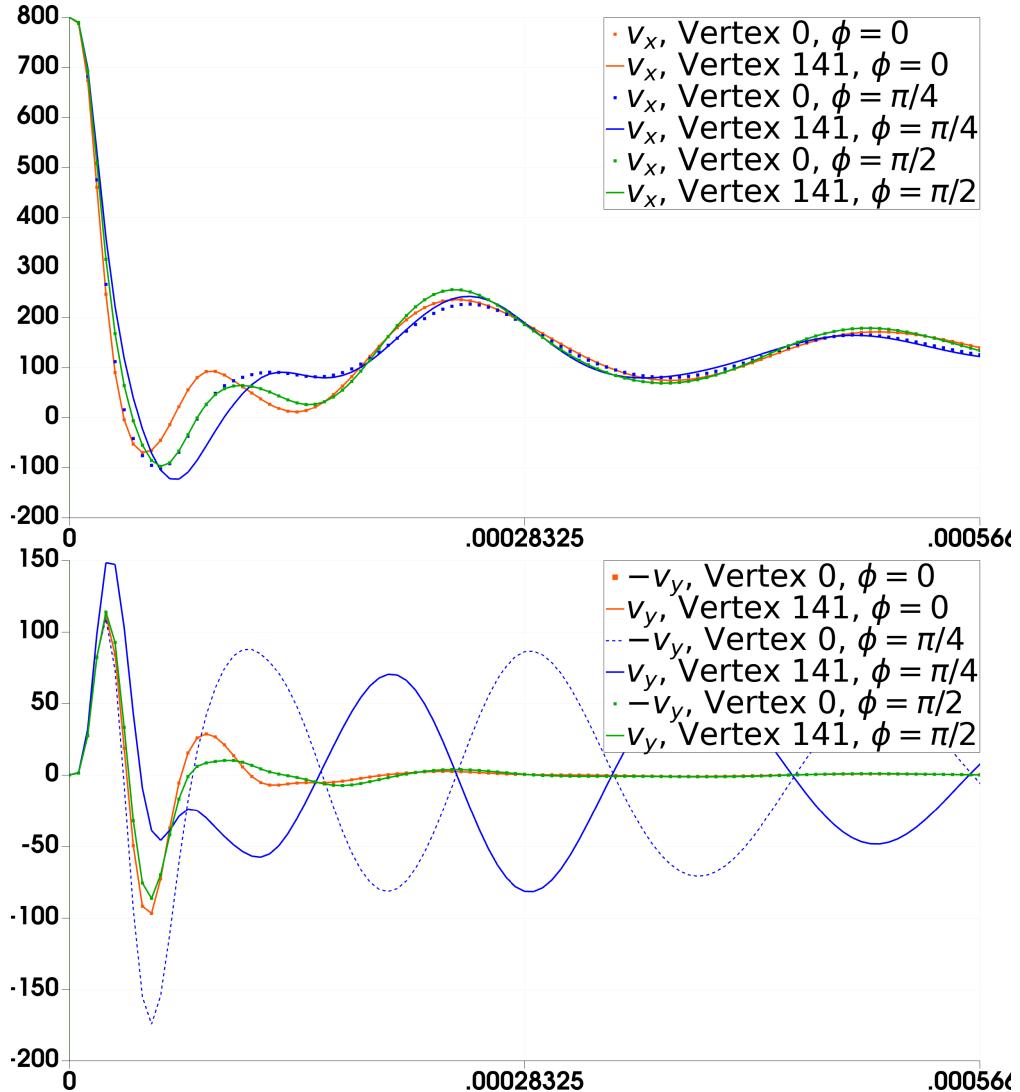


Figure 6.31: Projectile-plate impact problem comparing fiber orientations at angles $\theta = 0, \pi/4$, and $\pi/2$. When the equation of state is isotropic (i.e., for $\theta = 0$ and $\pi/2$), the velocity profiles exhibit symmetry across opposing vertices. In contrast, for $\theta = \pi/4$, we observe an antisymmetric behavior in the tangential velocity component v_y , indicating directional sensitivity introduced by the fiber orientation.

REFERENCES

- [1] F L Addessio, J R Baumgardner, J K Dukowicz, N L Johnson, B A Kashiwa, R M Rauenzahn, and C Zemach. Caveat: A computer code for fluid dynamics problems with large distortion and internal slip. Technical report, Los Alamos National Lab., NM (United States), 05 1992. URL <https://www.osti.gov/biblio/5366795>.
- [2] Robert W. Anderson, Veselin A. Dobrev, Tzanio V. Kolev, Robert N. Rieben, and Vladimir Z. Tomov. High-order multi-material ALE hydrodynamics. *SIAM J. Sci. Comput.*, 40(1):B32–B58, 2018.
- [3] Andrew J. Barlow, Pierre-Henri Maire, William J. Rider, Robert N. Rieben, and Mikhail J. Shashkov. Arbitrary lagrangian-eulerian methods for modeling high-speed compressible multimaterial flows. *J. Comput. Phys.*, 322(C):603–665, October 2016. ISSN 0021-9991. doi: 10.1016/j.jcp.2016.07.001. URL <https://doi.org/10.1016/j.jcp.2016.07.001>.
- [4] H. A. Bethe. The theory of shock waves for an arbitrary equation of state. Technical Report Report 545, Office of Scientific Research and Development, Washington, D.C., 1942.
- [5] Walter Boscheri, Michael Dumbser, Raphaël Loubère, and Pierre-Henri Maire. A second-order cell-centered lagrangian ader-mood finite volume scheme on multidimensional unstructured meshes for hydrodynamics. *Journal of Computational Physics*, 358:103–129, 2018. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2017.12.040>.
- [6] Walter Boscheri, Raphaël Loubère, and Pierre-Henri Maire. A 3d cell-centered ader mood finite volume method for solving updated lagrangian hyperelasticity on unstructured grids. *Journal of Computational Physics*, 449:110779, 2022. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2021.110779>. URL <https://www.sciencedirect.com/science/article/pii/S0021999121006744>.
- [7] Walter Boscheri, Michael Dumbser, and Pierre-Henri Maire. A new thermodynamically compatible finite volume scheme for lagrangian gas dynamics. *SIAM Journal on Scientific*

Computing, 46(4):A2224–A2247, 2024. doi: 10.1137/23M1580863.

- [8] Erik Burman, Alexandre Ern, and Miguel Fernández. Explicit runge–kutta schemes and finite elements with symmetric stabilization for first-order linear pde systems. SIAM Journal on Numerical Analysis, 48, 12 2010. doi: 10.1137/090757940.
- [9] D. E. Burton. Conservation of energy, momentum, and angular momentum in Lagrangian staggered-grid hydrodynamics. In Presented at the 1990 Nuclear Explosives Code Developers' Conference, pages 6–9, November 1990.
- [10] Herbert B Callen. Thermodynamics and an introduction to thermostatistics; 2nd ed. Wiley, New York, NY, 1985. URL <https://cds.cern.ch/record/450289>.
- [11] E. J. Caramana, D. E. Burton, M. J. Shashkov, and P. P. Whalen. The construction of compatible hydrodynamics algorithms utilizing conservation of total energy. J. Comput. Phys., 146(1):227–262, 1998.
- [12] E.J. Caramana and M.J. Shashkov. Elimination of artificial grid distortion and hourglass-type motions by means of lagrangian subzonal masses and pressures. Journal of Computational Physics, 142(2):521–561, 1998. ISSN 0021-9991. doi: <https://doi.org/10.1006/jcph.1998.5952>. URL <https://www.sciencedirect.com/science/article/pii/S0021999198959526>.
- [13] E.J. Caramana, M.J. Shashkov, and P.P. Whalen. Formulations of artificial viscosity for multi-dimensional shock wave computations. Journal of Computational Physics, 144(1):70–97, 1998. ISSN 0021-9991. doi: <https://doi.org/10.1006/jcph.1998.5989>. URL <https://www.sciencedirect.com/science/article/pii/S0021999198959897>.
- [14] G. Carré, S. Del Pino, B. Després, and E. Labourasse. A cell-centered Lagrangian hydrodynamics scheme on general unstructured meshes in arbitrary dimension. J. Comput. Phys., 228(14):5160–5183, 2009.
- [15] Krit Chaimoon and Prinya Chindaprasirt. An anisotropic hyperelastic model with an application to soft tissues. European Journal of Mechanics - A/Solids, 78:103845, 2019. ISSN 0997-7538. doi: <https://doi.org/10.1016/j.euromechsol.2019.103845>. URL <https://www.sciencedirect.com/science/article/pii/S0997753819300607>.

- //www.sciencedirect.com/science/article/pii/S0997753818309884.
- [16] Juan Cheng and Chi-Wang Shu. Positivity-preserving lagrangian scheme for multi-material compressible flow. *J. Comput. Phys.*, 257(PA):143–168, January 2014. ISSN 0021-9991.
- [17] Alexandre J. Chorin and Jerrold E. Marsden. *A Mathematical Introduction to Fluid Mechanics*. Springer New York, 1993. doi: 10.1007/978-1-4612-0883-9. URL <https://app.dimensions.ai/details/publication/pub.1009952911>.
- [18] P. G. Ciarlet. *The finite element method for elliptic problems*, volume 40 of *Classics in Applied Mathematics*. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2002. Reprint of the 1978 original [North-Holland, Amsterdam, The Netherlands].
- [19] Bennett Clayton, Jean-Luc Guermond, and Bojan Popov. Invariant domain-preserving approximations for the Euler equations with tabulated equation of state. *SIAM J. Sci. Comput.*, 44(1):A444–A470, 2022.
- [20] Bennett Clayton, Jean-Luc Guermond, Matthias Maier, Bojan Popov, and Eric J. Tovar. Robust second-order approximation of the compressible Euler equations with an arbitrary equation of state. *J. Comput. Phys.*, 478:Paper No. 111926, 21, 2023.
- [21] Bennett Clayton, Jean-Luc Guermond, Matthias Maier, Bojan Popov, and Eric J. Tovar. Robust second-order approximation of the compressible euler equations with an arbitrary equation of state. *Journal of Computational Physics*, 478:111926, 2023. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2023.111926>. URL <https://www.sciencedirect.com/science/article/pii/S0021999123000219>.
- [22] Bernardo Cockburn, San-Yih Lin, and Chi-Wang Shu. Tvb runge-kutta local projection discontinuous galerkin finite element method for conservation laws iii: One-dimensional systems. *Journal of Computational Physics*, 84(1):90–113, 1989. ISSN 0021-9991. doi: [https://doi.org/10.1016/0021-9991\(89\)90183-6](https://doi.org/10.1016/0021-9991(89)90183-6). URL <https://www.sciencedirect.com/science/article/pii/0021999189901836>.
- [23] Scott D. Crockett and Franz J. Freibert. Equation of state: Manhattan project developments and beyond. *Nuclear Technology*, 207(sup1):S286–S294, 2021. doi: 10.1080/00295450.

- 2021.1913036. URL <https://doi.org/10.1080/00295450.2021.1913036>.
- [24] Stéphane Del Pino, Bruno Després, and Alexiane Plessier. An implicit well-posed Lagrangian scheme for non viscous compressible fluids in high dimension. working paper or preprint, March 2024.
 - [25] Fabian Denner. The gilmore-nasg model to predict single-bubble cavitation in compressible liquids. *Ultrasonics Sonochemistry*, 70:105307, 2021. ISSN 1350-4177. doi: <https://doi.org/10.1016/j.ultsonch.2020.105307>. URL <https://www.sciencedirect.com/science/article/pii/S1350417720309779>.
 - [26] Bruno Després. Numerical methods for Eulerian and Lagrangian conservation laws. Frontiers in Mathematics. Birkhäuser/Springer, Cham, 2017.
 - [27] Bruno Després and Constant Mazeran. Lagrangian Gas Dynamics in Two Dimensions and Lagrangian systems. *Archive for Rational Mechanics and Analysis*, 178(3):327–372, December 2005. doi: 10.1007/s00205-005-0375-4.
 - [28] Veselin A. Dobrev, Tzanio V. Kolev, and Robert N. Rieben. High-order curvilinear finite element methods for Lagrangian hydrodynamics. *SIAM J. Sci. Comput.*, 34(5):B606–B641, 2012.
 - [29] John K Dukowicz and Bertrand JA Meltz. Vorticity errors in multidimensional lagrangian codes. *Journal of Computational Physics*, 99(1):115–134, 1992.
 - [30] Michael Dumbser and Eleuterio F. Toro. On universal osher-type schemes for general nonlinear hyperbolic conservation laws. *Communications in Computational Physics*, 10(3):635–671, 2011. doi: 10.4208/cicp.170610.021210a.
 - [31] Nira Dyn, David Levine, and John A. Gregory. A butterfly subdivision scheme for surface interpolation with tension control. *ACM Trans. Graph.*, 9(2):160–169, April 1990. ISSN 0730-0301. doi: 10.1145/78956.78958. URL <https://doi.org/10.1145/78956.78958>.
 - [32] Alexandre Ern and Jean-Luc Guermond. Finite elements III : first-order and time-dependent PDEs. Texts in Applied Mathematics ; Volume 74. Springer, Cham, Switzerland, 2021. ISBN

3-030-57348-6.

- [33] Alexandre Ern and Jean-Luc Guermond. Invariant-domain-preserving high-order time stepping: I. explicit runge–kutta schemes. *SIAM Journal on Scientific Computing*, 44(5):A3366–A3392, 2022. doi: 10.1137/21M145793X. URL <https://doi.org/10.1137/21M145793X>.
- [34] N. Favrie, S.L. Gavrilyuk, and R. Saurel. Solid–fluid diffuse interface model in cases of extreme deformations. *Journal of Computational Physics*, 228(16):6037–6077, 2009. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2009.05.015>. URL <https://www.sciencedirect.com/science/article/pii/S0021999109002654>.
- [35] N. Favrie, S. Gavrilyuk, and S. Ndanou. A thermodynamically compatible splitting procedure in hyperelasticity. *Journal of Computational Physics*, 270:300–324, 2014. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2014.03.051>. URL <https://www.sciencedirect.com/science/article/pii/S0021999114002381>.
- [36] Ronald P Fedkiw, Tariq Aslam, Barry Merriman, and Stanley Osher. A non-oscillatory eulerian approach to interfaces in multimaterial flows (the ghost fluid method). *Journal of Computational Physics*, 152(2):457–492, 1999. ISSN 0021-9991. doi: <https://doi.org/10.1006/jcph.1999.6236>. URL <https://www.sciencedirect.com/science/article/pii/S0021999199962368>.
- [37] Wenjing Feng, Juan Cheng, and Chi-Wang Shu. Second order conservative lagrangian dg schemes for compressible flow and their application in preserving spherical symmetry in two-dimensional cylindrical geometry. *Journal of Computational Physics*, 521:113530, 2025. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2024.113530>. URL <https://www.sciencedirect.com/science/article/pii/S0021999124007782>.
- [38] L. Formaggia and F. Nobile. A stability analysis for the arbitrary Lagrangian Eulerian formulation with finite elements. *East-West J. Numer. Math.*, 7(2):105–131, 1999. ISSN 0928-0200.
- [39] Stéphane Galera, Pierre-Henri Maire, and Jérôme Breil. A two-dimensional unstructured

- cell-centered multi-material ale scheme using vof interface reconstruction. *J. Comput. Phys.*, 229(16):5755–5787, August 2010. ISSN 0021-9991. doi: 10.1016/j.jcp.2010.04.019. URL <https://doi.org/10.1016/j.jcp.2010.04.019>.
- [40] Sergei K. Godunov and I. Bohachevsky. Finite difference method for numerical computation of discontinuous solutions of the equations of fluid dynamics. *Matematicheskij sbornik*, 47(89)(3):271–306, 1959. URL <https://hal.science/hal-01620642>.
- [41] Henri Gouin and Jean-François Debieve. Variational principle involving the stress tensor in elastodynamics. *International Journal of Engineering Science*, 24(7):1057–1066, 1986. ISSN 0020-7225. doi: [https://doi.org/10.1016/0020-7225\(86\)90001-7](https://doi.org/10.1016/0020-7225(86)90001-7). URL <https://www.sciencedirect.com/science/article/pii/0020722586900017>.
- [42] Jean-Luc Guermond and Zuodong Wang. Mass conservative limiting and applications to the approximation of the steady-state radiation transport equations. *Journal of Computational Physics*, 521:113531, 2025. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2024.113531>. URL <https://www.sciencedirect.com/science/article/pii/S0021999124007794>.
- [43] Jean-Luc Guermond, Bojan Popov, Laura Saavedra, and Yong Yang. Invariant domains preserving arbitrary Lagrangian Eulerian approximation of hyperbolic systems with continuous finite elements. *SIAM J. Sci. Comput.*, 39(2):A385–A414, 2017.
- [44] Jean-Luc Guermond, Murtazo Nazarov, Bojan Popov, and Ignacio Tomas. Second-order invariant domain preserving approximation of the euler equations using convex limiting. *SIAM Journal on Scientific Computing*, 40(5):A3211–A3239, 2018. doi: 10.1137/17M1149961. URL <https://doi.org/10.1137/17M1149961>.
- [45] Jean-Luc Guermond, Bojan Popov, Laura Saavedra, and Madison Sheridan. Invariant-domain preserving and locally mass conservative approximation of the lagrangian hydrodynamics equations. *Computer Methods in Applied Mechanics and Engineering*, 441:117927, 2025. ISSN 0045-7825. doi: <https://doi.org/10.1016/j.cma.2025.117927>. URL <https://www.sciencedirect.com/science/article/pii/S0045782525001999>.

- [46] Sarah Hank, Nicolas Favrie, and Jacques Massoni. Modeling hyperelasticity in non-equilibrium multiphase flows. *Journal of Computational Physics*, 330:65–91, 2017. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2016.11.001>. URL <https://www.sciencedirect.com/science/article/pii/S0021999116305800>.
- [47] Francis H. Harlow and Anthony A. Amsden. Fluid dynamics. Technical Report LA-4700, Los Alamos Scientific Laboratory, Los Alamos, NM 87544, June 1971.
- [48] Ami Harten and Stanley Osher. Uniformly high-order accurate nonoscillatory schemes. i. *SIAM Journal on Numerical Analysis*, 24(2):279–309, 1987. doi: 10.1137/0724022. URL <https://doi.org/10.1137/0724022>.
- [49] Gustave Adolphe Hirn. *Exposition analytique et expérimentale de la théorie mécanique de la chaleur. Tome premier*. Gauthier-Villars, Imprimeur-Libraire, Paris, troisième édition / entièrement refondue. edition, 1875.
- [50] C.W Hirt and B.D Nichols. Volume of fluid (vof) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1):201–225, 1981. ISSN 0021-9991. doi: [https://doi.org/10.1016/0021-9991\(81\)90145-5](https://doi.org/10.1016/0021-9991(81)90145-5). URL <https://www.sciencedirect.com/science/article/pii/0021999181901455>.
- [51] C.W Hirt, A.A Amsden, and J.L Cook. An arbitrary lagrangian-eulerian computing method for all flow speeds. *Journal of Computational Physics*, 14(3):227–253, 1974. ISSN 0021-9991. doi: [https://doi.org/10.1016/0021-9991\(74\)90051-5](https://doi.org/10.1016/0021-9991(74)90051-5). URL <https://www.sciencedirect.com/science/article/pii/0021999174900515>.
- [52] Pierre-Henri Hugoniot. Memoir on the propagation of movements in bodies, especially perfect gases (second part). *Journal de l'école Polytechnique*, 1:581, 1889.
- [53] Zupeng Jia and Shudao Zhang. A new high-order discontinuous galerkin spectral finite element method for lagrangian gas dynamics in two-dimensions. *Journal of Computational Physics*, 230(7):2496–2522, 2011. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2010.12.023>. URL <https://www.sciencedirect.com/science/article/pii/S002199911000687X>.

- [54] Cheng Juan and Chi-Wang Shu. A third order conservative lagrangian type scheme on curvilinear meshes for the compressible euler equations. *Communications in Computational Physics*, 4(5):1008–1024, 2008. ISSN 1991-7120. doi: <https://doi.org/2008-CiCP-7824>. URL <https://global-sci.com/article/81289/a-third-order-conservative-lagrangian-type-scheme-on-curvilinear-meshes>
- [55] G. Kluth and B. Després. Discretization of hyperelasticity on unstructured mesh with a cell-centered lagrangian scheme. *Journal of Computational Physics*, 229(24):9092–9118, 2010. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2010.08.024>. URL <https://www.sciencedirect.com/science/article/pii/S0021999110004717>.
- [56] P. D. Lax. Hyperbolic systems of conservation laws ii. *Communications on Pure and Applied Mathematics*, 10(4):537–566, 1957. doi: <https://doi.org/10.1002/cpa.3160100406>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/cpa.3160100406>.
- [57] Michel Lesoinne and Charbel Farhat. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Computer Methods in Applied Mechanics and Engineering*, 134(1):71–90, 1996.
- [58] Xiaodong Liu, Nathaniel R. Morgan, and Donald E. Burton. A lagrangian discontinuous galerkin hydrodynamic method. *Computers & Fluids*, 163:68–85, 2018. ISSN 0045-7930. doi: <https://doi.org/10.1016/j.compfluid.2017.12.007>. URL <https://www.sciencedirect.com/science/article/pii/S0045793017304449>.
- [59] R Loubere, PH Maire, and P Váchal. Staggered lagrangian discretization based on cell-centered riemann solver and associated hydrodynamics scheme. *Communication in Computational Physics*, 10(4):940–978, 2011.
- [60] R. Loubère, J. Ovadia, and R. Abgrall. A lagrangian discontinuous galerkin-type method on unstructured meshes to solve hydrodynamics problems. *International Journal for Numerical Methods in Fluids*, 44(6):645–663, 2004. doi: <https://doi.org/10.1002/fld.665>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.665>.

- [61] Raphaël Loubère, Pierre-Henri Maire, Mikhail Shashkov, Jérôme Breil, and Stéphane Galera. Reale: A reconnection-based arbitrary-lagrangian–eulerian method. *Journal of Computational Physics*, 229(12):4724–4761, 2010. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2010.03.011>. URL <https://www.sciencedirect.com/science/article/pii/S002199911000121X>.
- [62] P.H. Maire and Boniface Nkonga. Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics. *Journal of Computational Physics*, Volume 228(Issue 3, 20 February 2009):Pages 799–821, 2009. doi: 10.1016/j.jcp.2008.10.012. URL <https://inria.hal.science/inria-00290717>.
- [63] Pierre-Henri Maire. A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes. *J. Comput. Phys.*, 228(7):2391–2425, 2009. ISSN 0021-9991. doi: 10.1016/j.jcp.2008.12.007.
- [64] Pierre-Henri Maire, Rémi Abgrall, Jérôme Breil, and Jean Ovadia. A cell-centered lagrangian scheme for two-dimensional compressible flow problems. *SIAM Journal on Scientific Computing*, 29(4):1781–1824, 2007.
- [65] Pierre-Henri Maire, Jérôme Breil, and Stéphane Galera. A cell-centered arbitrary lagrangian–eulerian (ale) method. *International Journal for Numerical Methods in Fluids*, 56: 1161 – 1166, 03 2008. doi: 10.1002/fld.1557.
- [66] Olivier Le Métayer and Richard Saurel. The noble-abel stiffened-gas equation of state. *Physics of Fluids*, 28(4):046102, 2016. doi: 10.1063/1.4945981. URL <https://doi.org/10.1063/1.4945981>.
- [67] M. Mooney. A theory of large elastic deformation. *Journal of Applied Physics*, 11(9):582–592, 1940. ISSN 0021-8979. doi: 10.1063/1.1712836. URL <https://doi.org/10.1063/1.1712836>.
- [68] S. Ndanou, N. Favrie, and S. Gavrilyuk. Multi-solid and multi-fluid diffuse interface model: Applications to dynamic fracture and fragmentation. *Journal of Computational Physics*, 295:523–555, 2015. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2015.07.030>.

- 04.024. URL <https://www.sciencedirect.com/science/article/pii/S0021999115002806>.
- [69] Loann Neron and Richard Saurel. Noble–abel/first-order virial equations of state for gas mixtures resulting from multiple condensed reactive materials combustion. *Physics of Fluids*, 34(1), January 2022. ISSN 1089-7666. doi: 10.1063/5.0079187. URL <http://dx.doi.org/10.1063/5.0079187>.
- [70] W. F. Noh and Paul Woodward. Slic (simple line interface calculation). In Adriaan I. van de Vooren and Pieter J. Zandbergen, editors, *Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics June 28 – July 2, 1976 Twente University, Enschede*, pages 330–340, Berlin, Heidelberg, 1976. Springer Berlin Heidelberg. ISBN 978-3-540-37548-7.
- [71] Stanley Osher and Ronald P. Fedkiw. Level set methods: An overview and some recent results. *Journal of Computational Physics*, 169(2):463–502, 2001. ISSN 0021-9991. doi: <https://doi.org/10.1006/jcph.2000.6636>. URL <https://www.sciencedirect.com/science/article/pii/S0021999100966361>.
- [72] Cosmin G. Petra, NaiYuan Chiang, and Jingyi Wang. HiOp – User Guide. Technical Report LLNL-SM-743591, Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, 2018.
- [73] William John Macquorn Rankine. Xv. on the thermodynamic theory of waves of finite longitudinal disturbance. *Philosophical Transactions of the Royal Society of London*, 160:277–288, 1870. doi: 10.1098/rstl.1870.0015. URL <https://royalsocietypublishing.org/doi/abs/10.1098/rstl.1870.0015>.
- [74] R. S. Rivlin. Large elastic deformations of isotropic materials iv. further developments of the general theory. *Philosophical Transactions of the Royal Society of London. Series A, Mathematical and Physical Sciences*, 241(835):379–397, 1948. doi: 10.1098/rsta.1948.0024. URL <https://royalsocietypublishing.org/doi/abs/10.1098/rsta.1948.0024>.

- [75] Birte Schmidtmann, Rémi Abgrall, and Manuel Torrilhon. On third-order limiter functions for finite volume methods. *Bulletin of the Brazilian Mathematical Society, New Series*, 47(2):753–764, 2016. ISSN 1678-7714. doi: 10.1007/s00574-016-0183-5. URL <https://doi.org/10.1007/s00574-016-0183-5>.
- [76] Jörg Schröder and Patrizio Neff. Invariant formulation of hyperelastic transverse isotropy based on polyconvex free energy functions. *International Journal of Solids and Structures*, 40(2):401–445, 2003. ISSN 0020-7683. doi: [https://doi.org/10.1016/S0020-7683\(02\)00458-4](https://doi.org/10.1016/S0020-7683(02)00458-4). URL <https://www.sciencedirect.com/science/article/pii/S0020768302004584>.
- [77] G. Scovazzi. Stabilized shock hydrodynamics: II. design and physical interpretation of the supg operator for lagrangian computations. *Computer Methods in Applied Mechanics and Engineering*, 196(4):967–978, 2007. ISSN 0045-7825. doi: <https://doi.org/10.1016/j.cma.2006.08.009>. URL <https://www.sciencedirect.com/science/article/pii/S0045782506002386>.
- [78] G. Scovazzi and E. Love. A generalized view on galilean invariance in stabilized compressible flow computations. *International Journal for Numerical Methods in Fluids*, 64(10-12):1065–1083, 2010. doi: <https://doi.org/10.1002/fld.2417>. URL <https://onlinelibrary.wiley.com/doi/abs/10.1002/fld.2417>.
- [79] G. Scovazzi, M.A. Christon, T.J.R. Hughes, and J.N. Shadid. Stabilized shock hydrodynamics: I. a lagrangian method. *Computer Methods in Applied Mechanics and Engineering*, 196(4):923–966, 2007. ISSN 0045-7825. doi: <https://doi.org/10.1016/j.cma.2006.08.008>. URL <https://www.sciencedirect.com/science/article/pii/S0045782506002374>.
- [80] G. Scovazzi, J.N. Shadid, E. Love, and W.J. Rider. A conservative nodal variational multiscale method for lagrangian shock hydrodynamics. *Computer Methods in Applied Mechanics and Engineering*, 199(49):3059–3100, 2010. ISSN 0045-7825. doi: <https://doi.org/10.1016/j.cma.2010.03.027>. URL <https://www.sciencedirect.com/science/>

article/pii/S0045782510001040.

- [81] Johannes Diderik van der Waals. ver de continuiteit van dengesen vloeistoftoestand (On the Continuity of the Gaseous and Liquid States). PhD thesis, Universiteit Leiden, 1873.
- [82] François Vilar. Cell-centered discontinuous galerkin discretization for two-dimensional lagrangian hydrodynamics. Computers & Fluids, 64:64–73, 2012. ISSN 0045-7930. doi: <https://doi.org/10.1016/j.compfluid.2012.05.001>. URL <https://www.sciencedirect.com/science/article/pii/S0045793012001685>.
- [83] François Vilar, Pierre-Henri Maire, and Rémi Abgrall. A discontinuous galerkin discretization for solving the two-dimensional gas dynamics equations written under total lagrangian formulation on general unstructured grids. Journal of Computational Physics, 276:188–234, 2014. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2014.07.030>. URL <https://www.sciencedirect.com/science/article/pii/S0021999114005233>.
- [84] François Vilar, Chi-Wang Shu, and Pierre-Henri Maire. Positivity-preserving cell-centered lagrangian schemes for multi-material compressible flows: From first-order to high-orders. part ii: The two-dimensional case. Journal of Computational Physics, 312:416–442, 2016. ISSN 0021-9991. doi: <https://doi.org/10.1016/j.jcp.2016.01.037>. URL <https://www.sciencedirect.com/science/article/pii/S0021999116000802>.
- [85] John von Neumann and Robert D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. Journal of Applied Physics, 21(3):232–237, 1950. ISSN 0021-8979. doi: 10.1063/1.1699639. URL <https://doi.org/10.1063/1.1699639>.
- [86] M L Wilkins. Calculation of elastic-plastic flow. Technical report, California. Univ., Livermore. Lawrence Radiation Lab., 04 1963. URL <https://www.osti.gov/biblio/4153725>.
- [87] T. Wu, M. Shashkov, N. Morgan, D. Kuzmin, and H. Luo. An updated lagrangian discontinuous galerkin hydrodynamic method for gas dynamics. Computers & Mathematics with Applications, 78(2):258–273, 2019. ISSN 0898-1221. doi: <https://doi.org/10.1016/j.camwa.2018.03.040>. URL <https://www.sciencedirect.com/science/>

article/pii/S0898122118301767. Proceedings of the Eight International Conference on Numerical Methods for Multi-Material Fluid Flows (MULTIMAT 2017).

APPENDIX A

FUNDAMENTAL DERIVATIONS AND INVARIANCE PROPERTIES

A.1 Proof of Reynold's Transport Theorem

Here we provide a proof of the Reynold's Transport Theorem, stated formally in Theorem 1.4.4.

Proof. Begin by rewriting the left hand side in terms of the initial configuration of the continuum of particles

$$\int_{D^t} \alpha(\mathbf{x}, t) dx = \int_{\tilde{D}} \alpha(\Phi(\xi, t), t) J \partial\xi.$$

We can now interchange the derivative and the integral in the fixed region \tilde{D} to obtain

$$\begin{aligned} \frac{d}{dt} \int_{D^t} \alpha(\mathbf{x}, t) dx &= \frac{d}{dt} \int_{\tilde{D}} \alpha(\Phi(\xi, t), t) J \partial\xi \\ &= \int_{\tilde{D}} \frac{d}{dt} \{\alpha(\Phi(\xi, t), t) J\} \partial\xi \\ &= \int_{\tilde{D}} \left\{ \frac{\partial}{\partial t} \alpha(\Phi(\xi, t), t) + \Phi_t \cdot \nabla \alpha \right\} J \partial\xi + \int_{\tilde{D}} \alpha(\Phi(\xi, t), t) \frac{d}{dt} J \partial\xi. \end{aligned}$$

We claim that

$$\frac{d}{dt} J = J \nabla \cdot \mathbf{v}(\mathbf{x}, t). \quad (\text{A.1.1})$$

We will show this is true through some matrix manipulation. We first simplify the material derivative of the entries of the Jacobian matrix

$$\frac{d}{dt} \frac{\partial x_i}{\partial \xi_j} = \frac{\partial}{\partial \xi_j} \frac{dx_i}{dt} = \frac{\partial v_i}{\partial \xi_j}.$$

Note that we can interchange the order of derivation since $\frac{d}{dt}$ holds ξ constant. Recall that \mathbf{v} is a

function of current particle position at time t , thus applying the chain rule we have

$$\frac{d}{dt} \frac{\partial x_i}{\partial \xi_j} = \frac{\partial v_i}{\partial \xi_j} = \sum_{k \in \{1:d\}} \frac{\partial v_i}{\partial x_k} \frac{\partial x_k}{\partial \xi_j}.$$

For a 2×2 Jacobian matrix of this type, we then have

$$\begin{aligned} \frac{d}{dt} (J) &= \frac{d}{dt} \left(\frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_2}{\partial \xi_1} \frac{\partial x_1}{\partial \xi_2} \right) \\ &= \frac{d}{dt} \left(\frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} + \frac{\partial x_1}{\partial \xi_1} \frac{d}{dt} \left(\frac{\partial x_2}{\partial \xi_2} \right) - \left[\frac{d}{dt} \left(\frac{\partial x_2}{\partial \xi_1} \right) \frac{\partial x_1}{\partial \xi_2} + \frac{\partial x_2}{\partial \xi_1} \frac{d}{dt} \left(\frac{\partial x_1}{\partial \xi_2} \right) \right] \right) \\ &= \left(\frac{\partial v_1}{\partial x_1} \frac{\partial x_1}{\partial \xi_1} + \cancel{\frac{\partial v_1}{\partial x_2} \frac{\partial x_2}{\partial \xi_1}} \right) \frac{\partial x_2}{\partial \xi_2} + \frac{\partial x_1}{\partial \xi_1} \left(\cancel{\frac{\partial v_2}{\partial x_1} \frac{\partial x_1}{\partial \xi_2}} + \frac{\partial v_2}{\partial x_2} \frac{\partial x_2}{\partial \xi_2} \right) \\ &\quad - \left[\left(\cancel{\frac{\partial v_2}{\partial x_1} \frac{\partial x_1}{\partial \xi_1}} + \frac{\partial v_2}{\partial x_2} \frac{\partial x_2}{\partial \xi_1} \right) \frac{\partial x_1}{\partial \xi_2} + \frac{\partial x_2}{\partial \xi_1} \left(\frac{\partial v_1}{\partial x_1} \frac{\partial x_1}{\partial \xi_2} + \cancel{\frac{\partial v_1}{\partial x_2} \frac{\partial x_2}{\partial \xi_1}} \right) \right] \\ &= \frac{\partial v_1}{\partial x_1} \frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_2}{\partial \xi_1} \frac{\partial v_1}{\partial x_1} \frac{\partial x_1}{\partial \xi_2} + \frac{\partial x_1}{\partial \xi_1} \frac{\partial v_2}{\partial x_2} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial v_2}{\partial x_2} \frac{\partial x_2}{\partial \xi_1} \frac{\partial x_1}{\partial \xi_2} \\ &= \frac{\partial v_1}{\partial x_1} \left(\frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_2}{\partial \xi_1} \frac{\partial x_1}{\partial \xi_2} \right) + \frac{\partial v_2}{\partial x_2} \left(\frac{\partial x_1}{\partial \xi_1} \frac{\partial x_2}{\partial \xi_2} - \frac{\partial x_2}{\partial \xi_1} \frac{\partial x_1}{\partial \xi_2} \right) \\ &= \left(\frac{\partial v_1}{\partial x_1} + \frac{\partial v_2}{\partial x_2} \right) J \\ &= (\nabla \cdot \mathbf{v}) J. \end{aligned}$$

This can of course be shown as well for a 3×3 matrix with additional bookkeeping. Hence

$$\begin{aligned} \frac{d}{dt} \int_{D^t} \alpha(\mathbf{x}, t) \partial \mathbf{x} &= \int_{\tilde{D}} \left\{ \frac{\partial \alpha}{\partial t} + \nabla \alpha \cdot \Phi_t + \alpha \nabla \cdot \Phi_t \right\} J \partial \boldsymbol{\xi} \\ &= \int_{\tilde{D}} \left\{ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \Phi_t) \right\} J \partial \boldsymbol{\xi}. \end{aligned}$$

Changing integration variables back to \mathbf{x} we then have

$$\frac{d}{dt} \int_{D^t} \alpha(\mathbf{x}, t) \partial \mathbf{x} = \int_{D^t} \left\{ \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha v) \right\} \partial \mathbf{x}.$$

□

In addition, it follows from (1.4.6) and (A.1.1) that

$$\frac{d}{dt}(\alpha J) = J \left[\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) \right]. \quad (\text{A.1.2})$$

This is a direct consequence of the chain rule

$$\begin{aligned} \frac{d}{dt}(\alpha J) &= J \frac{d\alpha}{dt} + \alpha \frac{dJ}{dt} \\ &= J \frac{\partial \alpha}{\partial t} + J(\nabla \alpha \cdot \mathbf{v}) + \alpha J \nabla \cdot \mathbf{v} \\ &= J \left(\frac{\partial \alpha}{\partial t} + \nabla \alpha \cdot \mathbf{v} + \alpha \nabla \cdot \mathbf{v} \right) \\ &= J \left(\frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}) \right). \end{aligned}$$

A.2 Alternate Derivation of the Euler Equations in a Lagrangian Frame

In this section we provide an alternate derivation of the Euler Equations in a moving Lagrangian frame to the derivation presented in 1.4.1. We follow in large part the approach developed in [64] and [17]. To begin, we plug in the conservative variables $\alpha = \rho$, 1 , $\rho \mathbf{v}$, ρE into equation (A.1.2), which yields

$$\begin{aligned} \frac{dJ}{dt} &= J \nabla \cdot \mathbf{v}, \\ \frac{d}{dt}(\rho J) &= J \left[\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) \right], \\ \frac{d}{dt}(\rho \mathbf{v} J) &= J \left[\frac{\partial}{\partial t}(\rho \mathbf{v}) + \nabla \cdot (\rho \mathbf{v} \otimes \mathbf{v}) \right], \\ \frac{d}{dt}(\rho E J) &= J \left[\frac{\partial}{\partial t}(\rho E) + \nabla \cdot (\rho E \mathbf{v}) \right]. \end{aligned}$$

After reorganizing the right hand side of the above equations using the Euler system (1.4.1), we then have

$$\frac{d}{dt} (\rho\tau J) - J\nabla \cdot \mathbf{v} = 0, \quad (\text{A.2.2a})$$

$$\frac{d}{dt} (\rho J) = 0, \quad (\text{A.2.2b})$$

$$\frac{d}{dt} (\rho J \mathbf{v}) + J\nabla \cdot \boldsymbol{\sigma} = \mathbf{0}, \quad (\text{A.2.2c})$$

$$\frac{d}{dt} (\rho JE) + J\nabla \cdot (\boldsymbol{\sigma} \mathbf{v}) = 0, \quad (\text{A.2.2d})$$

where $\tau = \frac{1}{\rho}$ is the specific volume. Notice that these equations are not fully in a Lagrangian formulation since the gradient and divergence operators are still in terms of the Eulerian coordinate system. Hence we call the system of equations in (A.2.2) to be in semi-lagrangian form.

In order to proceed with our spatial discretization of the system (A.2.2) we multiply by a test function ψ and integrate over the Lagrangian domain \tilde{D} . Knowing that $\Phi(\tilde{D}) = D^t$ and $Jd\tilde{D} = dD^t$, we have

$$\begin{aligned} \partial_t \int_{D^t} \tau(\mathbf{x}, t) \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} &= \int_{D^t} \nabla \cdot [\mathbf{v}(\mathbf{x}, t)] \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} \\ \partial_t \int_{D^t} \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} &= 0, \\ \partial_t \int_{D^t} \mathbf{v}(\mathbf{x}, t) \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} &= \int_{D^t} -\nabla \cdot [\boldsymbol{\sigma}(\mathbf{u}(\mathbf{x}, t))] \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} \\ \partial_t \int_{D^t} E(\mathbf{x}, t) \rho(\mathbf{x}, t) \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x} &= \int_{D^t} -\nabla \cdot [\boldsymbol{\sigma}(\mathbf{u}(\mathbf{x}, t)) \mathbf{v}(\mathbf{x}, t)] \psi(\Phi_t^{-1}(\mathbf{x})) d\mathbf{x}. \end{aligned}$$

A.3 Galilean Invariance

The principle of Galilean invariance requires that the equations of motion for an isolated system retain the same form under a change of reference frame corresponding to a uniform translation with constant velocity \mathbf{V}_G . In other words, the fundamental laws of physics must remain the same in all inertial frames, regardless of whether the system is observed from a stationary frame or one moving with constant velocity. This property is not merely a formal requirement: numerical schemes

that fail to preserve Galilean invariance can exhibit spurious dependence on the background flow, leading to errors such as incorrect shock speeds, artificial diffusion, or a breakdown of conservation properties. Ensuring Galilean invariance is therefore a key requirement for the reliability of computational models in mechanics and fluid dynamics.

Previous studies have highlighted the importance of Galilean invariance in Lagrangian flow computations. For example, the computations in [77] showed that non-invariant Streamline Upwind/Petrov–Galerkin (SUPG) formulations can produce spurious or unstable solution patterns. In the analysis presented in [78], Galilean transformations are cast as a matrix group acting on the vector flow equations. While these works provide valuable insights into the role of invariance, our approach follows that of [59], who derive the subcell force in their scheme by enforcing Galilean invariance and thermodynamic consistency. We likewise impose that our update remain Galilean invariant under a uniform translation with constant velocity. Begin by rewriting all terms of our low order update (3.1.5) on one side of the equation

$$(\mathbf{U}_c^{n+1} - \mathbf{U}_c^n) + \frac{\Delta t}{m_c} \left\{ \sum_{c' \in \mathcal{I}^*(c)} (\mathbb{f}(\mathbf{U}_{c'}^n) + \mathbb{f}(\mathbf{U}_c^n)) \mathbf{c}_{cc'} \right. \\ \left. + \sum_{F \in \mathcal{F}^\partial(K_c^n)} 2\mathbb{f}(\mathbf{U}_c^n) \mathbf{c}_{cF}^\partial - \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n (\mathbf{U}_{c'}^n - \mathbf{U}_c^n) \right\} = \mathbf{0}. \quad (\text{A.3.1})$$

Let $\mathbf{V}_G \in \mathbb{R}^d$ denote the uniform translation velocity. Then (A.3.1) becomes

$$\left[\mathbf{U}_c^{n+1} + \widehat{\mathbf{V}}_G - \left(\mathbf{U}_c^{n+1} + \widehat{\mathbf{V}}_G \right) \right] + \frac{\Delta t}{m_c} \left\{ \sum_{c' \in \mathcal{I}^*(c)} (\mathbb{f}(\mathbf{U}_{c'}^n) + \mathbb{V}_G + \mathbb{f}(\mathbf{U}_c^n) + \mathbb{V}_G) \mathbf{c}_{cc'} \right. \\ \left. + \sum_{F \in \mathcal{F}^\partial(K_c^n)} 2(\mathbb{f}(\mathbf{U}_c^n) + \mathbb{V}_G) \mathbf{c}_{cF}^\partial - \sum_{c' \in \mathcal{I}^*(c)} d_{cc'}^n \left[\mathbf{U}_{c'}^n + \widehat{\mathbf{V}}_G - \left(\mathbf{U}_c^n + \widehat{\mathbf{V}}_G \right) \right] \right\} = \mathbf{0}. \quad (\text{A.3.2})$$

$$\text{where } \mathbb{V}_G = (\mathbf{V}, \ 0, \ p\mathbf{V}_G)^\top \quad \text{and} \quad \widehat{\mathbf{V}}_G = \left(0, \ \mathbf{V}_G, \ \frac{1}{2} |\mathbf{V}_G|^2 \right)^\top \quad (\text{A.3.3})$$

By substituting (A.3.1) into this last equation we get

$$\frac{\Delta t}{m_c} \left(\sum_{c' \in \mathcal{I}^*(c)} 2\mathbb{V}_G \mathbf{c}_{cc'} + \sum_{F \in \mathcal{F}^\partial(K_c^n)} 2\mathbb{V}_G \mathbf{c}_{cF}^\partial \right) = \mathbf{0},$$

which must hold for all vectors \mathbf{V}_G . Therefore our update remains invariant under uniform translation if

$$\sum_{c' \in \mathcal{I}^*(c)} \mathbf{c}_{cc'} + \sum_{F \in \mathcal{F}^\partial(K_c^n)} \mathbf{c}_{cF}^\partial = \mathbf{0}. \quad (\text{A.3.4})$$

But this is true by definition by the divergence theorem

$$\sum_{c' \in \mathcal{I}^*(c)} \mathbf{c}_{cc'} + \sum_{F \in \mathcal{F}^\partial(K_c^n)} \mathbf{c}_{cF}^\partial = \sum_{F \in \mathcal{F}(K_c^n)} \frac{1}{2} \int_F \mathbf{n}_c ds = \frac{1}{2} \int_{\partial K_c^n} \mathbf{n}_c ds = \mathbf{0}.$$

APPENDIX B

ALTERNATIVE MESH VELOCITY APPROACHES

In the description of our robust low order method in Chapter 3 we outline that mass conservation imposed via a simple, global post-processing of the density/specific volume at the end of each time step (see §3.5). This appendix records several alternative strategies we evaluated—but did not adopt in the production workflow—including (i) a constrained optimization-based velocity redistribution under constraints representing local cell wise mass conservation and (ii) face-based mesh movement to correct for cell mass loss/gain. We include these methods for reproducibility and to document trade-offs (added solver/communication cost, parameter sensitivity, occasional degradation near shocks) relative to the post-processing baseline. Unless otherwise noted, comparisons in this appendix should be interpreted with the post-processing approach in §3.5 as the reference method.

B.1 Strategy 1: Optimization

One way to enforce mass conservation consists of moving the vertices of the mesh so that we have $|K_c^{n+1}| = \mathcal{T}_c^{n+1}m_c$ in every cell $c \in \mathcal{N}^{\text{Cel}}$ at the end of the time step. Let us denote $\mathbf{v}_h^{\text{vtx}}$ the (yet to be computed) mesh velocity based on the vertices and let $K_c(\mathbf{v}_h^{\text{vtx}})$ be the image of the cell K_c^n after being transported by flow $\mathbf{v}_h^{\text{vtx}}$. To ease the notation, we define $\mathbf{P}_h^{\text{vtx}}(\mathcal{T}_h^n)$ to be the \mathbb{R}^d -valued Lagrange finite space based on the vertices of the mesh \mathcal{T}_h^n , i.e., $\mathbf{v}_h^{\text{vtx}} \in \mathbf{P}_h^{\text{vtx}}(\mathcal{T}_h^n)$. The problem is then to find $\mathbf{v}_h^{\text{vtx}} \in \mathbf{P}_h^{\text{vtx}}(\mathcal{T}_h^n)$ so that $|K_c(\mathbf{v}_h^{\text{vtx}})| = \frac{\mathcal{T}_c^{n+1}}{\mathcal{T}_c^n} |K_c^n|$.

Let us estimate in two space dimensions the number of degrees of freedom and the number of constraints for this problem. The degrees of freedom are the two components of the mesh velocity at the vertices. Denoting $N_v := \text{card}(\mathcal{N}^{\text{vtx},n})$ the number of vertices, the number of degree of freedom is $2N_v$. Denoting $N_c = \text{card}(\mathcal{N}^{\text{Cel}})$ the number of cells, the number of mass conservation constraints is equal to N_c . Let N_v^∂ be the number of boundary vertices. Let I be the number of holes in D (first Betti number of D). Let $\nu = 4$ be the number of faces of the reference cell \widehat{K} .

Then

$$2N_v - N_v^\partial = 2(1 - I) + (\nu - 2)N_c. \quad (\text{B.1.1})$$

If one enforces the normal component of the mesh velocity at the boundary, then there are N_v^∂ less degrees of freedom, meaning the total number of degrees of freedom is $2N_v - N_v^\partial$ in this case.

For quadrangular meshes we have $\nu = 4$, and (B.1.1) shows that $2N_v - N_v^\partial \simeq 2N_c$, meaning that there are twice as many degrees of freedom as constraints. We regularize the problem by transforming it into a least square minimization problem with constraints. More precisely, let us consider a cell K_c^n with the four vertices $\mathbf{a}_{i_1}^n, \dots, \mathbf{a}_{i_4}^n$ where $i_1 := j^{\text{geo}}(1, c), \dots, i_4 := j^{\text{geo}}(4, c)$. The convention is that \mathbf{a}_{i_1} and \mathbf{a}_{i_3} are located at opposite corners of the cell. Let $\mathbf{V}_{i_1} := \mathbf{v}_h^{\text{vtx}}(\mathbf{a}_{i_1}^n), \dots, \mathbf{V}_{i_4} := \mathbf{v}_h^{\text{vtx}}(\mathbf{a}_{i_4}^n)$ be the velocities at these vertices. Then

$$|K_c(\mathbf{v}_h^{\text{vtx}})| = \frac{1}{2}(\Delta t(\mathbf{V}_{i_4} - \mathbf{V}_{i_2}) + \mathbf{a}_{i_4}^n - \mathbf{a}_{i_2}^n)^\perp \cdot (\Delta t(\mathbf{V}_{i_3} - \mathbf{V}_{i_1}) + \mathbf{a}_{i_3}^n - \mathbf{a}_{i_1}^n). \quad (\text{B.1.2})$$

Let $\bar{\mathbf{V}}^n$ be the solution of (3.4.10a) and let $\bar{\mathbf{v}}_h \in \mathbf{P}_h^{\text{vtx}}(\mathcal{T}_h^n)$ be the velocity constructed with these vertex values. The minimization problem that we consider is as follows:

$$\mathbf{v}_h^{\text{vtx}} = \arg \min_{\mathbf{v}_h \in \mathbf{P}_h^{\text{vtx}}(\mathcal{T}_h^n)} \sum_{i \in \mathcal{N}^{\text{vtx}, n}} m_i \|\mathbf{V}_i - \bar{\mathbf{V}}_i\|_{\ell^2}^2, \quad m_i = \frac{1}{\text{card}(\mathcal{T}(i))} \sum_{c \in \mathcal{T}(i)} m_c. \quad (\text{B.1.3a})$$

$$\text{subject to } |K_c(\mathbf{v}_h)| = \frac{\mathsf{T}_c^{n+1}}{\mathsf{T}_c^n} |K_c^n|, \forall c \in \mathcal{N}^{\text{Cel}}. \quad (\text{B.1.3b})$$

We solve this constrained optimization problem with a Lagrange multiplier method using the HiOp Optimization solver [72] developed at LLNL.

The spirit of the method introduced above is similar to a nonlinear optimization strategy proposed in [24] to maintain local mass conservation. The mesh motion is obtained therein by solving the implicit hydrodynamics update with the entropy being kept constant from t^n to t^{n+1} , see [24, §4]. The total specific energy is updated at the end of the iterative process. The main difference between the two approaches is that the hydrodynamics is implicit in [24, §4] whereas it is explicit here.

Remark B.1.1 (Optimization on triangular meshes). *For triangular grids ($\nu = 3$), the Euler relation (B.1.1) implies that the number of mass–conservation constraints matches the number of vertex–velocity degrees of freedom. The resulting constrained problem is a very stiff nonlinear optimization that frequently stalls/locks in practice and is not computationally efficient. [See 45, § 6.1.2].*

B.2 Strategy 2: Face Correction

Recall that exact mass conservation $\frac{|K_c^{n+1}|}{\mathsf{T}_c^{n+1}} = \frac{|K_c^n|}{\mathsf{T}_c^n} = \dots = \frac{|K_c^0|}{\mathsf{T}_c^0} =: \mathbf{m}_c$ is equivalent to $\mathbf{m}_c(\mathsf{T}_c^{n+1} - \mathsf{T}_c^n) = |K_c^{n+1}| - |K_c^n|$. But the time variation of the measure of the cell is fully determined by the mesh motion as stated in the following classical result (see e.g., [43, Lem. 3.3], [38, Prop. 3.1], [57, §4.3]), which is consequence of Reynolds’ theorem.

Lemma 3. *Let $(\omega_l, \zeta_l)_{l \in \mathcal{L}}$ be an integration quadrature rule that is exact for all polynomials of degree at most $d - 1$, i.e., $\int_0^1 q(\zeta) d\zeta = \sum_{l \in \mathcal{L}} \omega_l q(\zeta_l)$ for all $q \in \mathbb{P}_{d-1}$. Let $c \in \mathcal{N}^{\text{Cel}}$. Let $\{F_{c,m}^{n+\zeta}\}_{m \in \widehat{\mathcal{N}}^{\text{Fac}}}$ be the faces of the cell K_c at time $t^n + \zeta \Delta t$ (i.e., $F_{c,m}^{n+\zeta} := \Phi_h^n(F_{c,m}^n, t^n + \zeta \Delta t)$). Then*

$$|K_c^{n+1}| - |K_c^n| = \Delta t \sum_{m \in \widehat{\mathcal{N}}^{\text{Fac}}} \sum_{l \in \mathcal{L}} \omega_l \int_{F_{c,m}^{n+\zeta_l}} \mathbf{v}_h(\mathbf{x}, t^n + \zeta_l \Delta t) \cdot \mathbf{n}_c ds. \quad (\text{B.2.1})$$

Hence, to have exact mass conservation, the identity (B.2.1) requires that the update T_c^{n+1} satisfy the following equality:

$$\mathbf{m}_c \frac{\mathsf{T}_c^{n+1} - \mathsf{T}_c^n}{\Delta t} = \sum_{m \in \widehat{\mathcal{N}}^{\text{Fac}}} \sum_{l \in \mathcal{L}} \omega_l \int_{F_{c,m}^{n+\zeta_l}} \mathbf{v}_h(\mathbf{x}, t^n + \zeta_l \Delta t) \cdot \mathbf{n}_c ds. \quad (\text{B.2.2})$$

In view of (3.4.6), this is achieved by making sure that \mathbf{v}_h^n is defined so that the following identity holds:

$$\sum_{m \in \widehat{\mathcal{N}}^{\text{Fac}}} \sum_{l \in \mathcal{L}} \omega_l \int_{F_{c,m}^{n+\zeta_l}} \mathbf{v}_h(\mathbf{x}, t^n + \zeta_l \Delta t) \cdot \mathbf{n}_c ds = \sum_{m \in \widehat{\mathcal{N}}^{\text{Fac}}} \mathbf{V}_{F_{c,m}^n} \cdot \mathbf{n}_{F_{c,m}^n} |F_{c,m}^n|. \quad (\text{B.2.3})$$

Since computing the vector field \mathbf{v}_h so that the above identity (B.2.3) holds has an algorithmic

complexity similar to the optimization problem presented in §B.1, we propose instead to just enforce the sufficient condition

$$\sum_{l \in \mathcal{L}} \omega_l \int_{F_{c,m}^{n+\zeta_l}} \mathbf{v}_h(\mathbf{x}, t^n + \zeta_l \Delta t) \cdot \mathbf{n}_c \, ds = \mathbf{V}_{F_{c,m}^n} \cdot \mathbf{n}_{F_{m,c}^n} |F_{c,m}^n|, \quad (\text{B.2.4})$$

which we know is the correct answer in one-space dimension, see (3.4.9) (with $\zeta_1 = 0$). Solving (B.2.4) can be done on each face independently if one allows only one degree of freedom to vary per face and the degree of freedom is in the middle of the face.

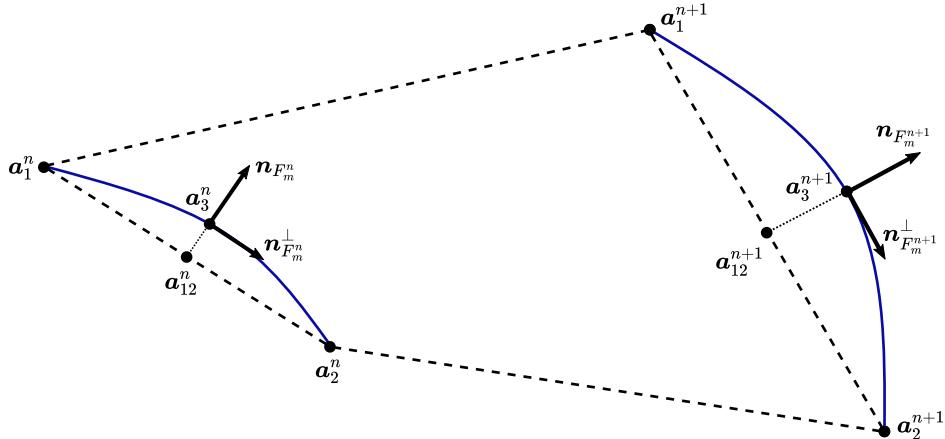


Figure B.1: Motion of the face which preserves mass.

The strategy we propose consists of using a Q_2 approximation for the geometric finite element. We first define \mathbf{v}_h^n at the vertices by solving (3.4.10). Then the normal component of \mathbf{v}_h^n at the middle of the faces is obtained by solving (B.2.4), as is depicted in Figure B.1. In two space dimensions, one can choose the midpoint rule for the time quadrature. Hence, there is only one quadrature point $\zeta_1 = \frac{1}{2}$. Exact analytical expressions for the solution of (B.2.4) can be easily derived. In addition, we want to prevent the mesh from further degeneration, so we seek to modify the perpendicular velocity so that the new node a_3^{n+1} lies directly above $a_{12}^{n+1} := \frac{1}{2}(a_1^{n+1} + a_2^{n+1})$. We do not report these expressions here, as we now consider this venue a dead end. We nevertheless

illustrate the method in the numerical section of the paper.

B.3 Comparison with Post-Processing

Here we compare the results of using these alternative mesh velocity descriptions on the distorted sod shocktube problem, described in §6.2.2. Table B.1 reports errors, observed orders, and CPU time per step and per degree of freedom. The alternative mesh–velocity formulations—most notably the optimization-based variant—are computationally inefficient: the per-step, per-DoF cost grows with refinement, revealing poor scalability. In several runs (especially under stronger distortions), these alternatives also stalled and occasionally led to early mesh tangling/collapse. Although the error levels are comparable, we therefore discard the alternatives and retain the post-processing approach as the only robust and efficient option.

Table B.1: Convergence tests for the distorted Sod shocktube on quadrilateral meshes. Optimization-based method vs. post-processing vs. face correction.

# dof	Optimization			Post-processing			Face Correction		
	$\delta^1(0.225)$	order	time/dof/ts	$\delta^1(0.225)$	order	time/dof/ts	$\delta^1(0.225)$	order	time/dof/ts
500	5.613E−2	–	3.316E−4	5.985E−2	–	4.093E−5	5.981E−2	–	5.290E−5
2000	3.567E−2	0.65	9.536E−4	3.786E−2	0.66	4.000E−5	3.786E−2	0.66	4.684E−5
8000	2.266E−2	0.65	4.069E−3	2.418E−2	0.65	4.026E−5	2.418E−2	0.65	5.250E−5
32000	1.384E−2	0.71	7.263E−3	1.470E−2	0.72	4.066E−5	1.471E−2	0.72	4.783E−5

B.4 Clear Mesh Distortion

We now illustrate the mass correction method described in §B.2. Recall that this method consists of using (serendipity) Q_2 polynomials for the mesh velocity \mathbf{v}_h^n . The mesh velocity at the vertices is computed by solving the 2×2 linear system (3.4.10) at each vertex, and the normal component of the velocity at the middle of the faces is computed by solving the scalar equation (B.2.4) on each face. The method is exactly mass conservative in the sense of Definition 8, but as explained in §B.2, the identity (B.2.4) is just a sufficient condition for local mass conser-

vation and unfortunately leads to mesh deformations that induce premature collapse of the mesh in some cases. We show in Figure B.2 samples of meshes obtained with this method to solve the Sod problem (left panel) and the isentropic vortex problem (right panel). Although the method is exactly mass conservative, invariant-domain-preserving, and is the right method in 1D, we do not recommend using it in two and higher space dimensions.

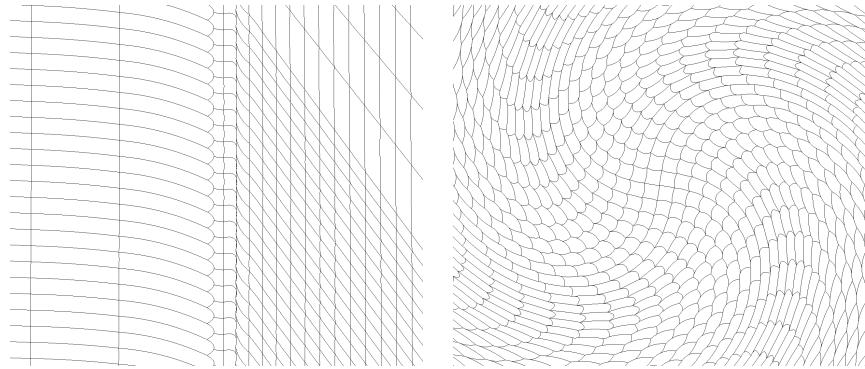


Figure B.2: Meshes obtained with the mass conservative method using strategy 2 consisting of correcting mass defects at cell interfaces. Left: distorted Sod shocktube. Right: isentropic vortex.

APPENDIX C

SELECTED ALGORITHMS

C.1 IDP-ERK Time Stepping

In this section we provide the algorithm which we use to step forward in time. The details are discussed in Section 4.3. This algorithm is reprinted with permission from [33].

Algorithm 1 s -stage IDP–ERK scheme

Require: $U^n \in \mathcal{B}^I$

- 1: Set $\mathbf{U}^{n,1} := \mathbf{U}^n$
 - 2: Set $\mathbf{U}^{n,l'} := \mathbf{U}^n$
 - 3: **for** $l = 2, \dots, s+1$ **do**
 - 4: $\mathbb{M}^L \mathbf{U}^{L,l} := \mathbb{M}^L \mathbf{U}^{n,l'} + \tau(c_l - c_{l'}) \mathbf{F}^L(\mathbf{U}^{n,l'})$ ▷ Low-order update (3.1.5)
 - 5: $\mathbb{M}^H \mathbf{U}^{H,l} := \mathbb{M}^H \mathbf{U}^{n,l'} + \tau \sum_{k=1}^{l-1} (a_{l,k} - a_{l',k}) \mathbf{F}^H(\mathbf{U}^{n,k})$ ▷ High-order update
 - 6: $\mathbf{U}^{n,l} := \ell(\mathbf{U}^{n,l'}, \Phi^{L,l}, \Phi^{H,l})$ ▷ with fluxes defined in (4.3.2)
 - 7: **end for**
 - 8: Set $\mathbf{U}^{n+1} := \mathbf{U}^{n,s+1}$

C.2 Limiter

In this section, we provide the algorithm which we use to limit the invariant-domain violating results from the high-order approximation. The details are discussed in Section 4.4.4. This algorithm is reprinted with permission from [42].

Algorithm 2 Local Limiting Algorithm

Require: Bounds $\{x_i^{\max}, x_i^{\min}\}_{i \in \mathcal{V}}$, $\{x_i\}_{i \in \mathcal{V}}$

```
1: for  $i \in \mathcal{V}$  do
2:    $\triangleright$  Loop over dofs
3:   if  $m_i = 0$  then
4:      $y_i = \min(x_i^{\max}, \max(x_i^{\min}, x_i))$ 
5:   else if  $x_i^{\max} < x_i$  then
6:      $\triangleright$  Local maximum principle violated
7:     Compute  $a_i^+, b_i^+, \ell_i^+$  using (2.2)
8:      $y_i = b_i^+$ 
9:   for  $j \in \mathcal{I}_i^+$  do
10:     $y_j = x_j + \ell_i^+ \max(0, x_j^{\max} - x_j)$ 
11:  end for
12:  else if  $x_i < x_i^{\min}$  then
13:     $\triangleright$  Local minimum principle violated
14:    Compute  $a_i^-, b_i^-, \ell_i^-$  using (2.4)
15:     $y_i = b_i^-$ 
16:    for  $j \in \mathcal{I}_i^-$  do
17:       $y_j = x_j + \ell_i^- \max(0, x_j - x_j^{\min})$ 
18:    end for
19:  else
20:     $\triangleright$  Do nothing
21:     $y_i = x_i$ 
22:  end if
23: end for
24: return  $\{y_i\}_{i \in \mathcal{V}}$ 
```

C.3 Relaxation

In this section, we provide the algorithm which we use to relax the local bounds given by the low order method that are then used to limit the high order values. The details are discussed in Section 4.4.3. This algorithm is reprinted with permission from [42].

Algorithm 3 RelaxBounds

Require: Bounds $\{u_i^{\max}, u_i^{\min}\}_{i \in \mathcal{V}}$, $\{u_i\}_{i \in \mathcal{V}}$, FEM basis $\{\varphi_i\}_{i \in \mathcal{V}}$, u^{\min} , u^{\max}

```

1:  $\beta_{ij} := \int_D \nabla \varphi_i \cdot \nabla \varphi_j d\mathbf{x}$                                 ▷ Set stiffness matrix
2:  $\alpha_i := \frac{\sum_{j \in \mathcal{I}(i)^*} \beta_{ij} (u_i - u_j)}{\sum_{j \in \mathcal{I}(i)^*} \beta_{ij} d_j}$           ▷ Estimate second-order increment
3: for  $i \in \mathcal{V}$  do
4:    $\beta_i \leftarrow \alpha_i$                                          ▷ Initialize relaxation parameter
5:   for  $j \in I(i)^*$  do
6:     if  $\beta_i \alpha_j \leq 0$  then
7:        $\beta_i \leftarrow 0$ 
8:     else if  $|\beta_i| > |\alpha_j|$  then
9:        $\beta_i \leftarrow \alpha_j$ 
10:    end if
11:   end for
12: end for
13: for  $i \in \mathcal{V}$  do
14:    $u_i^{\max} \leftarrow \min(u_i^{\max} + |\beta_i|, u^{\max})$           ▷ Relax upper bound
15:    $u_i^{\min} \leftarrow \max(u_i^{\min} - |\beta_i|, u^{\min})$           ▷ Relax lower bound
16: end for
17: return  $\{u_i^{\max}, u_i^{\min}\}_{i \in \mathcal{V}}$ 

```

C.4 Single Stage of Limited Laghos

In this section, we provide the algorithm which we use at each stage of the IDP ERK time stepping to limit the high order solution provided by Laghos. The details are discussed in Section 4.5.

Algorithm 4 Single State High Order IDP Update

Require: $\mathbf{U}^{n,l'} \in \mathcal{B}^I$

- 1: $\mathbf{U}^{L,n,l'} \leftarrow \text{InterpolationGridTransfer}(\mathbf{U}^{n,l'})$ ▷ See §4.5.2.
 - 2: $\mathbb{M}^L \mathbf{U}^{L,n,l} := \mathbb{M}^L \mathbf{U}^{L,n,l'} + \tau(c_l - c_{l'}) \mathbf{F}^L(\mathbf{U}^{L,n,l'})$ ▷ Low-order update (3.1.5)
 - 3: $\mathbb{M}^H \mathbf{U}^{H,n,l} := \mathbb{M}^H \mathbf{U}^{n,l'} + \tau \sum_{k=1}^{l-1} (a_{l,k} - a_{l',k}) \mathbf{F}^H(\mathbf{U}^{n,k})$ ▷ High-order update
 - 4: $\{\rho_i^{H,n,l}\}_{i \in \mathcal{V}} \leftarrow \text{ComputeDensity}(\mathbf{U}^{H,n,l})$ ▷ See §4.5.4.
 - 5: $\{\rho_i^{\min}, \rho_i^{\max}\}_{i \in \mathcal{V}} \leftarrow \text{ComputeBounds}(\mathbf{U}^{L,n,l})$
 - 6: $\{\widehat{\rho}_i^{\min}, \widehat{\rho}_i^{\max}\}_{i \in \mathcal{V}} \leftarrow \text{RelaxBounds}\left(\{\rho_i^{\min}, \rho_i^{\max}\}_{i \in \mathcal{V}}\right)$ ▷ Algorithm 3
 - 7: $\{\rho_i^{H,n,l}\}_{i \in \mathcal{V}} \leftarrow \text{LimitDensity}\left(\{\widehat{\rho}_i^{\min}, \widehat{\rho}_i^{\max}\}_{i \in \mathcal{V}}, \{\rho_i^{H,n,l}\}_{i \in \mathcal{V}}\right)$ ▷ Algorithm 2
 - 8: $M_{\mathcal{E}}, M_{\mathcal{V}} \leftarrow \text{UpdateMassMatrices}\left(\{\rho_i^{H,n,l}\}_{i \in \mathcal{V}}\right)$ ▷ See §4.5.5.
 - 9: **return** $\mathbf{U}^{n,l} \in \mathcal{B}^I$
-

C.5 Nonlinear Hyperelasticity

In this section, we provide the algorithm which we use to update our state variables in the nonlinear hyperelasticity setting. The details are discussed in Section 5.4.

Algorithm 5 Update State with Elasticity

Require: $\{\mathbf{U}_i^n\}_{i \in \mathcal{V}}$

- 1: **for all** $i \in \mathcal{V}$ **do**
 - 2: Compute $\boldsymbol{\sigma}$ at t^n
 - 3: $\mathbf{U}_i^{n+1} := \mathbf{U}_i^n + \frac{\Delta t}{m_i^\rho} \mathbf{F}^L(\mathbf{U}^n)$
 - 4: Compute and move mesh
 - 5: Post-process ρ_i^{n+1}
 - 6: $\mathbb{J}_i^{n+1} = \sum_{q=1}^{N_q} \omega_{i,q} \frac{\partial \Phi_{i,q}^{n+1}}{\partial \xi}$
 - 7: $\mathbf{C}_i^{n+1} = (\mathbb{J}_i^{n+1})^T \mathbb{J}_i^{n+1}, c = \frac{C}{|\mathbf{C}|^{1/3}}$
 - 8: Update $e_i^{s,n+1}(\dots)$ based on EOS
 - 9: **end for**
-

APPENDIX D

ELASTIC DEFINITIONS AND DERIVATIONS

Our goal in this appendix is to show that equations 5.3.7 and 5.3.7 are equivalent when the shear energy is isotropic.

In all of the following, let $\mathbb{A} \in \mathbb{R}^{d \times d}$ be a matrix with entries A_{ij} for $i, j \in \{1 : d\}$. Let $|\cdot| : \mathbb{R}^{d \times d} \rightarrow \mathbb{R}$ represent the determinant of the given matrix.

D.1 Some Linear Algebra

Definition 10 (Adjugate (classical adjoint)). *For $1 \leq i, j \leq d$, let \mathbb{M}_{ij} be the $(d - 1) \times (d - 1)$ minor obtained by deleting row i and column j of \mathbb{A} , and define the cofactor matrix \mathbb{C} with entries*

$$\mathbb{C}_{ij} := (-1)^{i+j} |\mathbb{M}_{ij}|.$$

The adjugate (or classical adjoint) of \mathbb{A} is the matrix

$$\text{adj}(\mathbb{A}) := \mathbb{C}^T,$$

i.e., the transpose of the cofactor matrix of \mathbb{A} .

Fundamental identity.

$$\mathbb{A} \text{adj}(\mathbb{A}) = \text{adj}(\mathbb{A}) \mathbb{A} = |\mathbb{A}| \mathbb{I}_d.$$

In particular, if $|\mathbb{A}| \neq 0$,

$$\mathbb{A}^{-1} = \frac{\text{adj}(\mathbb{A})}{|\mathbb{A}|}.$$

Useful properties. For $\mathbb{A}, \mathbb{B} \in \mathbb{R}^{d \times d}$ and scalar $c \in R$:

$$\text{adj}(AB) = \text{adj}(\mathbb{B}) \text{adj}(\mathbb{A}), \quad \text{adj}(\mathbb{A}^T) = \text{adj}(\mathbb{A})^T, \quad \text{adj}(cA) = c^{n-1} \text{adj}(\mathbb{A}).$$

Symmetric matrix. If $\mathbb{C} \in \mathbb{R}^{d \times d}$ is symmetric, then

$$\frac{\text{adj}(\mathbb{C})\mathbb{C}}{|\mathbb{C}|} = \mathbb{I}_d. \quad (\text{D.1.1})$$

D.2 Some Calculus

Definition 11 (Gradient). *Let $f : \mathbb{R}^{d \times d} \rightarrow \mathbb{R}$. Then the gradient of f with respect to \mathbb{A} is defined*

$$\nabla_{\mathbb{A}} f(\mathbb{A}) = \frac{\partial f(\mathbb{A})}{\partial \mathbb{A}} = \begin{bmatrix} \frac{\partial f}{\partial A_{11}} & \frac{\partial f}{\partial A_{12}} & \cdots & \frac{\partial f}{\partial A_{1d}} \\ \frac{\partial f}{\partial A_{21}} & \frac{\partial f}{\partial A_{22}} & \cdots & \frac{\partial f}{\partial A_{2d}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f}{\partial A_{d1}} & \frac{\partial f}{\partial A_{d2}} & \cdots & \frac{\partial f}{\partial A_{dd}} \end{bmatrix} \quad (\text{D.2.1})$$

Various useful derivatives. For $\mathbb{A} \in \mathbb{R}^{d \times d}$:

$$\frac{\partial}{\partial \mathbb{A}} \text{Tr}(\mathbb{A}) = \mathbb{I}_d, \quad (\text{D.2.2})$$

$$\frac{\partial}{\partial \mathbb{A}} \text{Tr}(\mathbb{A}^2) = 2\mathbb{C}, \quad (\text{D.2.3})$$

$$\frac{\partial}{\partial \mathbb{A}} |\mathbb{A}| = \text{adj}(\mathbb{A}^T). \quad (\text{D.2.4})$$

D.3 Cauchy Green Strain Tensor (Isotropic Case)

Putting the above definitions and relationships together, we are ready to show the equivalence of (5.3.7) and (5.3.7). To show this we must only evaluate the term representing the shear response. To begin, we evaluate the derivative of the invariants of the reduced Cauchy-Green strain tensor

(5.3.5) with respect to the right Cauchy-Green strain tensor

$$\begin{aligned}
\frac{\partial j_1}{\partial \mathbf{C}} &= \frac{\partial}{\partial \mathbf{C}} (\text{Tr}(\mathbf{c})) \\
&= \frac{\partial}{\partial \mathbf{C}} [|\mathbf{C}|^{-1/3} \text{Tr}(\mathbf{C})] \\
&= |\mathbf{C}|^{-1/3} \mathbb{I}_d - \frac{1}{3} |\mathbf{C}|^{-4/3} \text{Tr}(\mathbf{C}) \text{adj}(\mathbf{C}^T) \\
&= |\mathbf{C}|^{-1/3} \mathbb{I}_d - \frac{1}{3} |\mathbf{C}|^{-1/3} \text{Tr}(\mathbf{C}) \mathbf{C}^{-1} \\
&= |\mathbf{C}|^{-1/3} \mathbb{I}_d - \frac{1}{3} j_1 \mathbf{C}^{-1} \\
\frac{\partial j_2}{\partial \mathbf{C}} &= \frac{\partial}{\partial \mathbf{C}} (\text{Tr}(\mathbf{c}^2)) \\
&= \frac{\partial}{\partial \mathbf{C}} [|\mathbf{C}|^{-2/3} \text{Tr}(\mathbf{C}^2)] \\
&= 2 |\mathbf{C}|^{-2/3} \mathbf{C} - \frac{2}{3} |\mathbf{C}|^{-5/3} \text{Tr}(\mathbf{C}^2) \text{adj}(\mathbf{C}^T) \\
&= 2 |\mathbf{C}|^{-2/3} \mathbf{C} - \frac{2}{3} |\mathbf{C}|^{-2/3} \text{Tr}(\mathbf{C}^2) \mathbf{C}^{-1} \\
&= 2 |\mathbf{C}|^{-2/3} \mathbf{C} - \frac{2}{3} j_2 \mathbf{C}^{-1}
\end{aligned}$$

Then, by the chain rule we have

$$\begin{aligned}
\mathbb{J} \frac{\partial e^s}{\partial \mathbf{C}} \mathbb{J}^\top &= \mathbb{J} \left[\frac{\partial e^s}{\partial j_1} \frac{\partial j_1}{\partial \mathbf{C}} + \frac{\partial e^s}{\partial j_2} \frac{\partial j_2}{\partial \mathbf{C}} \right] \mathbb{J}^\top \\
&= \mathbb{J} \frac{\partial e^s}{\partial j_1} \left(|\mathbf{C}|^{-1/3} \mathbb{I}_d - \frac{1}{3} j_1 \mathbf{C}^{-1} \right) \mathbb{J}^\top + \mathbb{J} \frac{\partial e^s}{\partial j_2} \left(2 |\mathbf{C}|^{-2/3} \mathbf{C} - \frac{2}{3} j_2 \mathbf{C}^{-1} \right) \mathbb{J}^\top \\
&= \frac{\partial e^s}{\partial j_1} \left(\mathbf{b} - \frac{1}{3} j_1 \mathbb{I}_d \right) + \frac{\partial e^s}{\partial j_2} \left(2 \mathbf{b}^2 - \frac{2}{3} j_2 \mathbb{I}_d \right)
\end{aligned}$$

Substituting this equation into (5.3.7) yields (5.3.8).