Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering
Department of Physical Electronics

Dissertation thesis

Lagrangian methods for continuum dynamics

Praha 2020

Ing. David Fridrich

Bibliografický záznam

Autor:	Ing. David Fridrich České vysoké učení technické v Praze Fakulta jaderná a fyzikálně inženýrská Katedra fyzikální elektroniky
Název práce:	Lagrangeovské metody pro dynamiku kontinua
Studijní program:	Aplikace přírodních věd
Studijní obor:	Fyzikální inženýrství
Školitel:	prof. Ing. Richard Liska, CSc. České vysoké učení technické v Praze Fakulta jaderná a fyzikálně inženýrská Katedra fyzikální elektroniky
Akademický rok:	2019-2020
Počet stran:	137
Klíčová slova:	Lagrangeovská hydrodynamika, elasto-plasticita, Lax–Wendroff, umělá vazkost, zachování symetrie.

Bibliographic Entry

Author:	Ing. David Fridrich Czech Technical University in Prague Faculty of Nuclear Sciences and Physical Engineering Department of Physical Electronics
Title of Dissertation:	Lagrangian methods for continuum dynamics
Degree Programme:	Applications of Natural Sciences
Field of Study:	Physical Engineering
Supervisor:	prof. Ing. Richard Liska, CSc. Czech Technical University in Prague Faculty of Nuclear Sciences and Physical Engineering Department of Physical Electronics
Academic Year:	2019-2020
Number of Pages:	137
Keywords:	Lagrangian hydrodynamics, Elasto-plastic flow, Lax–Wendroff, Artificial viscosity, Symmetry preservation.

Abstrakt

Hlavním cílem této práce bylo vyvinout novou cell-centered numerickou metodu pro hydrodynamiku v 1D, 2D kartézské a 2D rz cylindrické geometrii a pro problémy elastoplasticity v 1D a 2D kartézské geometrii. Hydrodynamika je popsána soustavou zákonů zachování – Eulerových rovnic v Lagrangeovských souřadnicích. Systém je uzavřen stavovou rovnicí pro ideální plyn. Elastoplasticita je popsána Wilkinsovým modelem obsahujícím podobný systém zákonů zachování doplněný rovnicí pro výpočet tensoru napětí založenou na Hookeově zákoně a Mie-Grünneisenovou stavovou rovnicí. Zmíněné soustavy parciálních difirenciálních rovnic jsou numericky řešeny pomocí hybridní cell-centered metody konečných objemů. Fyzikální toky jsou diskretizovány pomocí Richtmyerovy dvoukrokové formulace Laxova-Wenroffova schematu typu prediktor-korektor. Původní Laxův-Friedrichsův prediktor je vylepšen Wendroffovým-Weightovým vážením [B. Wendroff, A. B. White, Comput. Math. with Appl. **18**:761, (1989)], které významně zlepšuje chování metody na nerovnoměrných výpočetních sítích. K potlačení nefyzikálních oscilací způsobených disperzivitou LW schématu je metoda doplněna umělou disipativitou založenou na HLL schematu – do rovnice pro hybnost je přidána umělá vazkost a do rovnice pro energii umělý tok energie. Rozlišení kontaktní nespojitosti či rozhraní různých materiálů je vylepšeno pomocí speciálního algoritmu, znaného *interface fix*. Tenzor napětí je počítán metodou inspirovanou [P. H. Maire et al., J. Comput. Phys. **235**:626,(2013)], zatímco plasticita je započítána pomocí Wilkinsova radial-return algoritmu. Numerické vlastnosti metody jsou ilustrovány na několika analýzách řádu konvergence a typických testovacích úlohách. Uvedená metoda splňuje geometrický zákon zachování a na polární síti zachovává přesnou symetrii pro hydrodynamiku v cylindrických rz souřadnicích. Symetrie na pravoúhlých sítích zůstává velmi dobrá.

Abstract

The main goal of this thesis was to develop a novel cell-centered numerical method for calculating 1D, 2D Cartesian, and 2D axisymmetric cylindrical hydrodynamics and 1D and 2D Cartesian elastic-plastic flows in Lagrangian coordinates. The hydrodynamics is described by a system of conservation laws – Euler equations in Lagrangian coordinates closed by an ideal gas equation of state. For the description of elastic-plastic flow, Wilkins model is used incorporating a similar set of conservation laws supplemented by incremental constitutive law based on Hooke's law for evolution of elastic deviatoric stress, Mie-Grüneisen equation of state, and plasticity due to von Mises yield condition. The aforesaid systems of partial differential equations are numerically treated by a hybrid explicit cell-centered finite volume method. The physical fluxes are discretized using the Richtmyer two-step predictor-corrector formulation of the Lax-Wendroff scheme. The original Lax-Friedrichs predictor is improved by inverse area weighting [B. Wendroff and A. B. White, Comput. Math. with Appl. **18**:761, (1989)], which significantly improves the performance of the method on nonuniform meshes. To mitigate nonphysical oscillations due to LW dispersivity, HLL based artificial dissipation is added to momentum and energy equations, and a limiting procedure for this artificial dissipation is also proposed. Contact discontinuity capturing and material interfaces treatment are enhanced by a simple *interface fix* procedure. The discretization of incremental constitutive law is inspired by the method due to [P. H. Maire et al., J. Comput. Phys. **235**:626,(2013)], while the plasticity is calculated using the Wilkins radial return algorithm. The numerical performance of the method is illustrated on several convergence rate analyses and typical benchmark tests. The presented method satisfies Geometric Conservation Law and preserves the exact symmetry in axisymmetric rz hydrodynamics on polar meshes while the symmetry on initially rectangular meshes remains very good.

Acknowledgements

This project was supported by the Czech Science Foundation projects 14-21318S, 18-20962S, Czech Ministry of Education project RVO 68407700, Czech Technical University in Prague projects SGS16/247/OHK4/3T/14, SGS19/191/OHK4/3T/14 and the project CZ.02.1.01/0.0/0.0/16_019/0000778 from European Regional Development Fund.

Firstly, I would like to thank to my supervisor Richard Liska for very helpful and patient leading, cooperation, and innumerable discussions, ideas, remarks, and clarifications.

Many thanks belong to Burton Wendroff for cooperation on articles summarized in this thesis, Pavel Váchal and Milan Kuchařík for their valuable insights, numerous discussions, and answering all my questions ranging through the most aspects of this work, Matěj Klíma for introductory lectures and comments on elastoplasticity, Raphaël Loubère for his help and remarks regarding hydrodynamics, and François Vilar and Pierre-Henri Maire for providing 1D smooth problem.

I am also grateful to my chiefs at IPP CAS, namely Jakub Urban, Martin Hron, Michael Komm, Radomír Pánek, and David Tskhakaya, for always being helpful and supportive.

Last but not least, I have to thank to my family: parents Irena and Slavomír, grandmother Malvína and brother Tomáš, without whose support and understanding this work would not have been possible.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

This work deals with simulations of high-speed continuum flows described by the systems of conservation laws, namely hydrodynamics studying the behavior of inviscid compressible fluid flows, and Wilkins model [1] addressing rapid deformations of elastic-plastic solids.

Based on the used reference frame, the problems of continuum mechanics can be formulated using two different frameworks. One option is the Eulerian description, where the fluid is observed using a fixed reference frame, which means that the computational mesh does not change in the time, and the fluid flow is expressed in terms of mass fluxes through the edges of computational cells. This approach works well for a considerable majority of engineering applications, such as Computational Fluid Dynamics (CFD), where the flows usually do not undergo rapid changes.

The second option is a Lagrangian description, where the computational mesh moves with the flow velocity, and therefore there is no mass flux through the cell edges. The main advantages of this method are the ability to resolve material interfaces sharply and dealing with the changing computational domain and moving boundary conditions. However, besides conserving variables, one has to compute also the vertex velocities with which the mesh moves. The additional complexity due to the discretization of the nodal motion is usually redeemed by a smaller number of computational cells. Thus, the Lagrangian methods are better suited for research fields dealing with high-speed phenomena, such as astrophysics, Inertial Confinement Fusion (ICF), or high-velocity impacts of solids.

The Lagrangian methods can be further divided into two groups depending on the computational quantities organization and the way of how the vertex velocities are computed. The most straightforward way of how to obtain vertex velocities is to use staggered discretization, where position, velocity, and kinetic energy are computed in cell vertices, while the thermodynamical variables, like density, pressure, sound speed, and internal energy are computed in cell centers. These methods are referred to as *Staggered methods* and were historically the first developed.

The first Lagrangian method for 1D hydrodynamics was published in 1950 by von Neumann and Richtmyer [2], and used staggered discretization. Later it was extended for 2D elastic-plastic flows in [3]. The original version of the scheme did not conserve the total energy and suffered from spurious numerical oscillations.

Over the decades, these staggered methods have been continuously improved in many different directions, overviewed in great detail by Benson in [4]. More recently, Caramana and Shashkov suggested a new subzonal forces discretization [5] based on subzonal pressures, which helps to mitigate spurious vorticity and hourglass effects. The conservation of total energy has been achieved by the compatible discretization of divergence and gradient operators proposed in [6], [7].

Moreover, the artificial viscosity (AV) has drawn large attention: The multidimensional formulation of AV has been proposed in [8], followed by the schemes constructed using mimetic Finite Difference Method in [9], [10], based on Riemann solvers [11]–[15], or those focused on symmetry preservation [16]–[18].

Despite these achievements, which made the staggered schemes very successful and widely used, there are still some problems difficult or impossible to overcome: the implementation of all mentioned improvements can be demanding, they suffer from mesh imprinting and symmetry breaking, and due to the staggered discretization, variables are not conserved over the same space, which complicates the use in the context of Arbitrary Lagrangian-Eulerian (ALE) frameworks or assessment of analytical properties of numerical solutions.

The second group of Lagrangian methods is called *Cell-centered methods*. The cell-centered methods keep all quantities in the cells. The history of cell-centered methods dates to the 50s when the first cell-centered 1D method, although in Eulerian coordinate, was proposed by S. K. Godunov[19], [20], and it has taken his name. Godunov method assumes that each cell is represented by a constant state and each edge hence represents the interface between those states, so the intercell fluxes can be then treated like a Riemann problem, either by using an exact or approximate Riemann solver. This method is robust and straightforward, yet its extension into either multiple dimensions or Lagrangian coordinates is not trivial.

The first attempt to extend the Godunov method into multiple spatial dimensions and Lagrangian coordinates was the computational code CAVEAT [21], where the edge fluxes were computed using 1D approximate Riemann solver, and nodal velocities were obtained using the weighted least squares algorithm. The method was not able to satisfy the Geometric Conservation Law (GCL), which led to spurious artificial mesh movement. This drawback was later partially solved in [22] by incorporating complex filtering procedures.

The lack of GCL compatibility and the need to use computationally expensive filtering was two decades, later addressed by first-order cell-centered Finite Volume GLACE (Godunov LAgrangian scheme Conservative for total Energy) scheme [23], [24]. The GLACE scheme introduced a multidimensional nodal solver that was able to recover nodal velocities in a way compatible with GCL. A unique nodal velocity is computed using node-centered approximate Riemann solver, assuming multiple nodal pressures, one per each surrounding cell. The scheme conserves the momentum and the total energy, but it is sensitive to the cell aspect ratio, which can lead to spurious numerical instabilities [25]. This can be a severe problem for Lagrangian simulations of Inertial Confinement Fusion (ICF), where one usually uses the computational grids with cells characterized by a high aspect ratio.

In 2007 P.H. Maire et al. proposed an alternative EUCCLHYD (Explicit Unstructured Cell-Centered Lagrangian HYDrodynamics) scheme [26] that does not suffer from the sensitivity to the cell aspect ratio. This scheme conserves momentum and total energy, satisfies GCL, and was later extended in multiple directions, including high-order accuracy, based first on the piece-wise linear monotonic reconstruction of cell pressure and velocity [27] and later using Generalized Riemann Problem [28] presented in [29], [30], application to axisymmetric rz geometry [29] and 3D Cartesian geometry [31], elastic-plastic flow [32]–[34] and unified formulation of subzonal force-based discretization [35].

Based on these ground-breaking papers, plenty of cell-centered methods have been proposed over the past decade, including, e.g. a new tensor dissipation model enforcing symmetry of viscous stress tensor [36], utilizing different Riemann solvers [37], and many others [38]–[43].

We should not omit the schemes based on the Finite Element Method. According to [44], the first interpretation of classical methods [3], [45] was proposed by Lascaux in [46], [47]. Further development of this approach brings various interesting staggered schemes, e.g., compatible finite element scheme [48]. Another interesting method based on a variational multiscale stabilized approach has been proposed in [49], [50]. Based on the observation that the spatial discretization based on straight lines limits to the second order of accuracy [51], several methods using curvilinear cell edges have been proposed [44], [52], [53].

The different development direction in Lagrangian cell-centered schemes was started by the Lax-Wendroff (LW) scheme in [54], followed by its extension into 2D based on gradient matrices of flux functions [55], which was, however, not very practical. In [56], Richtmyer proposed to write the LW scheme in a two-step predictor-corrector form. Each step of the scheme begins with evaluating auxiliary nodal quantities by the substep, usually referred to as Lax-Friedrichs (LF) predictor, followed by a corrector in which the cell value update is computed. The stability of these methods in 2D was assessed in [57] in Eulerian coordinates. In Lagrangian coordinates, the mesh positions can be updated using nodal velocities computed by the LF predictor.

The Richtmyer form of the LW scheme is second-order accurate in space and time. It is a well-known fact that the second-order accurate methods suffer from dispersion caused by third-order truncation error [58]. This dispersion is manifested by the formation of spurious nonphysical oscillations behind shock waves, and the presence of such features, however, makes the computed solution useless. This drawback was addressed by the composite schemes [59]–[61] and later by application of Flux Corrected Transport methodology in [62]. However, none of these methods suggested to use artificial viscosity, although already proposed in the first paper [54]. Designing a new artificial dissipation for the Richtmyer scheme based on Harten-Lax-van Leer (HLL) scheme [63] will be the main subject of this thesis.

In this work, we will present LW+n schemes originally published in [64], followed by its extension for solid dynamics [65], cylindrical rz geometry [66], [67], and unstructured meshes [68].

Chapter 2

Lagrangian and Eulerian description

In this chapter, we will follow definitions and notations presented in [26], [69]–[73]. Let us start by introducing d-dimensional Euclidean space $\mathcal{R}^d$, where $d \in \{1, 2, 3\}$. For $d = 3$ this space has coordinates (x, y, z) and orthonormal basis $(\boldsymbol{e}_x, \boldsymbol{e}_y, \boldsymbol{e}_z)$, the reduction to the lower dimension is clear. Next, let us consider the particle P moving with the fluid, which at initial time $t = 0$ occupies initial position $\boldsymbol{X} = (X_1, X_2, X_3) = (X, Y, Z)$, and at time $t > 0$, it has moved to position $\boldsymbol{x} = (x_1, x_2, x_3) = (x, y, z)$. Then current position $\boldsymbol{x}$ can be written as a function of initial position $\boldsymbol{X}$ and time t in terms of transformation

$$\boldsymbol{\varphi} : \boldsymbol{X} \to \boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t), \tag{2.1}$$

while

$$\boldsymbol{x}(0) = \boldsymbol{\varphi}(\boldsymbol{X}, 0) = \boldsymbol{X}.$$

We assume that $\boldsymbol{\varphi}$ has continuous derivatives up to second order almost everywhere and that the inverse transformation

$$\boldsymbol{\Phi} : \boldsymbol{x} \to \boldsymbol{X} = \boldsymbol{\varphi}^{-1}(\boldsymbol{x}, t) \tag{2.2}$$

exists. Let Ω be a region in $\mathcal{R}^d$ then its moving image using flow map (2.1) $\omega = \omega(t)$ reads

$$\omega(t) = \{\boldsymbol{x} = \boldsymbol{\varphi}(\boldsymbol{X}, t) | \boldsymbol{X} \in \Omega\}, \tag{2.3}$$

while $\omega(0) = \Omega$. The boundaries of Ω and ω will be labeled $\partial\Omega$ and $\partial\omega$ respectively, $\boldsymbol{N}$ is the outward unit normal vector of Ω and $\boldsymbol{n}$ is an outward unit normal vector of ω. The illustration of flow map(2.1) is presented in Figure 2.1.

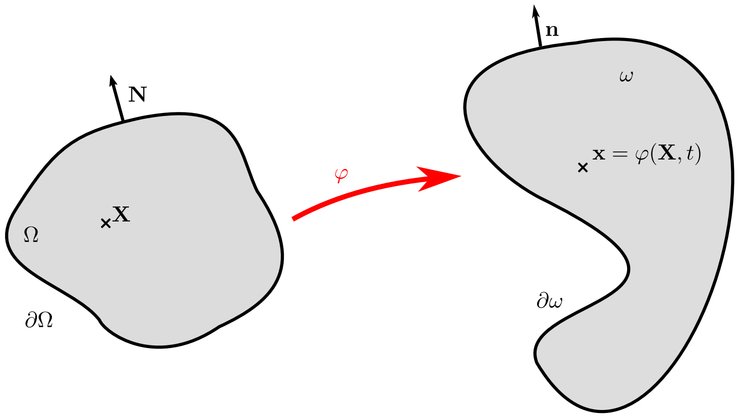


Figure 2.1: Flow map (2.1) notation.

The variables $(\boldsymbol{x}, t)$ will be referred to as *spatial variables*, while the variables $(\boldsymbol{X}, t)$ will be called *material variables*.

Now we can introduce two formalisms of flow dynamics description, namely Eulerian and Lagrangian. The Eulerian description, which is also known as *spatial description*, is characterized by observing the fluid flow in the positions fixed in space and uses spatial variables $(\boldsymbol{x}, t)$ as its independent variables. In other words, the fluid flow is characterized by the fluxes through a given control volume around a fixed point in space. Such control volume does not change with time. A fixed reference frame is used.

On the other hand, the Lagrangian description, referred to also as *material description*, observes the fluid flow in moving coordinates bound to the fixed flow particles and uses material variables $(\boldsymbol{X}, t)$ as independent variables. That means there is no mass flux through the control volume around the flow particle, but such control volume moves and deforms with the time. Moving reference frame is used.

Any scalar or vector flow quantity f can be expressed in both variables using the following relation:

$$f = f(\boldsymbol{x}, t) = f(\boldsymbol{\varphi}(\boldsymbol{X}, t), t) = f(\boldsymbol{X}, t). \tag{2.4}$$

More precisely, $f(\boldsymbol{x}, t)$ is the value of quantity f observed by fluid particle located at position $\boldsymbol{x}$ and time t, while $f(\boldsymbol{X}, t)$ is the value of quantity f observed at time t by the particle that was initially located at $\boldsymbol{X}$.

2.1 Material derivative

Let a flow quantity f be a function sufficiently smooth to allow computation of its first partial derivatives with respect to $\boldsymbol{x}$ and $\boldsymbol{X}$, then we can define *material derivative* $\mathrm{d}f/\mathrm{d}t$ and *spatial derivative* $\partial f/\partial t$ as:

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t}(\boldsymbol{X}, t), \qquad \frac{\partial f}{\partial t} = \frac{\partial f}{\partial t}(\boldsymbol{x}, t) \tag{2.5}$$

The *material derivative* of f measures the rate of change of f following a fluid particle initially at $\boldsymbol{X}$ along with its motion, while the *spatial derivative* gives the rate of change of f observed at a fixed location $\boldsymbol{x}$.

The fluid velocity is denoted by $\boldsymbol{U}$ and is defined as the material derivative of the position vector $\boldsymbol{x}(t)$:

$$\boldsymbol{U}(\boldsymbol{X},t) = \frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \frac{\partial \boldsymbol{\varphi}(\boldsymbol{X},t)}{\partial t} \tag{2.6}$$

Although the velocity $\boldsymbol{U}$ is a function of Lagrangian variables, it can also be expressed in Eulerian variables writing $\boldsymbol{U} = \boldsymbol{U}(\boldsymbol{x},t)$. If the velocity field expressed in Eulerian variables is known, the fluid flow can be determined by solving the system of ordinary differential equations

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{U}(\boldsymbol{x},t) \tag{2.7}$$

with initial condition $\boldsymbol{x}(\boldsymbol{X},0) = \boldsymbol{X} \forall \boldsymbol{x} \in \omega(t)$. Let us consider scalar physical quantity f expressed in Eulerian variables, i.e. $f = f(\boldsymbol{x},t)$, then using the chain rule and the definition (2.1), its material derivative is

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial f}{\partial t} + \left(\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t}\right) \cdot \boldsymbol{\nabla}_x f \tag{2.8}$$

where $\boldsymbol{\nabla}_x$ is a gradient operator with respect to Eulerian variables given by

$$\boldsymbol{\nabla}_x f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z}\right).$$

After substitution of velocity definition (2.6) into (2.8) we finally get the material derivative of f:

$$\frac{\mathrm{d}f}{\mathrm{d}t} = \frac{\partial}{\partial t} f + \boldsymbol{U} \cdot \boldsymbol{\nabla}_x f. \tag{2.9}$$

The Jacobian matrix of transformation (2.1) reads

$$\mathbb{F} = \frac{\partial(x,y,z)}{\partial(X,Y,Z)} = \left(\frac{\partial x_i}{\partial X_j}\right). \tag{2.10}$$

Assuming the existence of inversion (2.2), the determinant $J = \det \mathbb{F}$ exists, it is finite and nonzero. Let us remark that $\mathbb{F}$ is often referred to as deformation tensor and $\mathbb{F} = \mathbb{I}$ for $t = 0$, where $\mathbb{I}$ is matrix identity. Let $\mathrm{d}V$ denote a Lagrangian volume element, and $\mathrm{d}v$ denote its Eulerian image, then the transformation from Lagrangian to Eulerian volume is given by:

$$\mathrm{d}v = J\mathrm{d}V. \tag{2.11}$$

The material derivative of flow map Jacobian J writes:

$$\frac{\mathrm{d}J}{\mathrm{d}t} = J\boldsymbol{\nabla}_x \cdot \boldsymbol{U}. \tag{2.12}$$

To prove this useful lemma, let us first rewrite the Jacobian (2.10) using the Levi-Civita symbol ε_{ijk}. Following Einstein's summation convention, we can write:

$$J = \varepsilon_{ijk}\frac{\partial x_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k} \tag{2.13}$$

Using the definition of velocity (2.6), and assuming sufficient smoothness of $\boldsymbol{\varphi}$, we can write:

$$\begin{aligned}\frac{\mathrm{d}J}{\mathrm{d}t} &= \frac{\mathrm{d}}{\mathrm{d}t}\varepsilon_{ijk}\left(\frac{\partial x_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k}\right)\\ &= \varepsilon_{ijk}\left(\frac{\partial U_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k}\right) + \varepsilon_{ijk}\left(\frac{\partial x_1}{\partial X_i}\frac{\partial U_2}{\partial X_j}\frac{\partial x_3}{\partial X_k}\right) + \varepsilon_{ijk}\left(\frac{\partial x_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial U_3}{\partial X_k}\right).\end{aligned} \tag{2.14}$$

Let us now focus on the first term of the right-hand side of (2.14). After the use of chain rule

$$\frac{\partial U_1}{\partial X_i} = \frac{\partial U_1}{\partial x_1}\frac{\partial x_1}{\partial X_i} + \frac{\partial U_1}{\partial x_2}\frac{\partial x_2}{\partial X_i} + \frac{\partial U_1}{\partial x_3}\frac{\partial x_3}{\partial X_i}$$

the term can be rewritten into:

$$\begin{aligned}&\varepsilon_{ijk}\frac{\partial U_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k} =\\ &= \frac{\partial U_1}{\partial x_1}\varepsilon_{ijk}\frac{\partial x_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k} + \frac{\partial U_1}{\partial x_2}\varepsilon_{ijk}\frac{\partial x_2}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k} + \frac{\partial U_1}{\partial x_3}\varepsilon_{ijk}\frac{\partial x_3}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k}.\end{aligned} \tag{2.15}$$

Since the second and the third terms in (2.15) pose determinants with equal rows, the only nonzero term is the first one and therefore:

$$\varepsilon_{ijk}\frac{\partial U_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k} = \frac{\partial U_1}{\partial x_1}\varepsilon_{ijk}\frac{\partial x_1}{\partial X_i}\frac{\partial x_2}{\partial X_j}\frac{\partial x_3}{\partial X_k} = \frac{\partial U_1}{\partial x_1}J. \tag{2.16}$$

The similar arguments also apply for the second and the third terms of (2.14), so we can finally write

$$\frac{\mathrm{d}J}{\mathrm{d}t} = \frac{\partial U_1}{\partial x_1}J + \frac{\partial U_2}{\partial x_2}J + \frac{\partial U_3}{\partial x_3}J = J\boldsymbol{\nabla}_x \cdot \boldsymbol{U}. \tag{2.17}$$

2.2 Reynolds transport formula

Let us assume that $\omega = \omega(t)$ is an arbitrary moving volume, and $f = f(\boldsymbol{x}, t)$ is a scalar function representing some physical quantity. The time rate of change of integral of function f over moving volume $\omega(t)$ is

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega(t)} f(\boldsymbol{x}, t)\mathrm{d}v. \tag{2.18}$$

Our objective is to rewrite this integral into terms of Lagrangian coordinates. More let f be a function continuously differentiable over $\omega = \omega(t)$, which moves with the velocity $\boldsymbol{U}$. Because of the dynamic boundary, the time derivative of the integral $\int_{\omega(t)} f(\boldsymbol{x}, t)\mathrm{d}v$ cannot be computed directly. Therefore we will first change the variables into the fixed

Lagrangian coordinates:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega(t)} f(\boldsymbol{x},t)\mathrm{d}v = \frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega} f(\boldsymbol{X},t)J\mathrm{d}V, \tag{2.19}$$

where ω is the image of Ω transformed using (2.3). After performing this change of variables the time derivative (2.18) can be using (2.12) rewritten:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega} f(\boldsymbol{x},t)\mathrm{d}v &= \frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega} f(\boldsymbol{X},t)J\mathrm{d}V\\ &= \int_{\Omega}\frac{\mathrm{d}}{\mathrm{d}t}(fJ)\mathrm{d}V\\ &= \int_{\Omega}\left(J\frac{\mathrm{d}}{\mathrm{d}t}f + f\frac{\mathrm{d}}{\mathrm{d}t}J\right)\mathrm{d}V\\ &= \int_{\Omega} J\left(\frac{\mathrm{d}f}{\mathrm{d}t} + f\boldsymbol{\nabla}_x\cdot\boldsymbol{U}\right)\mathrm{d}V\\ &= \int_{\omega}\left(\frac{\mathrm{d}f}{\mathrm{d}t} + f\boldsymbol{\nabla}_x\cdot\boldsymbol{U}\right)\mathrm{d}v\end{aligned} \tag{2.20}$$

which is the Reynolds transport formula for scalar function f:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega} f(\boldsymbol{x},t)\mathrm{d}v = \int_{\omega}\left(\frac{\mathrm{d}f}{\mathrm{d}t} + f\boldsymbol{\nabla}_x\cdot\boldsymbol{U}\right)\mathrm{d}v. \tag{2.21}$$

After substitution of material derivative (2.9) into (2.21) one obtains an alternative form:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega} f(\boldsymbol{x},t)\mathrm{d}v = \int_{\omega}\left(\frac{\partial f}{\partial t} + \boldsymbol{\nabla}_x\cdot(f\boldsymbol{U})\right)\mathrm{d}v, \tag{2.22}$$

Using the Green theorem

$$\int_{\omega}\boldsymbol{\nabla}\cdot\boldsymbol{U}\mathrm{d}v = \int_{\partial\omega}\boldsymbol{U}\cdot\boldsymbol{n}\mathrm{d}s, \tag{2.23}$$

where $\boldsymbol{n}$ is the unit outward vector normal to the boundary $\partial\omega$, the equation (2.22) can be rewritten into:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega} f(\boldsymbol{x},t)\mathrm{d}v = \int_{\omega}\frac{\partial f}{\partial t}\mathrm{d}v + \int_{\partial\omega}(f\boldsymbol{U})\cdot\boldsymbol{n}\mathrm{d}s, \tag{2.24}$$

Let us now repeat the derivation for a vector function $\boldsymbol{f}$:

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega} \boldsymbol{f}(\boldsymbol{x},t)\mathrm{d}v &= \frac{\mathrm{d}}{\mathrm{d}t}\int_{\Omega} \boldsymbol{f}(\boldsymbol{X},t)J\mathrm{d}V\\ &= \int_{\Omega}\frac{\mathrm{d}}{\mathrm{d}t}(\boldsymbol{f}J)\mathrm{d}V\\ &= \int_{\Omega}\left(J\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{f} + \boldsymbol{f}\frac{\mathrm{d}}{\mathrm{d}t}J\right)\mathrm{d}V\\ &= \int_{\omega}\left(\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}t} + \boldsymbol{f}\boldsymbol{\nabla}_x\cdot\boldsymbol{U}\right)\mathrm{d}v\end{aligned} \tag{2.25}$$

Reynolds transport theorem for vector function $\boldsymbol{f}$ is then:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\boldsymbol{f}(\boldsymbol{x},t)\mathrm{d}v=\int_{\omega}\left(\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}t}+\boldsymbol{f}\boldsymbol{\nabla}_{x}\cdot\boldsymbol{U}\right)\mathrm{d}v. \tag{2.26}$$

The equations (2.21) and (2.26) can be used for the derivation of governing equations, which will be demonstrated in the following section.

2.3 Conservation laws

2.3.1 Continuity equation

Let us start by deriving the continuity equation from the law of conservation of mass, which states that the mass in arbitrary material volume ω of fluid does not change due to transport. Let m_ω be the mass of fluid filling the volume ω, $\varrho(\boldsymbol{x},t)$ be the mass density (i.e. mass per unit volume) of fluid. Then the law of conservation of mass can be written as:

$$\frac{\mathrm{d}}{\mathrm{d}t}m_{\omega}=\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho(\boldsymbol{x},t)\mathrm{d}v=0. \tag{2.27}$$

Rewriting the integrand in Lagrangian coordinates, we get:

$$\int_{\omega}\varrho(\boldsymbol{x},t)\mathrm{d}v=\int_{\Omega}\varrho(\boldsymbol{X},t)J\mathrm{d}V=\int_{\Omega}\varrho^{0}(\boldsymbol{X})\mathrm{d}V, \tag{2.28}$$

where ϱ^0 is the initial density. As the law (2.27) holds for any arbitrary volume ω, it can be immediately rewritten into the differential form of the equation of continuity in a material variable:

$$\frac{\mathrm{d}}{\mathrm{d}t}(\varrho(\boldsymbol{X},t)J)=0, \tag{2.29}$$

and

$$\varrho J=\varrho^{0}. \tag{2.30}$$

The continuity equation in spatial variable can be derived by setting $f=\varrho$ into Reynolds transport theorem (2.21) and assuming (2.27):

$$\int_{\omega}\left(\frac{\mathrm{d}\varrho}{\mathrm{d}t}+\varrho\boldsymbol{\nabla}_{x}\cdot\boldsymbol{U}\right)\mathrm{d}v=0. \tag{2.31}$$

Because

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{1}{\varrho}\right)=-\frac{1}{\varrho^{2}}\frac{\mathrm{d}\varrho}{\mathrm{d}t},$$

we can also write:

$$\int_{\omega}\left(\varrho\frac{\mathrm{d}\eta}{\mathrm{d}t}-\boldsymbol{\nabla}_{x}\cdot\boldsymbol{U}\right)\mathrm{d}v=0, \tag{2.32}$$

where $\eta=1/\varrho$ is the specific volume (volume per unit mass). Since this equation should hold for any volume ω:

$$\varrho\frac{\mathrm{d}\eta}{\mathrm{d}t}-\boldsymbol{\nabla}_{x}\cdot\boldsymbol{U}=0 \tag{2.33}$$

Another useful formula expresses the time rate of change of the integral of density ϱ multiplied by an arbitrary scalar function f over the moving volume ω:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho f \mathrm{d}v = \int_{\omega}\varrho\frac{\mathrm{d}f}{\mathrm{d}t}\mathrm{d}v. \tag{2.34}$$

The variant for vector $\boldsymbol{f}$ reads

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho \boldsymbol{f} \mathrm{d}v = \int_{\omega}\varrho\frac{\mathrm{d}\boldsymbol{f}}{\mathrm{d}t}\mathrm{d}v. \tag{2.35}$$

Both can be derived by inserting ϱf or $\varrho \boldsymbol{f}$ into (2.21), or (2.26) respectively, and assuming (2.33). For further details, please refer to [69]–[71] or [73].

2.3.2 Conservation of momentum

The law of conservation of momentum is based on the second Newton's law applied to the moving control volume ω. Let $\boldsymbol{\tau}$ be an external force per unit surface acting on the control volume boundary $\partial\omega$. Then assuming no extraneous force

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho \boldsymbol{U} \mathrm{d}v - \int_{\partial\omega}\boldsymbol{\tau}\mathrm{d}s = \boldsymbol{0}. \tag{2.36}$$

If $\boldsymbol{n}$ is a unit outward normal vector to $\partial\omega$ located at $\boldsymbol{x} \in \partial\omega$, then the external force $\boldsymbol{\tau} = \boldsymbol{\tau}(\boldsymbol{x}, t, \boldsymbol{n})$ is completely defined by

$$\boldsymbol{\tau}(\boldsymbol{x}, t, \boldsymbol{n}) = \mathbb{T}\boldsymbol{n}, \tag{2.37}$$

where $\mathbb{T}$ is Cauchy tensor. After substitution of (2.37) into (2.36) and using the Green formula for tensor $\mathbb{T}$

$$\int_{\omega}\boldsymbol{\nabla}\cdot\mathbb{T}\mathrm{d}v - \int_{\partial\omega}\mathbb{T}\boldsymbol{n}\mathrm{d}s = 0, \tag{2.38}$$

the first integral on the right-hand side of (2.36) writes

$$\int_{\partial\omega}\boldsymbol{\tau}\mathrm{d}s = \int_{\omega}\boldsymbol{\nabla}\cdot\mathbb{T}\mathrm{d}v. \tag{2.39}$$

Applying this intermediate result to the left-hand side of (2.34), the equation (2.36) can be rewritten as:

$$\int_{\omega}\varrho\frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t}\mathrm{d}v - \int_{\omega}\boldsymbol{\nabla}\cdot\mathbb{T}\mathrm{d}v = \boldsymbol{0}. \tag{2.40}$$

Again under the assumption that this holds for any arbitrary volume, we obtain:

$$\varrho\frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t} - \boldsymbol{\nabla}\cdot\mathbb{T} = \boldsymbol{0}, \tag{2.41}$$

which is nothing else than the local equation of motion. For a perfect fluid $\boldsymbol{\tau} = -P\mathbb{I}$, where P is pressure, and $\mathbb{I}$ is tensor identity, hence (2.41) writes

$$\varrho\frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t} + \boldsymbol{\nabla} P = \boldsymbol{0}. \tag{2.42}$$

2.3.3 Conservation of energy

The law of conservation of energy states that the time rate of increase of total energy in control volume ω equals to the sum of the power of applied external forces:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\omega \varrho E \mathrm{d}v - \int_{\partial\omega} \boldsymbol{\tau}\cdot\boldsymbol{U}\mathrm{d}s = 0, \tag{2.43}$$

where

$$E = \frac{1}{2}|\boldsymbol{U}|^2 + \varepsilon$$

is the specific total energy (energy per unit mass), ε is the specific internal energy. Following similar arguments as in the derivation of momentum equation, the equation (2.43) can be rewritten as:

$$\int_\omega \varrho\frac{\mathrm{d}E}{\mathrm{d}t}\mathrm{d}v - \int_\omega \boldsymbol{\nabla}\cdot(\mathbb{T}\boldsymbol{U})\mathrm{d}v = 0, \tag{2.44}$$

which again holds for any arbitrary volume ω and therefore:

$$\varrho\frac{\mathrm{d}E}{\mathrm{d}t} - \boldsymbol{\nabla}\cdot(\mathbb{T}\boldsymbol{U}) = 0. \tag{2.45}$$

For a perfect fluid, this becomes:

$$\varrho\frac{\mathrm{d}E}{\mathrm{d}t} + \boldsymbol{\nabla}\cdot(P\boldsymbol{U}) = 0. \tag{2.46}$$

2.3.4 Geometric Conservation Law

Geometry Conservation Law (GCL) expresses the time rate of change of moving control volume $\omega = \int_\omega \mathrm{d}v$. For GCL derivation, we will use equations (2.12) and (2.11):

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\omega \mathrm{d}v = \frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega J\mathrm{d}V = \frac{\mathrm{d}}{\mathrm{d}t}\int_\Omega J\boldsymbol{\nabla}\cdot\boldsymbol{U}\mathrm{d}V = \int_\Omega J\boldsymbol{\nabla}\cdot\boldsymbol{U}\mathrm{d}V = \int_\omega \boldsymbol{\nabla}\cdot\boldsymbol{U}\mathrm{d}v \tag{2.47}$$

Using the Green theorem (2.23) we can also write:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_\omega \mathrm{d}v = \int_{\partial\omega} \boldsymbol{U}\cdot\boldsymbol{n}\mathrm{d}v. \tag{2.48}$$

As one can see, GCL connects the time rate of change of control volume ω with the displacement of its boundary $\partial\omega$.

2.3.5 Euler equations in Lagrangian coordinates

Let us now for the sake of simplicity rewrite the whole system of Euler equations in Lagrangian coordinates (2.33),(2.41), and (2.45):

$$\varrho\frac{\mathrm{d}\eta}{\mathrm{d}t} - \boldsymbol{\nabla}\cdot\boldsymbol{U} = 0, \tag{2.49a}$$

$$\varrho\frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t} + \boldsymbol{\nabla}P = \boldsymbol{0}, \tag{2.49b}$$

$$\varrho\frac{\mathrm{d}E}{\mathrm{d}t} + \boldsymbol{\nabla}\cdot(P\boldsymbol{U}) = 0, \tag{2.49c}$$

where equation (2.49a) is the equation of continuity (the law of conservation of mass), (2.49b) is the law of conservation of momentum, and (2.49c) is the law of conservation of total energy.

2.4 Lagrangian finite volume

The integral form of system (2.49) with Geometric Conservation Law (2.48) writes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho\mathrm{d}v = 0, \tag{2.50a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\mathrm{d}v - \int_{\partial\omega}\boldsymbol{U}\cdot\boldsymbol{n}\mathrm{d}s = 0, \tag{2.50b}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho\boldsymbol{U}\mathrm{d}v + \int_{\partial\omega}P\boldsymbol{n}\mathrm{d}s = \boldsymbol{0}, \tag{2.50c}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho E\mathrm{d}v + \int_{\partial\omega}P\boldsymbol{U}\cdot\boldsymbol{n}\mathrm{d}s = 0. \tag{2.50d}$$

Let us now consider an arbitrary moving fluid element V enclosed by ∂V. Its volume V, mass m_V, momentum $\boldsymbol{M}_V$, and total energy $\mathcal{E}_V$ are given by:

$$V = \int_V \mathrm{d}v, \qquad m_V = \int_V \varrho\mathrm{d}v, \qquad \boldsymbol{M}_V = \int_V \varrho\boldsymbol{U}\mathrm{d}v, \qquad \mathcal{E}_V = \int_V \varrho E\mathrm{d}v. \tag{2.51}$$

Then system (2.50) can be simplified to the following single vector equation:

$$\frac{\mathrm{d}\boldsymbol{W}_V}{\mathrm{d}t} = \int_{\partial V}\boldsymbol{F}\cdot\boldsymbol{n}\mathrm{d}s, \tag{2.52}$$

where $\boldsymbol{W}$ is the vector of conserved variables, and $\boldsymbol{F}$ is the vector of fluxes through the boundary $\partial\omega$ given by:

$$\boldsymbol{W}_V = (V, \boldsymbol{M}_V, \mathcal{E}_V)^t, \qquad \boldsymbol{F} = (\boldsymbol{U}, -P\mathbb{I}, -P\boldsymbol{U})^t. \tag{2.53}$$

2.5 Ideal Gas Equation of State

The system (2.50) contains more variables than the equations, and hence some additional relation has to be provided. The common closure is some equation of state (EOS), which connects thermodynamical quantities like density, Cauchy stress tensor or pressure, specific internal energy, and temperature. For hydrodynamics, we will use the ideal gas EOS.

The ideal gas equation of state is usually given by equation

$$P = R\varrho T, \tag{2.54}$$

where P is pressure, R is the ideal gas constant, ϱ is density, and T is temperature. The ideal gas constant is

$$R = C_P - C_V,$$

where C_P and C_V are specific heats at constant pressure and volume respectively. For a polytropic ideal gas, the specific internal energy is a function of temperature:

$$\mathcal{E} = C_P T.$$

If we introduce Poisson constant $\gamma = \frac{C_P}{C_V}$, we will obtain pressure as the function of density and specific internal energy:

$$P(\varrho, \varepsilon) = (\gamma - 1)\varrho\varepsilon, \tag{2.55}$$

that is the form of ideal gas EOS that will be of the most use.

Since several later presented tests are defined by initial pressure and density, we can also express the specific internal energy as the function of density and pressure:

$$\varepsilon(\varrho, P) = \frac{P}{(\gamma - 1)\varrho}. \tag{2.56}$$

Finally, let us define the sound speed c_s, which will be necessary for time step computation. For the case of ideal gas EOS, it is given by:

$$c_s = \sqrt{\frac{\gamma P}{\varrho}} = \sqrt{\gamma(\gamma - 1)\varepsilon}. \tag{2.57}$$

Chapter 3

Lagrangian hydrodynamics in one spatial dimension

3.1 Governing equations in 1D

In 1D, the governing equations (2.49a-2.49c) can be simplified to:

$$\varrho \frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{\partial U}{\partial x} = 0, \tag{3.1a}$$

$$\varrho \frac{\mathrm{d}U}{\mathrm{d}t} + \frac{\partial P}{\partial x} = 0, \tag{3.1b}$$

$$\varrho \frac{\mathrm{d}E}{\mathrm{d}t} + \frac{\partial UP}{\partial x} = 0, \tag{3.1c}$$

where the total (material) derivative (2.9) reduces to:

$$\frac{\mathrm{d}}{\mathrm{d}t} = \frac{\partial}{\partial t} + U\frac{\partial}{\partial x}. \tag{3.2}$$

Using the mass coordinate s:

$$s = \int_{x_0}^{x} \varrho(x', t)\mathrm{d}x', \qquad \frac{\mathrm{d}s}{\mathrm{d}x} = \varrho, \tag{3.3}$$

the system(3.1a-3.1c) can be rewritten into the following form:

$$\frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{\partial U}{\partial s} = 0, \tag{3.4a}$$

$$\frac{\mathrm{d}U}{\mathrm{d}t} + \frac{\partial P}{\partial s} = 0, \tag{3.4b}$$

$$\frac{\mathrm{d}E}{\mathrm{d}t} + \frac{\partial (PU)}{\partial s} = 0, \tag{3.4c}$$

which is better suited for a numerical solution.

3.2 Geometry

The computational domain is given by a 1D interval divided into N cells by $N+1$ points. The nodes will be indexed by half-integers $\frac{1}{2}, \frac{3}{2}, \ldots, N+\frac{1}{2}$ while the cells will be indexed using integers $1, 2, \ldots, N$. The cell positions are given by a simple average of nodal positions:

$$x_i^n = \frac{x_{i+\frac{1}{2}}^n + x_{i-\frac{1}{2}}^n}{2}. \tag{3.5}$$

Let us now define some geometrical quantities. The mass of i-th cell m_i, and the mass of $i+\frac{1}{2}$th node $m_{i+\frac{1}{2}}$ will be $m_{i+\frac{1}{2}}$. The volume (length) of i-th cell at time level n, V_i^n and $i+\frac{1}{2}$-th node $V_{i+\frac{1}{2}}^n$ are

$$V_i^n = \Delta x_i^n = x_{i+\frac{1}{2}}^n - x_{i-\frac{1}{2}}^n, \qquad V_{i+\frac{1}{2}}^n = \Delta x_{i+\frac{1}{2}}^n = \frac{\Delta x_{i+1}^n + \Delta x_i^n}{2}. \tag{3.6}$$

The 1D mesh is illustrated in Figure 3.1. The masses are computed only once at the beginning of simulation from density and volume:

$$m_i = \varrho_i^0 V_i^0, \qquad m_{i+\frac{1}{2}} = \frac{m_i + m_{i+1}}{2}, \tag{3.7}$$

the density ϱ_i^0 is given by the initial conditions, and the volume V_i^0 is computed from the initial mesh.

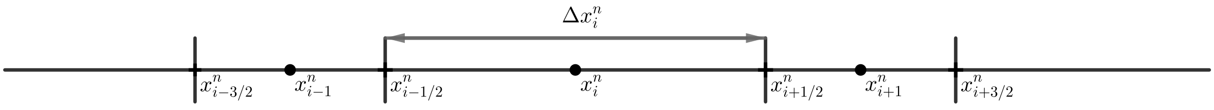


Figure 3.1: Notations for 1D geometry.

3.3 Lax-Wendroff scheme

The basic method used in this work is the well-known Lax-Wendroff scheme, which was firstly published in 1960 in [54]. Later in 1962, R.D. Richtmyer proposed that LW can be rewritten in a simple predictor-corrector form [56]. One step of the LW method is divided into two sub-steps, usually known as Lax-Friedrichs (LF) predictor and the corrector. In the LF predictor phase, nodal estimates of state variables are obtained. They are then used in the corrector for evaluation fluxes over the cell edges. The scheme is formally second-order accurate.

3.3.1 Derivation of the Lax Wendroff scheme

Let us briefly illustrate the derivation of the Lax-Wendroff scheme [54] for scalar advection equation in Eulerian variable:

$$\frac{\partial u(t,x)}{\partial t} + a\frac{\partial u(t,x)}{\partial x} = 0, \tag{3.8}$$

where $u(t, x)$ is a scalar function, and a is the advection speed. 1D Taylor expansion of function u in time $t + \Delta t$ is:

$$u(t + \Delta t, x) = u(t, x) + \Delta t \frac{\partial u(t, x)}{\partial t} + \frac{\Delta t^2}{2} \frac{\partial^2 u(t, x)}{\partial^2 t} + \mathcal{O}(\Delta t^3) \tag{3.9}$$

From the advection equation (3.8), we know that the time derivative can be expressed using the derivative with respect to x:

$$\frac{\partial u(t, x)}{\partial t} = -a \frac{\partial u(t, x)}{\partial x}. \tag{3.10}$$

After differencing this relation with respect to t and using (3.10) again, we obtain:

$$\frac{\partial^2 u(t, x)}{\partial^2 t} = -a \frac{\partial^2 u(t, x)}{\partial t \partial x} = -a \frac{\partial}{\partial x} \left(\frac{\partial u(t, x)}{\partial t} \right) = a^2 \frac{\partial^2 u(t, x)}{\partial^2 x}. \tag{3.11}$$

Now the Taylor expansion (3.9) can be rewritten using (3.10) and (3.11):

$$u(t + \Delta t, x) = u(t, x) - a \Delta t \frac{\partial u(t, x)}{\partial x} + a^2 \frac{\Delta t^2}{2} \frac{\partial^2 u(t, x)}{\partial^2 x} + \mathcal{O}(\Delta t^3). \tag{3.12}$$

Let us now define a discrete computational mesh. For $t \in \langle 0, \infty)$ and $x \in \mathbb{R}$ we define discrete values

$$t^n = n\Delta t, n \in \mathbb{N}_0, \quad x_i = i\Delta x, i \in \mathbb{Z},$$

the values of function $u(t, x)$ are approximated by discrete values $u(t^n, x_i) \approx u_i^n$. By inserting second-order accurate central differences

$$\frac{\partial u(t, x)}{\partial x} \approx \frac{u_{i+1} - u_{i-1}}{2\Delta x_i}, \qquad \frac{\partial^2 u(t, x)}{\partial^2 x} \approx \frac{u_{i+1} - 2u_i + u_{i-1}}{(\Delta x_i)^2}$$

into (3.12), omitting $\mathcal{O}$ and rearranging, we obtain:

$$\frac{u^{n+1} - u_i^n}{\Delta t} + a \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x_i} - a^2 \frac{\Delta t}{2} \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{(\Delta x_i)^2} = 0, \tag{3.13}$$

which is the Lax-Wendroff scheme for the advection equation (3.8). Using von Neumann analysis, one can show that the scheme is stable under CFL condition

$$a \frac{\Delta t}{\Delta x} \leqslant 1.$$

3.3.2 Extension of Lax-Wendroff for nonlinear systems of conservation laws

The scheme (3.13) can be extended for the nonlinear systems. In this work, we will use the form due to Richtmyer, which was introduced in [56]. One can easily verify that the scheme (3.13) can be rewritten into the predictor-corrector form:

$$u_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{u_i^n + u_{i+1}^n}{2} - a\frac{\Delta t}{2}\frac{u_{i+1}^n - u_i^n}{\Delta x}, \tag{3.14a}$$

$$u_i^{n+1} = u_i^n - a\frac{\Delta t}{2}\frac{u_{i+\frac{1}{2}}^{n+\frac{1}{2}} - u_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x}. \tag{3.14b}$$

where (3.14a) is Lax-Friedrichs (LF) predictor, and (3.14b) is the corrector. In the predictor phase, the nodal estimates of state variables at the time level $n+\frac{1}{2}$ are obtained. In the corrector phase, the nodal predictions are used to compute numerical fluxes over the nodes.

Let us now move to a nonlinear system of conservation laws given by

$$\frac{\partial \boldsymbol{u}(t,x)}{\partial t} + \frac{\partial}{\partial x}\boldsymbol{f}(\boldsymbol{u}(t,x)) = 0, \tag{3.15}$$

where $\boldsymbol{f}(\boldsymbol{u}(t,x))$ is a vector function of vector $\boldsymbol{u}$. For a special scalar choice $f(u) = au$, the equation (3.15) reduces to (3.8). For a nonlinear system (3.15), Richtmyer proposed the following form:

$$\boldsymbol{u}_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{\boldsymbol{u}_i^n + \boldsymbol{u}_{i+1}^n}{2} - \frac{\Delta t}{2}\frac{\boldsymbol{f}_{i+1}^n - \boldsymbol{f}_i^n}{\Delta x}, \tag{3.16a}$$

$$\boldsymbol{u}_i^{n+1} = \boldsymbol{u}_i^n - \frac{\Delta t}{2}\frac{\boldsymbol{f}_{i+\frac{1}{2}}^{n+\frac{1}{2}} - \boldsymbol{f}_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x}, \tag{3.16b}$$

where

$$\boldsymbol{f}_{i+1}^n = \boldsymbol{f}(\boldsymbol{u}_{i+1}^n), \qquad \boldsymbol{f}_i^n = \boldsymbol{f}(\boldsymbol{u}_i^n), \qquad \boldsymbol{f}_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \boldsymbol{f}(\boldsymbol{u}_{i+\frac{1}{2}}^{n+\frac{1}{2}}), \qquad \boldsymbol{f}_{i-\frac{1}{2}}^{n+\frac{1}{2}} = \boldsymbol{f}(\boldsymbol{u}_{i-\frac{1}{2}}^{n+\frac{1}{2}})$$

are numerical fluxes evaluated at cells $i, i+1$ or nodes $i \pm \frac{1}{2}$ respectively. For a scalar $f(u) = au$, this scheme reduces to (3.14a-3.14b).

3.3.3 Discretization in Lagrangian coordinates

Let us now rewrite system (3.4) into a vector form, which will allow us to discretize this system using scheme (3.16a-3.16b). In 1D, we can start directly from the system (3.4a-3.4c) and rewrite it into the following single vector equation:

$$\frac{\mathrm{d}\boldsymbol{w}}{\mathrm{d}t} + \frac{\partial \boldsymbol{F}}{\partial s} = \boldsymbol{0}, \tag{3.17}$$

where

$$\boldsymbol{w} = (\eta, U, E)^t, \tag{3.18}$$

are the vector of specific (per unit mass) conserved variables, and

$$\boldsymbol{F} = (U, -P, -PU)^t \tag{3.19}$$

is the vector of physical fluxes. Let us now show the discretization of system (3.17-3.19) using the Lax-Wendroff scheme. The Lax-Friedrichs predictor, which advances the solution from time level n to $n+\frac{1}{2}$, writes

$$\boldsymbol{w}_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{m_i \boldsymbol{w}_i^n + m_{i+1}\boldsymbol{w}_{i+1}^n}{m_i + m_{i+1}} + \frac{\Delta t}{m_i + m_{i+1}}(\boldsymbol{F}_{i+1}^n - \boldsymbol{F}_i^n), \tag{3.20}$$

where $\boldsymbol{F}_i^n = \boldsymbol{F}(\boldsymbol{w}_i^n)$ and $\boldsymbol{F}_{i+1}^n = \boldsymbol{F}(\boldsymbol{w}_{i+1}^n)$ are the numerical fluxes in i-th and $i+1$-th cells respectively, and Δt is the time step. Then the nodal pressures and sound speeds can be updated using EOS (2.55), and the predictor substep is finished. The corrector substep writes:

$$\boldsymbol{w}_i^{n+1} = \boldsymbol{w}_i^n + \frac{\Delta t}{m_i}(\boldsymbol{F}_{i+\frac{1}{2}}^{n+\frac{1}{2}} - \boldsymbol{F}_{i-\frac{1}{2}}^{n+\frac{1}{2}}), \tag{3.21}$$

where $\boldsymbol{F}_{i-\frac{1}{2}}^{n+\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{w}_{i-\frac{1}{2}}^{n+\frac{1}{2}})$ and $\boldsymbol{F}_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{w}_{i+\frac{1}{2}}^{n+\frac{1}{2}})$ are fluxes over the cell boundaries computed from predicted nodal values. Then the nodal positions are advanced from time level n to new time level $n+1$:

$$x_{i+\frac{1}{2}}^{n+1} = x_{i+\frac{1}{2}}^n + \Delta t U_{i+\frac{1}{2}}^{n+\frac{1}{2}} \tag{3.22}$$

and the geometry is updated using (3.5-3.6). Finally, we update the pressures and sound speeds in cell centers using EOS and compute the new time step Δt.

3.3.4 Time step computation

The scheme described in the previous subsection is stable under the Courant-Friedrichs-Lewy (CFL) condition [74], each time step Δt is therefore computed using

$$\Delta t^{n+1} = C_{\text{CFL}} \min_{i\in\hat{N}} \left\{ \frac{\Delta x_i^n}{(c_s)_i^n} \right\}, \tag{3.23}$$

where $\hat{N} = 1, 2, \ldots, N$, C_{CFL} is CFL constant, usually set to 0.4 in 1D, and $(c_s)_i^n$ is the sound speed in cell i computed by (2.57). Additionally, we restrict each time step to increase no more than by 10% and set it to $\Delta t/2$ if the volume in any cell changes more than by 10%:

$$\Delta t^{n+1} = \begin{cases} \min(1.1\Delta t^n, \Delta t^{n+1}) & \text{if} \quad \Delta x^{n+1}/\Delta x^n \in \langle 0.9, 1.1\rangle \\ \Delta t^{n+1}/2 & \text{otherwise} \end{cases}$$

3.3.5 Accuracy of Lax-Wendroff scheme – test problem

Before we move on to the actual accuracy test, let us first define L_1 error and Numerical Order of Convergence (NOC). L_1 error is for arbitrary quantity q defined by:

$$L_1(q) = \sum_c |q_c - q_c^{ex}| V_c, \tag{3.24}$$

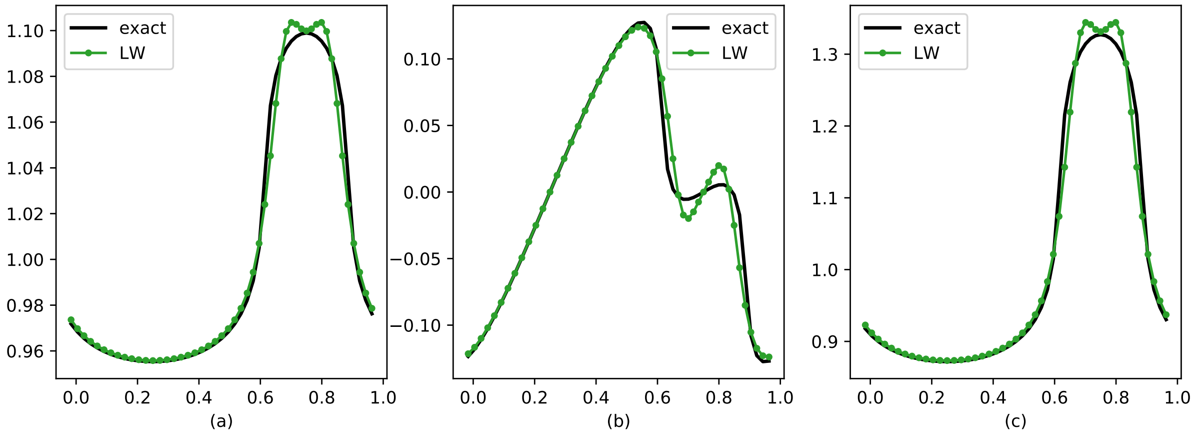
where q_c is the value of q in cell c computed by the presented method, q_c^{ex} is the exact solution in cell c, and the sum $\sum_c$ goes over all cells within the computational domain.

The numerical order of convergence is then binary logarithm of fraction of L_1 errors computed on meshes with N and $2N$ cells:

$$\mathrm{NOC}(q,N) = \log_2\left(\frac{L_1(q^N)}{L_1(q^{2N})}\right). \tag{3.25}$$

The LW scheme defined by (3.20) and (3.21) on a uniform mesh is equivalent to the LW scheme in Eulerian coordinates (3.16a-3.16b), and therefore it is nominally second-order accurate and stable under the CFL condition. The scheme remains second-order accurate and stable even for mesh smooth in mass coordinates, which is numerically verified on the following test problem taken from [75]. This test case is characterized by the initial smooth density $\varrho^0(x) = 1 + 0.1\sin(2\pi x)$, zero velocity, and the initial pressure $P^0(x) = (\varrho^0)^\gamma$, where $\gamma = 3$. For this choice of γ the exact solution can be constructed. We refer the reader to [75] for the details. The computational domain is the 1D interval $\langle 0,1\rangle$. The final time is $T = 0.8$. Periodic boundary conditions are used.

This test has been computed on initially uniform mesh with $80, 160, 320, 640$, and 1280 cells using the pure LW scheme. The results computed on a coarse mesh of 50 cells are shown in Figure 3.2. In Table 3.1, we present a Numerical Order of Convergence (NOC) and L_1 errors of density, velocity, and pressure. Pure LW scheme achieves numerically second order of convergence in the regions where the solution remains smooth.



(a) (b) (c)

Figure 3.2: The density (a), the velocity (b), and the pressure (c) by pure LW on a coarse mesh of 50 cells for smooth flow problem [75] on an initially uniform grid.

mesh	$L_1(\varrho)$	$L_1(P)$	$L_1(U)$	NOC(ϱ)	NOC(P)	NOC(U)
50	3.3e-03	1.1e-02	5.3e-03	1.5	1.5	1.5
100	1.2e-03	3.9e-03	1.8e-03	1.7	1.7	1.6
200	3.8e-04	1.2e-03	5.9e-04	1.8	1.9	1.8
400	1.1e-04	3.4e-04	1.7e-04	2.0	2.0	2.0
800	2.7e-05	8.6e-05	4.2e-05	2.0	2.0	2.0
1600	6.6e-06	2.1e-05	1.0e-05			

Table 3.1: Convergence for the problem with initially smooth data [75] computed using the pure LW scheme. L_1 error and numerical order of convergence.

3.3.6 Wendroff-White averaging in Lagrangian coordinates

The major drawback of the presented Lax-Wendroff method with predictor (3.20) is a bad performance on highly nonuniform meshes. Let us assume the smooth test described in the previous subsection on a grid generated using the following recurrent formula:

$$x_{i+1} = x_i + (\mathrm{mod}(i,2)+1)h, \qquad x_0 = 0, \qquad h = \frac{1}{3N}, \tag{3.26}$$

where N is the number of cells over the computational domain. Clearly, the odd cells of such mesh will have the length of $2h$ while the even cells will have the length of h, and hence the cell sizes will follow $h-2h-h-2h\dots$ pattern. As we can see in Figure 3.3, the result computed using the LW method on highly non-uniform mesh (3.26) is completely distorted. To address this drawback, we will use the Wendroff-White (WW) weighting proposed in [76] instead of weighting the average nodal state using sub zonal masses as in (3.20). The modified WW predictor will be

$$\boldsymbol{w}^{n+\frac{1}{2}}_{i+\frac{1}{2}} = \frac{\Delta x^n_{i+1}\boldsymbol{w}^n_i + \Delta x^n_i \boldsymbol{w}^n_{i+1}}{\Delta x^n_i + \Delta x^n_{i+1}} + \frac{\Delta t}{m_i + m_{i+1}}(\boldsymbol{F}^n_{i+1} - \boldsymbol{F}^n_i). \tag{3.27}$$

Let us note that WW weighting is the same as a linear interpolation in 1D. The results due to WW predictor (3.27) are also presented in Figure 3.3. The oscillations caused by the nonuniform grid disappeared completely.

For sake of completeness, numerical orders of convergence for the alternative weighting (3.27) can be found in Table 3.2. As one can see, the use of WW weighting in the predictor did not significantly worsen the numerical order of accuracy. All the following presented results will be computed using the new WW predictor unless otherwise stated. For more results due to LW+n schemes with originally proposed predictor (3.20), we refer the reader to the previous works [64], [65].

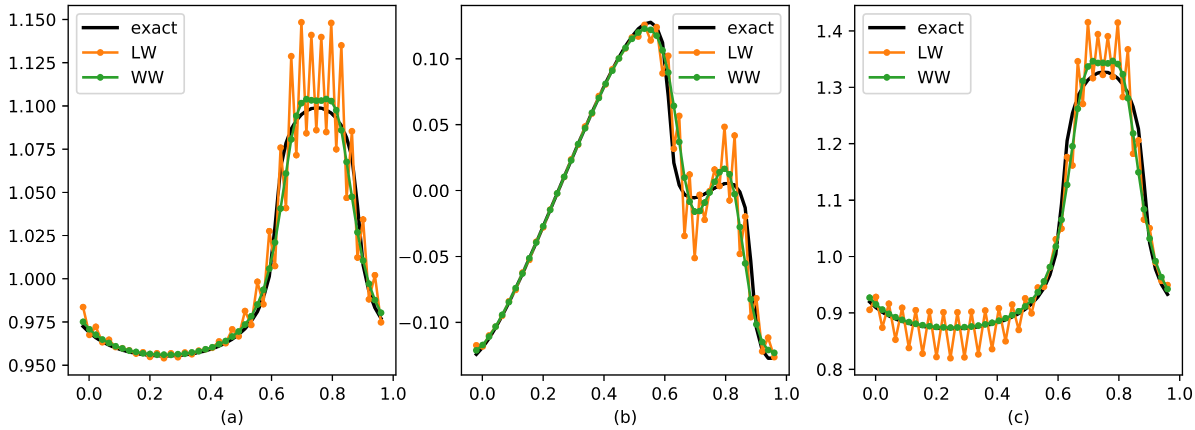


Figure 3.3: The density (a), the velocity (b), and the pressure (c) by LW and WW on 50 cells mesh for smooth flow problem [75] on an initially non-uniform grid defined by (3.26).

mesh	$L_1(\varrho)$	$L_1(P)$	$L_1(U)$	NOC(ϱ)	NOC(P)	NOC(U)
50	3.3e-03	1.1e-02	5.4e-03	1.5	1.5	1.5
100	1.2e-03	3.9e-03	1.8e-03	1.6	1.6	1.6
200	3.8e-04	1.2e-03	5.9e-04	1.8	1.8	1.8
400	1.1e-04	3.4e-04	1.7e-04	2.0	2.0	2.0
800	2.7e-05	8.7e-05	4.3e-05	2.0	2.0	2.0
1600	7.0e-06	2.2e-05	1.1e-05			

Table 3.2: L_1 error and NOC for smooth flow problem [75] computed using the Wendroff-White predictor (3.27).

3.4 HLL based dissipation

Since the Lax-Wendroff scheme is dispersive, it produces nonphysical oscillations behind the shock waves, which makes the obtained numerical solution useless. Several approaches for how to prevent the formation of these pathological features exist in the literature, e.g. [59], [61], [77], but probably the most popular approach is adding an artificial viscosity to the velocity equation, which has been firstly introduced by von Neumann and Richtmyer in [2]. In this paper, the authors suggest adding an artificial viscosity term to the pressure in the momentum equation, which acts as an additional viscous term smearing the shock front into multiple cells (about $3-5$) [78]. The common form of artificial viscosity is by Kuropatenko [79], [80].

In this work, we will use new artificial viscosity based on the Harten-Lax-van Leer scheme [63]. Let us start with the presentation of the Riemann problem and the HLL scheme.

3.4.1 Riemann problem

Riemann problem [58], [81] is a special initial value problem given by hyperbolic partial differential equation (or hyperbolic system) in 1D, which can be solved analytically. It consists of two constant states, the left $\boldsymbol{W}_\mathrm{l}$ and the right $\boldsymbol{W}_\mathrm{r}$, forming a single discontinuity in one or multiple state variables. Riemann problem is given by vector equation

$$\frac{\partial \boldsymbol{W}(x,t)}{\partial t} + \frac{\partial \boldsymbol{F}\left(\boldsymbol{W}(x,t)\right)}{\partial x} = \mathbf{0} \tag{3.28}$$

for $x \in (-\infty, \infty)$ and $t \in (0, \infty)$ and initial data with a single jump discontinuity at arbitrary position x_0:

$$\boldsymbol{W}(x,0) = \begin{cases} \boldsymbol{W}_\mathrm{l} & \text{if} \qquad x < x_0 \\ \boldsymbol{W}_\mathrm{r} & \text{if} \qquad x > x_0. \end{cases} \tag{3.29}$$

For a hyperbolic system of m equations, the solution to the Riemann problem generally consists of m waves, each associated with one of m unique eigenvalues of Jacobian $\partial \boldsymbol{F}/\partial \boldsymbol{W}$. The general structure of the solution for Euler equations is illustrated in Figure 3.4.

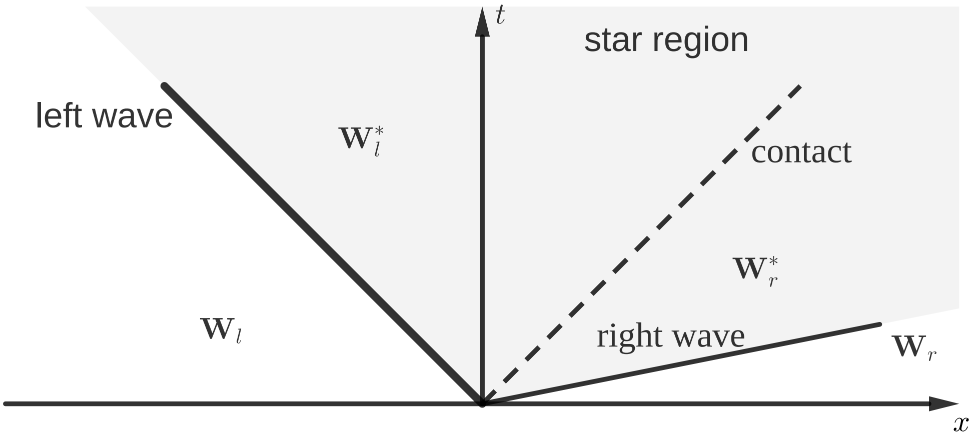


Figure 3.4: Structure of the general solution of the Riemann problem for Euler equations.

More specifically, the solution for the system (3.1a-3.1c) will be formed from three waves – the middle one will always be the contact discontinuity, the right and the left one can be either shock or rarefaction waves.

3.4.2 Godunov method

In 1959, Godunov proposed a method for the solution of hyperbolic conservation laws, which is after him called Godunov method [19]. The data in each computational cell are constant, and hence the task of finding nodal values between every two cells is a Riemann problem, which can be solved analytically or using some approximation. The solution of the Riemann problem is usually called Riemann Solver (RS). One approximate RS will be described in detail in the next subsection.

In a conservative form, the discretization of (3.28) using the Godunov method can be written as [81]:

$$\boldsymbol{W}_i^{n+1} = \boldsymbol{W}_i^n + \frac{\Delta t}{\Delta x}\left(\boldsymbol{F}_{i+\frac{1}{2}} - \boldsymbol{F}_{i-\frac{1}{2}}\right), \tag{3.30}$$

where $\boldsymbol{W}_i^n$ is the integral average of the state vector (2.53) within the i-th cell computed by:

$$\boldsymbol{W}_i^n = \frac{1}{\Delta x_i}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \boldsymbol{W}(x, t^n)\mathrm{d}x. \tag{3.31}$$

The values $\boldsymbol{W}_i^n$ are assumed to be constant within each cell. In the equation (3.30) the $\boldsymbol{F}_{i\pm\frac{1}{2}}$ are inter-cell numerical fluxes given by

$$\boldsymbol{F}_{i\pm\frac{1}{2}} = \boldsymbol{F}(\boldsymbol{W}_{i\pm\frac{1}{2}}), \tag{3.32}$$

where $\boldsymbol{W}_{i\pm\frac{1}{2}}$ are the solutions of local Riemann problem $\mathcal{RP}$ on the interface between $i-1$-th and i-th or i-th and $i+1$-th cells respectively:

$$\boldsymbol{W}_{i-\frac{1}{2}} = \mathcal{RP}(\boldsymbol{W}_{i-1}, \boldsymbol{W}_i), \quad \boldsymbol{W}_{i+\frac{1}{2}} = \mathcal{RP}(\boldsymbol{W}_i, \boldsymbol{W}_{i+1}). \tag{3.33}$$

Due to the substantial complexity of the Riemann solvers, we will not present it here in more detail and rather refer the reader to [81].

3.4.3 HLL scheme

HLL scheme [63] is a method of Godunov type. Instead of using the exact Riemann solver, the intercell flux is given by the two-wave approximation. Let us now briefly show the derivation of the HLL solver, which has been taken from [81].

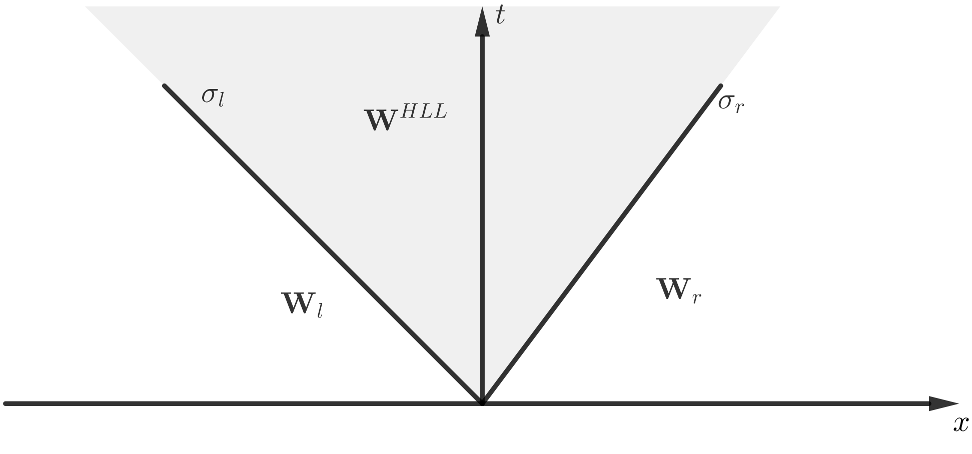


Figure 3.5: Wave diagram for HLL scheme – 3 states $\boldsymbol{W}_\mathrm{l}, \boldsymbol{W}^\mathrm{HLL}, \boldsymbol{W}_\mathrm{r}$.

Let us assume the solution to a Riemann problem, which consists only of three states $\boldsymbol{W}_\mathrm{l}, \boldsymbol{W}^\mathrm{HLL}, \boldsymbol{W}_\mathrm{r}$. The situation is illustrated in Figure 3.5.

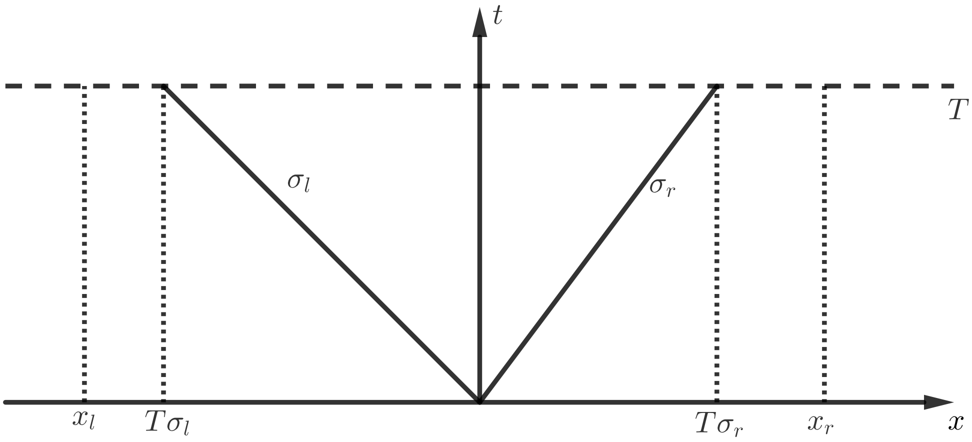


Figure 3.6: Control volume for the derivation of the HLL scheme.

Let us assume control volume $\langle x_{\mathrm{l}}, x_{\mathrm{r}}\rangle \times \langle 0, T\rangle$ as depicted in Figure 3.6. The left wave is moving by signal velocity σ_{l} while the right wave is moving with σ_{r}. The integral form of Riemann problem (3.28-3.29) can be expressed as:

$$\int_{x_{\mathrm{l}}}^{x_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x = \int_{x_{\mathrm{l}}}^{x_{\mathrm{r}}} \boldsymbol{W}(x,0)\mathrm{d}x + \int_0^T \boldsymbol{F}(\boldsymbol{W}(x_{\mathrm{l}},t))\mathrm{d}t - \int_0^T \boldsymbol{F}(\boldsymbol{W}(x_{\mathrm{r}},t))\mathrm{d}t \tag{3.34}$$

which can be rewritten into *consistency equation*:

$$\int_{x_{\mathrm{l}}}^{x_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x = x_{\mathrm{r}}\boldsymbol{W}_{\mathrm{r}} - x_{\mathrm{l}}\boldsymbol{W}_{\mathrm{l}} + T\boldsymbol{F}_{\mathrm{l}} - T\boldsymbol{F}_{\mathrm{r}}, \tag{3.35}$$

where $\boldsymbol{F}_{\mathrm{l}} = \boldsymbol{F}(\boldsymbol{W}_{\mathrm{l}})$ and $\boldsymbol{F}_{\mathrm{r}} = \boldsymbol{F}(\boldsymbol{W}_{\mathrm{r}})$. The integral on the left-hand side of (3.34) can be split by the x-intervals into:

$$\int_{x_{\mathrm{l}}}^{x_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x = \int_{x_{\mathrm{l}}}^{T\sigma_{\mathrm{l}}} \boldsymbol{W}(x,T)\mathrm{d}x + \int_{T\sigma_{\mathrm{l}}}^{T\sigma_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x + \int_{T\sigma_{\mathrm{r}}}^{x_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x \tag{3.36}$$

After evaluating the first and the third integrals on the right-hand side of (3.36) where the data remain unchanged:

$$\int_{x_{\mathrm{l}}}^{x_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x = (T\sigma_{\mathrm{l}} - x_{\mathrm{l}})\boldsymbol{W}_{\mathrm{l}} + \int_{T\sigma_{\mathrm{l}}}^{T\sigma_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x + (T\sigma_{\mathrm{r}} - x_{\mathrm{r}})\boldsymbol{W}_{\mathrm{r}}. \tag{3.37}$$

Putting (3.37) into (3.35), we obtain the expression for integral over the star region:

$$\int_{T\sigma_{\mathrm{l}}}^{T\sigma_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x = T\left(\sigma_{\mathrm{r}}\boldsymbol{W}_{\mathrm{r}} - \sigma_{\mathrm{l}}\boldsymbol{W}_{\mathrm{l}} + \boldsymbol{F}_{\mathrm{l}} - \boldsymbol{F}_{\mathrm{r}}\right). \tag{3.38}$$

If we divide (3.38) by the interval length $T(\sigma_{\mathrm{r}} - \sigma_{\mathrm{l}})$, then we obtain

$$\frac{1}{T(\sigma_{\mathrm{r}} - \sigma_{\mathrm{l}})} \int_{T\sigma_{\mathrm{l}}}^{T\sigma_{\mathrm{r}}} \boldsymbol{W}(x,T)\mathrm{d}x = \frac{\sigma_{\mathrm{r}}\boldsymbol{W}_{\mathrm{r}} - \sigma_{\mathrm{l}}\boldsymbol{W}_{\mathrm{l}} + \boldsymbol{F}_{\mathrm{r}} - \boldsymbol{F}_{\mathrm{l}}}{\sigma_{\mathrm{l}} - \sigma_{\mathrm{r}}}. \tag{3.39}$$

The left-hand side of (3.39) is nothing else than the integral average of the exact solution in the region between the two signal speeds σ_l, σ_r, hence we will denote the right-hand side of (3.39) by

$$\boldsymbol{W}^{\mathrm{HLL}} = \frac{\sigma_\mathrm{r}\boldsymbol{W}_\mathrm{r} - \sigma_\mathrm{l}\boldsymbol{W}_\mathrm{l} + \boldsymbol{F}_\mathrm{r} - \boldsymbol{F}_\mathrm{l}}{\sigma_\mathrm{l} - \sigma_\mathrm{r}}. \tag{3.40}$$

Let us now evaluate the left flux along the t-axis $\boldsymbol{F}_{0l}$. Considering interval $\langle x_\mathrm{l}, 0\rangle \times \langle 0, T\rangle$:

$$\boldsymbol{F}_{0l} = \boldsymbol{F}_\mathrm{l} - \sigma_\mathrm{l}\boldsymbol{W}_\mathrm{l} - \frac{1}{T}\int_{T\sigma_\mathrm{l}}^{0} \boldsymbol{W}(x,T)\mathrm{d}x \tag{3.41}$$

and similarly for $\langle 0, x_\mathrm{r}\rangle \times \langle 0, T\rangle$:

$$\boldsymbol{F}_{0r} = \boldsymbol{F}_\mathrm{r} - \sigma_\mathrm{r}\boldsymbol{W}_\mathrm{r} + \frac{1}{T}\int_{0}^{T\sigma_\mathrm{r}} \boldsymbol{W}(x,T)\mathrm{d}x \tag{3.42}$$

we get the right flux along the t axis $\boldsymbol{F}_{0r}$. Complete HLL approximate Riemann solver is defined by:

$$\widetilde{\boldsymbol{W}}(x,t) = \begin{cases} \boldsymbol{W}_\mathrm{l} & \text{if} \quad x \leqslant \sigma_\mathrm{l} t, \\ \boldsymbol{W}^{\mathrm{HLL}} & \text{if} \quad \sigma_\mathrm{l} t \leqslant x \leqslant \sigma_\mathrm{r} t, \\ \boldsymbol{W}_\mathrm{r} & \text{if} \quad \sigma_\mathrm{r} t \leqslant x_\mathrm{r}. \end{cases} \tag{3.43}$$

Let us remark that the HLL flux is not given by $\boldsymbol{F}(\boldsymbol{W}^{\mathrm{HLL}})$, but by substituting (3.40) into (3.41) or (3.42), which gives us:

$$\begin{aligned} \boldsymbol{F}^{\mathrm{HLL}} &= \boldsymbol{F}_\mathrm{l} + \sigma_\mathrm{l}(\boldsymbol{W}^{\mathrm{HLL}} - \boldsymbol{W}_\mathrm{l}), \\ \boldsymbol{F}^{\mathrm{HLL}} &= \boldsymbol{F}_\mathrm{r} + \sigma_\mathrm{r}(\boldsymbol{W}^{\mathrm{HLL}} - \boldsymbol{W}_\mathrm{r}) \end{aligned} \tag{3.44}$$

After substitution of (3.40) into (3.44) we finally obtain the desired HLL flux:

$$\boldsymbol{F}^{\mathrm{HLL}} = \frac{\sigma_\mathrm{r}\boldsymbol{F}_\mathrm{l} - \sigma_\mathrm{l}\boldsymbol{F}_\mathrm{r} + \sigma_\mathrm{l}\sigma_\mathrm{r}(\boldsymbol{W}_\mathrm{r} - \boldsymbol{W}_\mathrm{l})}{\sigma_\mathrm{r} - \sigma_\mathrm{l}}. \tag{3.45}$$

Now we can rewrite the HLL inter-cell flux for node $i + \frac{1}{2}$:

$$\boldsymbol{F}^{\mathrm{HLL}}_{i+\frac{1}{2}} = \begin{cases} \boldsymbol{F}_i & \text{if} \quad 0 \leqslant \sigma_i \\ \frac{\sigma_{i+1}\boldsymbol{F}_i - \sigma_i\boldsymbol{F}_{i+1} + \sigma_i\sigma_{i+1}(\boldsymbol{W}_{i+1} - \boldsymbol{W}_i)}{\sigma_{i+1} - \sigma_i}. & \text{if} \quad \sigma_i \leqslant 0 \leqslant \sigma_{i+1}\ . \\ \boldsymbol{F}_{i+1} & \text{if} \quad 0 \geqslant \sigma_{i+1} \end{cases} \tag{3.46}$$

3.4.4 HLL scheme in Lagrangian moving frame

1D cell-centered HLL scheme can be written in the mass coordinate:

$$m_i \boldsymbol{w}_i^{n+1} = m_i \boldsymbol{w}_i^{n} + \Delta t(\boldsymbol{F}^n_{i+\frac{1}{2}} - \boldsymbol{F}^n_{i-\frac{1}{2}}) \tag{3.47}$$

where the flux expressed in a reference frame moving with node $i+\frac{1}{2}$ writes:

$$\boldsymbol{F}_{i+\frac{1}{2}} = \frac{\sigma_i \boldsymbol{w}_i + \sigma_{i+1}\boldsymbol{w}_{i+1}}{\sigma_i + \sigma_{i+1}} + \frac{\sigma_i \sigma_{i+1}(\boldsymbol{w}_i - \boldsymbol{w}_{i+1})}{\sigma_i + \sigma_{i+1}}, \tag{3.48}$$

where σ_i, σ_{i+1} are HLL speeds. There are more possible choices of σ, e.g. [82] suggests to use acoustic impedances (Lagrangian sound speeds) defined as $-\sigma_i = -\varrho_i (c_s)_i$, $\sigma_{i+1} = \varrho_{i+1}(c_s)_{i+1}$, but instead, we use modified acoustic impedances

$$\sigma_{i+1} = \varrho_{i+1}\left((c_s)_{i+1} + |u_{i+1} - u_i|\right), \qquad \sigma_i = \varrho_i\left((c_s)_i + |u_{i+1} - u_i|\right). \tag{3.49}$$

The first term on the right-hand side of (3.48) is the physical flux over the node, the second term

$$\boldsymbol{D}^d_{i+\frac{1}{2}} = \frac{\sigma_i \sigma_{i+1}(\boldsymbol{w}_i - \boldsymbol{w}_{i+1})}{\sigma_i + \sigma_{i+1}} \tag{3.50}$$

is the dissipative part which we will use as the basic building block of our artificial dissipation.

3.4.5 LW+n schemes

The LW+n scheme is the Lax-Wendroff scheme presented in Section 3.3 with the additional HLL-based dissipative flux:

$$m_i \boldsymbol{w}_i^{d,n+1} = m_i \boldsymbol{w}_i^{d,n} + \Delta t \left[(\boldsymbol{F}^n_{i+\frac{1}{2}} - \boldsymbol{F}^n_{i-\frac{1}{2}}) + D_\tau \cdot \left(\boldsymbol{D}^d_{i+\frac{1}{2}} - \boldsymbol{D}^d_{i-\frac{1}{2}} \right) \right] \tag{3.51}$$

where $\boldsymbol{D}^d_{i\pm\frac{1}{2}}$ are dissipative fluxes from (3.50) and D_τ is a diagonal matrix of nondimensional coefficients:

$$D_\tau = \begin{pmatrix} \tau_d & 0 & 0 \\ 0 & \tau_u & 0 \\ 0 & 0 & \tau_e \end{pmatrix}. \tag{3.52}$$

Coefficients τ in the matrix D_τ control how much dissipation is added to the scheme. Constant τ_d specifies the amount of artificial diffusion (mass flux) added to the continuity equation, τ_u specifies the amount of artificial viscosity added to the momentum equation, and τ_e specifies the amount of artificial energy flux added to the energy equation. By different choices of τ parameters, one can obtain schemes with qualitatively different properties. To simplify future reference to some specific choices of parameters, let us introduce the following naming convention: The scheme adding only artificial viscosity ($\tau_d = \tau_e = 0, \tau_u > 0$) will be referred to as LW+1, the scheme adding artificial viscosity and energy flux ($\tau_d = 0, \tau_e > 0, \tau_u > 0$) will be LW+2, and the scheme with all $\tau > 0$ (hence adding also artificial diffusion) will be referred to as LW+3. The most usual choice is LW+2 with $\tau_d = 0$, and $\tau_u = \tau_e \in \langle 1, 2 \rangle$.

Let us note that the LW+3 scheme requires a different definition of density, which violates GCL. The details can be found in [64].

To illustrate some differences between LW+1, LW+2, and LW+3 schemes and different values of τ, let us first define the 1D Noh problem.

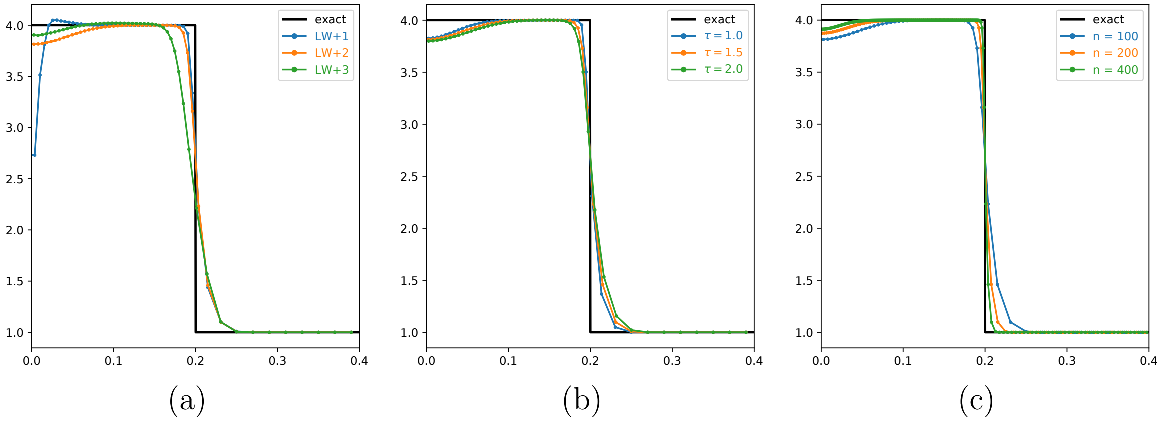
3.4.6 Noh problem

The Noh problem was firstly presented in [83] and deals with a symmetric infinitely strong shock wave reflected from the origin. The exact solution exists for one-, two- and three-dimensional geometries. The computational domain is a segment, square, or cube with the edge length of 2 and center in the origin, i.e. $\langle -1, 1\rangle$ in 1D, $\langle -1, 1\rangle^2$ in 2D, and $\langle -1, 1\rangle^3$ in 3D. In the beginning, the region is filled by an ideal gas characterized by $\gamma = 5/3$ with unit density, zero pressure (numerically we set 10^{-6}), and velocity field of unit magnitude pointing toward the origin. The exact solution is characterized by a single symmetric shock wave (e.g., circular in 2D or spherical in 3D) advancing with speed 1/3. The exact solution depends on constant $\nu = 1, 2, 3$ for one-, two- and three-dimensional geometry, respectively:

$$\varrho(t, \boldsymbol{x}) = \begin{cases} 4^\nu & \text{if } |\boldsymbol{x}| < t/3 \\ (1 + \frac{t}{|\boldsymbol{x}|})^{\nu - 1} & \text{otherwise} \end{cases}, \quad P(t, \boldsymbol{x}) = \begin{cases} 4^\nu/3 & \text{if } |\boldsymbol{x}| < t/3 \\ 0 & \text{otherwise} \end{cases},$$

$$\boldsymbol{U}(t, \boldsymbol{x}) = \begin{cases} \boldsymbol{0} & \text{if } |\boldsymbol{x}| < t/3 \\ -\frac{\boldsymbol{x}}{|\boldsymbol{x}|} & \text{otherwise} \end{cases}.$$

The final time $T = 0.6$ implies that the exact position of the shock front is located at $|\boldsymbol{x}| = 0.2$. The boundary conditions are given by prescribed velocity similar to the initial condition. Considering the symmetry, the problem can be computed just in the region $\langle 0, 1\rangle^\nu$ with reflecting boundary conditions on the axes. The results for different LW+n scheme variants and different values of τ can be seen in Figure 3.7.



(a) (b) (c)

Figure 3.7: 1D Noh problem; on coarse mesh (100 cells) computed using different LW+n schemes (a), on coarse mesh (100 cells) computed using LW+2 with different values of $\tau = \tau_u = \tau_e$ (b), convergence on different meshes using LW+2 with $\tau = 1.5$ (c).

As we can see from Figure 3.7(a), the result due to LW+1 suffers from a large density dip near the center of a shock state, which is caused by wall heating. This phenomenon is in great detail analyzed in [83]. For the mitigation wall-heating, W. F. Noh proposed to use among artificial viscosity also the artificial energy flux, which is the inspiration for the LW+2 scheme. As we can see, the density dip near the origin is significantly reduced when LW+2 and LW+3 schemes are used. The results due to LW+3 are slightly closer to the exact solution, but the introduction of artificial mass flux violates GCL, as discussed in [64]. Therefore, unless otherwise stated, we will, in the following, always present the results due to the LW+2 method.

As we can see in Figure 3.7(b), the value of τ controls the amount of added dissipation. Finally, in Figure 3.7(c) we present the convergence of LW+2 with $\tau = 1.5$ to the exact solution for subsequently finer meshes of $50 \times 50, 100 \times 100$ and 200×200 cells.

3.4.7 Treating material interfaces and contact discontinuities

Let us first address the contact discontinuity – a special wave where the pressure and velocity remain constant, while there is a discontinuity in the density. However, this feature of the solution is poorly resolved by the HLL scheme, which is built upon the assumption that the solution consists only of two waves. This fact results in large smearing of the contact when unlimited artificial dissipation is used. Due to the constant pressure and velocity on contact, the artificial energy flux can be turned off on the interface node lying between both states. For Lagrangian hydrodynamics, the contact discontinuity moves with the mesh and can be easily tracked as it remains at the same node as the initial interface between two different states. The influence of such a fix can be seen in Figure 3.10(c).

Although this simple switch improves the resolution of contact, it can cause problems when small CFL numbers are used, or the viscosity is limited in some way. In such situations, the computation can fail from a too-small time step, which is due to two small cells on the right of the interface at the beginning of the simulation when the three-wave structure is being formed. This flaw has motivated the introduction of the following special treatment of numerical flux at the material interface or contact discontinuities.

The first version of the *interface fix* proposed in [64] has been based on using the EUCCLHYD scheme [26] for computation of nodal velocity and pressure estimates from which the numerical fluxes on the contact discontinuity or material interface were computed.

The drawback of the interface fix based on the EUCCLHYD scheme is the additional computational complexity. Specifically, in more spatial dimensions, one has to implement also EUCCLHYD nodal solver [30]. Hence, in [67] a new treatment has been introduced. It is based on replacing Δt in the predictor of velocity at the interface point $f = i + \frac{1}{2}$ by local time step

$$\Delta T_f = \frac{\min\{\Delta x_i^n, \Delta x_{i+1}^n\}}{\max\{a_i, a_{i+1}\}},$$

where ΔT_f can be viewed as a maximum allowed time step of node (face) f. The predicted velocity at the node f is then given by:

$$U_f^{n+\frac{1}{2}} = \frac{U_i^n/\Delta x_i^n + U_{i+1}^n/\Delta x_{i+1}^n}{1/\Delta x_i^n + 1/\Delta x_{i+1}^n} + \frac{\Delta T_f}{m_i + m_{i+1}}(P_i^n - P_{i+1}^n). \tag{3.53}$$

From a different point of view, this means that the global time step Δt is locally replaced by ΔT_f, which speeds up the equilibration of velocity and pressure at the interface for Riemann problems. The effect of this modification makes the difference between the result presented in Fig. 3.10(d) (turned off) Fig. 3.10(e) (turned on). Although there is not a big difference in the contact itself, the oscillations at the end of the rarefaction wave are remarkably reduced when the interface fix is applied. This new modification performs comparably to that based on EUCCLHYD, although it is significantly less complicated.

3.4.8 Limiting artificial dissipation

When the proposed artificial viscosity is applied everywhere, it also smears the solution in continuous regions, which is shown in Figures 3.10(b-c), and which is usually not necessary and not even desirable. Ideally, the artificial dissipation should only smear the shock fronts while it should not take place in continuous solution to preserve higher accuracy where possible. We will now describe two strategies for limiting our artificial dissipation.

The easiest way of how to control the amount of dissipation being added to a particular region would be to simply switch it on or off. This approach can be easily applied in expanding regions. To detect whether the node $i+\frac{1}{2}$ is expanding, we simply compare the volumes of its adjacent cells in times t^n and t^{n+1} – i.e. whether both cell volumes Δx_i^{n+1} and Δx_{i+1}^{n+1} are greater than the old volumes Δx_i^n and Δx_{i+1}^n. The effect of this simple logical switch on the Sod problem is presented in Figures 3.10(d) (without *interface fix*) and 3.10(e) (with *interface fix*). As one can see, the resolution of the rarefaction wave is better when the dissipation is turned off in the expansion region.

Although this simple switch works reasonably well in 1D geometry, it can violate the symmetry in multiple dimensions for certain cases. Another problem is that it does not affect the regions of weak smooth compression. In such cases, artificial dissipation can be switched off too. To detect these situations, we use the limiter described in [5]. Let us first start by the definition of nodal velocity difference $\Delta u_{i+\frac{1}{2}} = u_{i+1} - u_i$ and nodal volume $\Delta x_{i+\frac{1}{2}} = (\Delta x_i + \Delta x_{i+1})/2$. Instead of the direct use of $\mathbf{D}^d_{i+\frac{1}{2}}$, we will use the limited term

$$\mathbf{D}^{\text{lim}}_{i+\frac{1}{2}} = (1 - \psi_{i+\frac{1}{2}})\mathbf{D}^d_{i+\frac{1}{2}} \tag{3.54}$$

where $\psi_{i+\frac{1}{2}}$ is a nodal limiter given by

$$\psi_{i+\frac{1}{2}} = \begin{cases} \max\{0, \min[(r_i + r_{i+1}), 2r_i, 2r_{i+1}, 1]\} & \text{if } \Delta u_{i+\frac{1}{2}} < 0 \\ 1 & \text{otherwise} \end{cases}, \tag{3.55}$$

and r_i, r_{i+1} are fractions of discrete spatial derivatives of velocity computed as:

$$r_{i+1} = \frac{\Delta u_{i+\frac{3}{2}}/\Delta x_{i+\frac{3}{2}}}{\Delta u_{i+\frac{1}{2}}/\Delta x_{i+\frac{1}{2}}}, \qquad r_i = \frac{\Delta u_{i-\frac{1}{2}}/\Delta x_{i-\frac{1}{2}}}{\Delta u_{i+\frac{1}{2}}/\Delta x_{i+\frac{1}{2}}}. \tag{3.56}$$

The stencil for the computation of limiter $\psi_{i+\frac{1}{2}}$ can be seen in Figure 3.8.

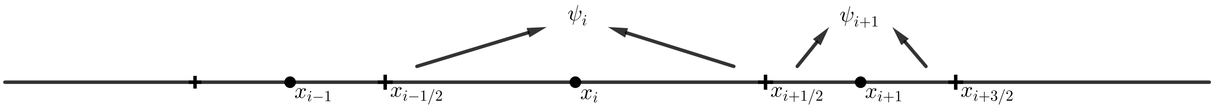
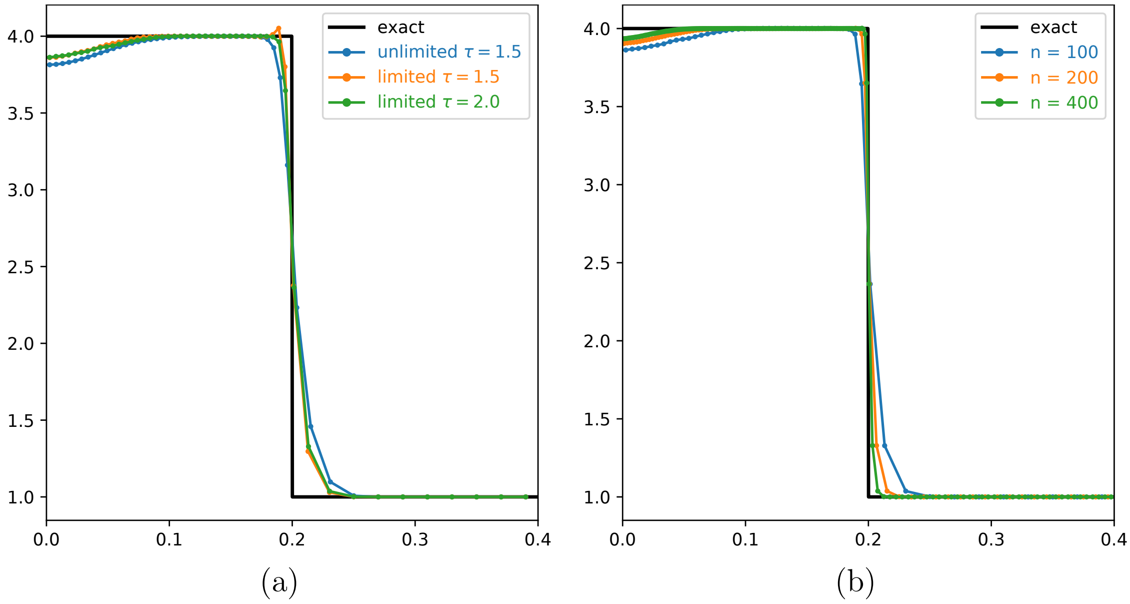


Figure 3.8: Stencil for 1D AV limiters.

The results computed with the limiter (3.55) can be found in Figure 3.10(f). Compared to the simple switching, the shock wave is better resolved and the density undershot at the tail of the rarefaction wave is no longer present.

We should note here that in some cases, the described limiting procedure may require increasing the value of the dissipative coefficient τ. We demonstrate this on the 1D Noh problem in Figure 3.9(a). When $\tau = 1.5$ is kept and the limiter is turned on, a small overshot near shock wave occurs, but it vanishes for $\tau = 2.0$. On the other hand, the shock wave computed using limited LW+2 with $\tau = 2.0$ is better resolved than the unlimited case with smaller τ. Figure 3.9(b) then shows the convergence for limited LW+2 with $\tau = 2.0$. Besides better capture of shock wave, the density dip near origin due to wall heating is also slightly reduced.



(a) (b)

Figure 3.9: The effect of dissipation limiting on Noh test computed on coarse mesh; limited and unlimited LW+2 with $\tau = 1.5$ and limited LW+2 with$\tau = 2.0$ (a); convergence for 1D Noh problem computed using limited LW+2 with $\tau = 2.0$ (b).

3.4.9 Recapitulation of the method

For the sake of simplicity, let us now shortly summarize a single time step of the LW+n method within which the solution is advanced from the old time level n to the new time level $n + 1$.

1. Knowing all cell data from time level n, we can compute a new time step Δt^n.

2. Next, the predicted nodal values $\boldsymbol{w}_p^{n+\frac{1}{2}}$ can be computed.
3. Nodal thermodynamical quantities (c_s,P,ε) are updated.
4. Knowing all the nodal data at $t^{n+\frac{1}{2}}$ the new cell state values $\boldsymbol{w}_c^{n+1}$ are computed.
5. Artificial dissipation and limiter terms are computed and applied to $\boldsymbol{w}^{n+1}$.
6. The nodal positions are advanced by Δt^{n+1} to time level t^{n+1} using nodal velocities $U_p^{n+\frac{1}{2}}$, geometry structures (cell centers and volumes) are updated.
7. Cell densities are recomputed from the new geometry using (3.7).
8. Cell thermodynamical quantities are updated.
9. Finally, we apply boundary conditions.

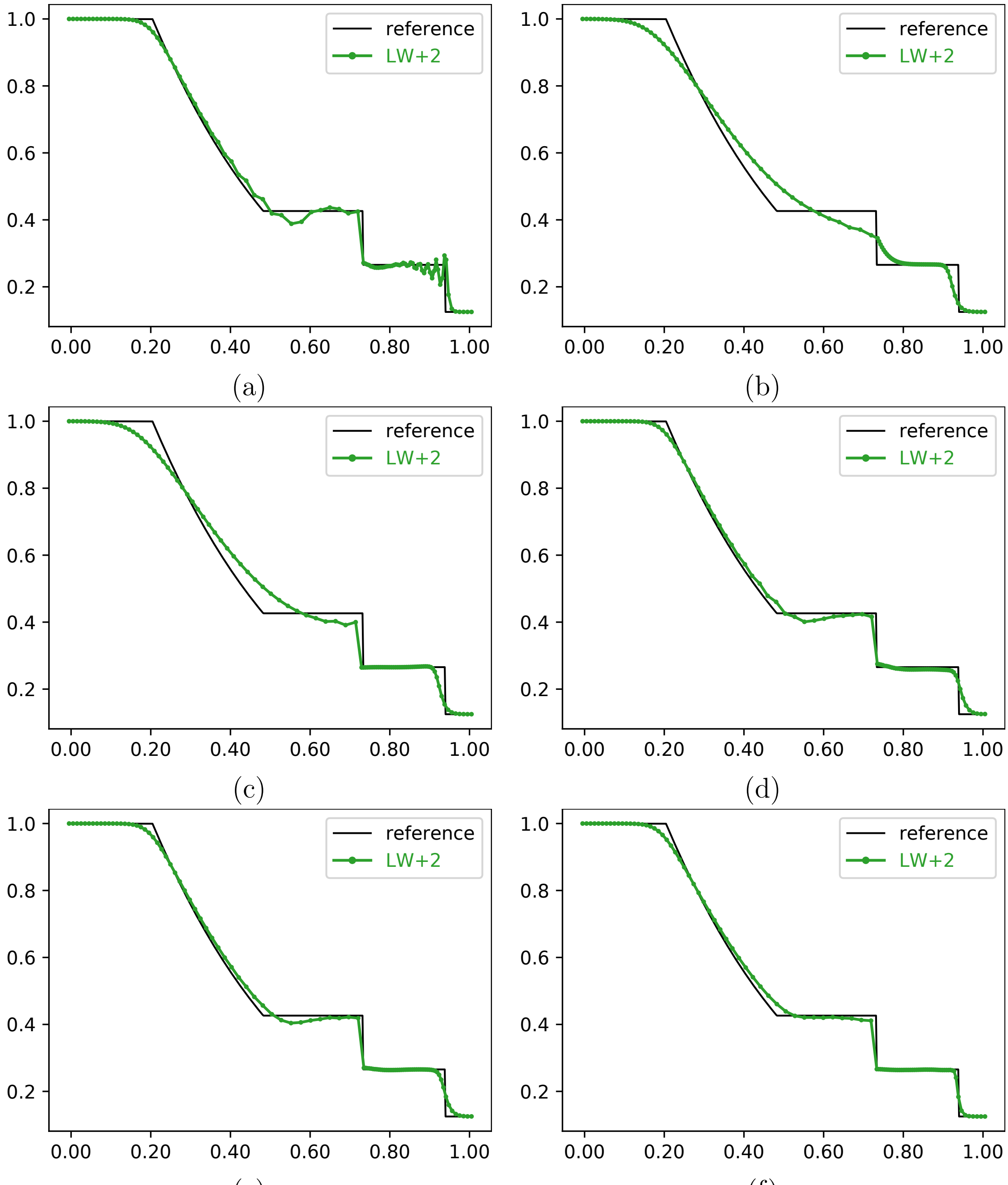
Let us note that steps 1-3 describe the predictor phase (substep), while steps 4-9 constitute the corrector phase. At the end of each time step, the mesh is checked for too large volume changes, and if no such problem is detected, the arrays storing new data are copied to those with the old values, and the whole sequence is repeated. If there is some problem, the whole step is restarted with a smaller time step $\Delta t^{lim} = \Delta t^{n+1}/2$. The algorithm stops if the final time is reached (success) or when the time step is too small (error).

3.5 Numerical results in 1D

3.5.1 Sod shock tube

This test was introduced by Sod in [84]. Let us assume a closed one-dimensional tube of a unit length separated by a membrane at position $x_0 = 0.5$ into two regions. Each region is initially filled with a perfect gas with $\gamma = 1.4$. The state in the left region is characterized by the density $\varrho_\mathrm{l} = 1$ and pressure $P_\mathrm{l} = 1$, the right state has the density $\varrho_\mathrm{r} = 0.125$ and pressure $P_\mathrm{r} = 0.1$. The gas is initially at rest, i.e. $u_\mathrm{l} = u_\mathrm{r} = 0$. At the time $t = 0$, the membrane is removed, and three waves are formed – the rarefaction wave on the left, the contact discontinuity in the middle, and the shock wave on the right. The final time is $T = 0.25$ and $C_\mathrm{CFL} = 0.4$. The 1D mesh is initially discretized into 100 equally spaced cells.

Similarly to [67], we present in Figure 3.10 some effects of individual ingredients described in the preceding two subsections. Figure 3.10(a) shows the results of the Sod problem computed using the pure LW scheme, where we can see the oscillations behind the shock wave and at the end of rarefaction. Figure 3.10(b) shows the result of the LW+2 scheme. As one can see, the spurious oscillations were successfully mitigated. On the other hand, the contact discontinuity is nearly completely smeared, while the rarefaction and shock wave is also not sharply resolved.



(a) (b) (c) (d) (e) (f)

Figure 3.10: Density for Sod problem with $\tau = 1.5$; 100 cells; different ingredients of the scheme: pure LW (a), LW+2 with dissipation (b), LW+2 with artificial energy flux off at interface (c), LW+2 with energy flux off at interface, and all dissipation turned off in expansion (d), LW+2 with energy flux off at interface with artificial viscosity, and energy flux turned off in expansion (e), LW+2 with energy flux off at interface, interface fix, limited dissipation with interface fix (f).

The comparison with the EUCCLHYD method [26] for the Sod shock tube problem is presented in Figure 3.11. The results due to both methods are comparable. EUCCLHYD resolves the head of the rarefaction wave better, while LW+2 performs better near the contact, which can be best seen in Fig. 3.11(d) showing the internal energy.

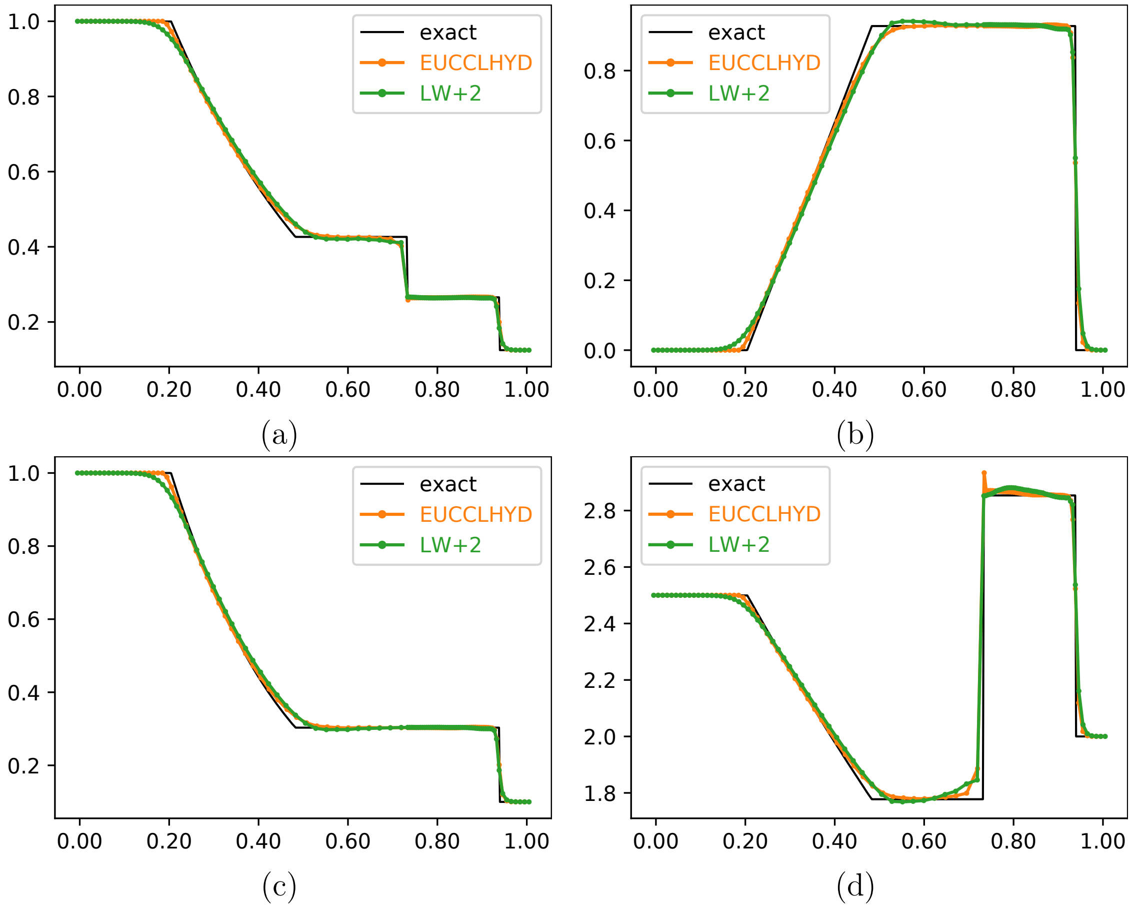


Figure 3.11: Comparison of EUCCLHYD method [26] (orange) with limited LW+2 with $\tau = 1.5$, and interface fix (green) for Sod problem computed on initially equidistant mesh with 100 cells. The density(a), the velocity(b), the pressure(c), and the internal energy(d).

3.5.2 Woodward-Collela blast wave

Another well-known test is the Woodward-Collela blast wave problem [85]. This problem involves multiple interactions between waves arising from two Riemann problems and reflecting boundaries. The problem is prescribed on 1D interval $\langle 0, 1\rangle$ and consists of three constant states of an ideal gas with $\gamma = 1.4$. The states are separated by $x_{\mathrm{l}} = 0.1$ and $x_{\mathrm{r}} = 0.9$. In all three states, the initial density is one, and the initial velocity is zero. The left pressure $P_{\mathrm{l}} = 1000$, the middle pressure $P_{\mathrm{m}} = 0.01$ and the right pressure $P_{\mathrm{r}} = 100$. The final time is $T = 0.038$. The results for the density computed by unlimited LW+2 with $\tau = 1.5$, *interface fix*, and zero artificial energy flux on both interfaces and $C_{\mathrm{CFL}} = 0.4$ on initially equally spaced mesh with 400 cells can be found in Figure 3.12. Since the exact solution for this problem is not known, we use a solution computed on a finer mesh with 2000 cells as a reference. Dissipation was not turned off in expansion for this case. The obtained result is reasonable compared with e.g. [86].

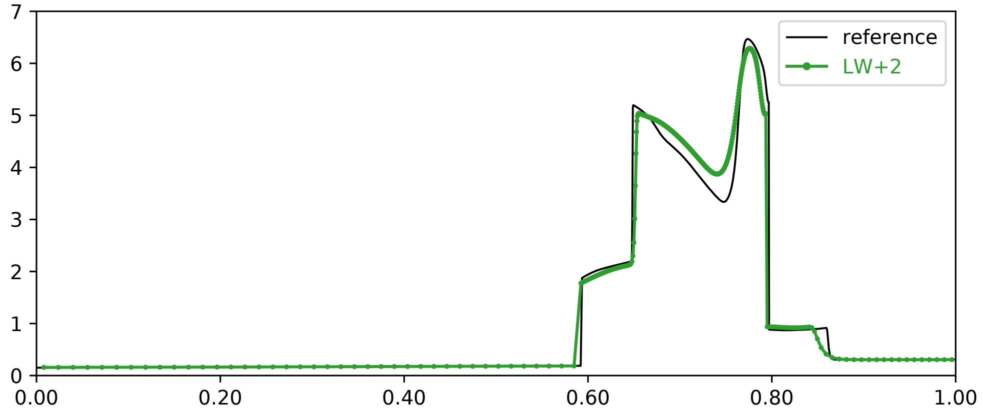


Figure 3.12: Density plot for Woodward-Colella problem at time $t = 0.038$ computed on 400 cells mesh.

Chapter 4

Lagrangian hydrodynamics in 2D Cartesian geometry

This chapter deals with 2D hydrodynamics in Cartesian geometry. Since in multiple dimensions, it is not possible to write Euler equations in mass coordinate similarly to the 1D case, we start by redefinition vector of specific conserved variables (3.17). Let m be mass of arbitrary moving control volume $V(t)$, both defined by (2.51), the vector of specific quantities can be then defined as:

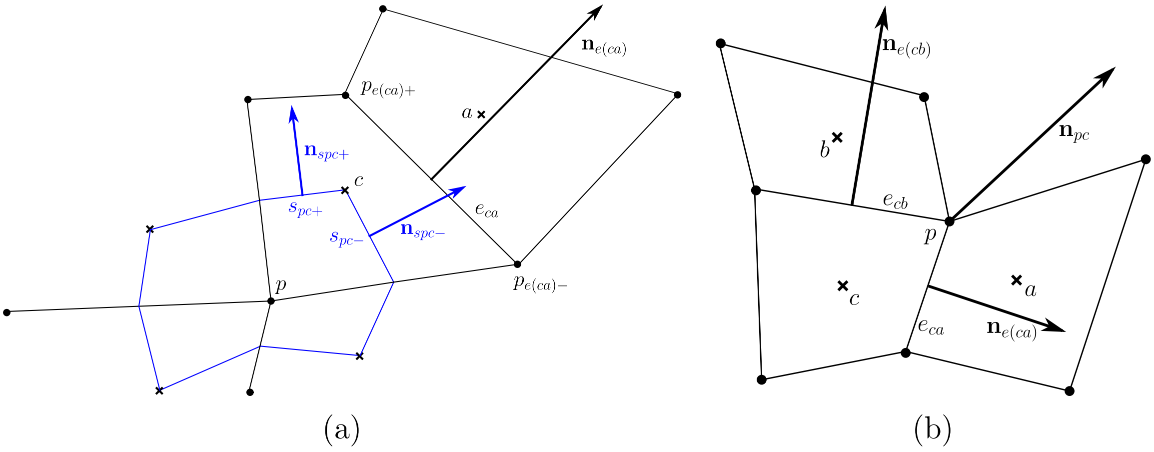
$$\boldsymbol{w} = \frac{1}{m}\boldsymbol{W} = (\eta, \boldsymbol{U}, E)^t. \tag{4.1}$$

The vector of specific quantities (4.1) represents dependent variables related to the vector of conserved variables (2.52). The numerical method will be presented in terms of $\boldsymbol{w}$.

4.1 Geometry

In the two-dimensional case, we consider arbitrarily polygonal computational domain, which is filled by nonoverlapping n-lateral cells without any gaps.

Each cell is formed by n points connected by straight lines. As one can see in Fig. 4.1(a), each cell is further divided into n subcells. Each subcell (subzone) is quadrilateral given by one node, one cell center, and two edge midpoints. Two edges of the subcell adjacents to the cell node are called *half-edges*, while the other two adjacents to the cell center are called *separators*. The mesh given by edges (plotted in Fig. 4.1(a) by black lines) will be further referred to as a *primary mesh*, while the mesh formed by separators (plotted by blue lines) will be called a *dual mesh*.



(a) (b)

Figure 4.1: pc notation in 2D geometry. Primary (black) and dual (blue) cell (a); Edges and corner vectors (b).

Similarly to [30], we adopt a pc notation [86]: each cell will be indexed by unique index c and each point by unique index p. Since every subzone is given by a single unique cell c and a single unique point p, it will be indexed by $pc = cp$. The edge between two adjacent cells c and a will be denoted $e_{ca} = e(ca)$. The separators of subcell pc are $s_{pc\pm}$. Let us define the set of points around the cell c as $p(c)$ and the set of cells around the point p as $c(p)$. Using these sets, we can define the sum of nodal values around cell c as $\sum_{p(c)}$ and similarly, the sum of cells around the point p as $\sum_{c(p)}$. Similarly, we can define $e(c)$ as the set of the edges of cell c and sum $\sum_{e(c)}$ going over the set $e(c)$.

The position $\boldsymbol{x}_c = (x_c, y_c)$ of cell center c is given by the simple arithmetic average of nodal positions:

$$\boldsymbol{x}_c = \frac{1}{n} \sum_{p(c)} \boldsymbol{x}_p. \tag{4.2}$$

Let V_{pc} be the volume of subcell pc and $m_{pc} = \varrho_c V_{pc}$ its mass. In the case of Cartesian geometry, subcell volume is given by:

$$V_{pc} = \frac{1}{2} \sum_{q(pc)} (y_{q+} - y_q)(x_q - x_{q+}), \tag{4.3}$$

Where $q+ = \mathrm{mod}(q+1, n)$ is the neighboring index, and $q(pc)$ is the set of nodes of subzone pc. Lagrangian hydrodynamics assumes a constant mass in a given control volume, i.e., the subzonal mass $m_{pc} = \varrho_c V_{pc}$ remains constant since initialized.

After sub-zonal volume and mass are known, we can define cell volume V_c, nodal volume V_p, cell mass m_c, and nodal mass m_p using the simple sums of adjacent subzones:

$$V_c = \sum_{p(c)} V_{pc}, \qquad V_p = \sum_{c(p)} V_{pc}, \qquad m_c = \sum_{p(c)} m_{pc}, \qquad m_p = \sum_{c(p)} m_{pc}. \tag{4.4}$$

Next, we can define the vector $\boldsymbol{n}_{e(ca)}$, which is the normal vector to the edge $e(ca)$ pointing outwards the cell c and having the same length as $e(ca)$. Corner vector $\boldsymbol{n}_{cp}$ adjacent to subzone pc is defined as the average of normal vectors of edges of cell c that are adjacent to point p:

$$\boldsymbol{n}_{cp} = \frac{1}{2}(\boldsymbol{n}_{e(ca)} + \boldsymbol{n}_{e(cb)}),$$

as shown in Figure 4.1(b). The vectors $\boldsymbol{n}^n_{spc\pm}$ are normals to the separators $s_{pc\pm}$ pointing outwards subzone pc having the length of $s_{pc\pm}$.

4.2 Lax-Wendroff Scheme for Finite Volumes

Let us assume that the fluid element V is a convex 2D polygon with n edges, $e(V)$ is the set of its edges, and $\boldsymbol{n}_e$ is an outward normal vector of the same length as the edge e pointing outwards the volume V. The equation (2.52) can be discretized using the following simple formula:

$$\frac{\boldsymbol{W}^{n+k}_V - \boldsymbol{W}^n_V}{k\Delta t} = \sum_{e(V)} \boldsymbol{F}^{n+k-\frac{1}{2}}_e \cdot \boldsymbol{n}^{n+k-\frac{1}{2}}_e, \tag{4.5}$$

where k is the time index shift and has value $k = \frac{1}{2}$ for predictor and $k = 1$ for corrector.

4.2.1 Lax-Friedrichs and Wendroff-White predictors

The LF predictor can be obtained by applying (4.5) with $k = \frac{1}{2}$ on dual cells p:

$$\boldsymbol{w}^{n+\frac{1}{2}}_p = \frac{\sum_{c(p)} m_{pc}\boldsymbol{w}^n_c}{\sum_{c(p)} m_{pc}} + \frac{\Delta t}{2}\sum_{c(p)} \boldsymbol{F}^n_c \cdot (\boldsymbol{n}^n_{spc+} + \boldsymbol{n}^n_{spc-}), \tag{4.6}$$

where the sum on the right-hand side goes over the separators $spc\pm$.

Similarly to 1D, one can also use the Wendroff-White predictor given by:

$$\boldsymbol{w}^{n+\frac{1}{2}}_p = \frac{\sum_{c(p)} \boldsymbol{w}^n_{pc}/V_{pc}}{\sum_{c(p)} 1/V_{pc}} + \frac{\Delta t}{2m_p}\sum_{c(p)} \boldsymbol{F}^n_c \cdot (\boldsymbol{n}^n_{spc+} + \boldsymbol{n}^n_{spc-}). \tag{4.7}$$

Let us here remark that on the rectangular meshes, the WW weighting is similar to bilinear interpolation. The computed nodal velocity estimates $\boldsymbol{U}^{n+\frac{1}{2}}_p$ are used to advance the mesh:

$$\boldsymbol{x}^{n+\frac{1}{2}}_p = \boldsymbol{x}^n_p + \frac{\Delta t}{2}\boldsymbol{U}^{n+\frac{1}{2}}_p \tag{4.8}$$

4.2.2 Lax-Wendroff corrector

The LW corrector can be obtained by applying (4.5) with $k = 1$ on primary cells c:

$$m_c\boldsymbol{w}^{n+1}_c = m_c\boldsymbol{w}^n_c + \Delta t\sum_{e(c)} \boldsymbol{F}^{n+\frac{1}{2}}_{e(ca)} \cdot \boldsymbol{n}^{n+\frac{1}{2}}_{e(ca)}, \tag{4.9}$$

where $e(c)$ is the set of edges of cell c, $\boldsymbol{n}^{n+\frac{1}{2}}_{e(ca)}$ is an outward vector normal to edge $e(ca)$ having the same size as the edge and $\boldsymbol{F}^{n+\frac{1}{2}}_{e(ca)}$ is flux over the edge computed from nodal state estimates $\boldsymbol{w}_p^{n+\frac{1}{2}}$ by

$$\boldsymbol{F}_{e(ca)}^{n+\frac{1}{2}} = \frac{\boldsymbol{F}_{p(ca)+}^{n+\frac{1}{2}} + \boldsymbol{F}_{p(ca)-}^{n+\frac{1}{2}}}{2}. \tag{4.10}$$

The corrector (4.9) can be rewritten using summation over the nodal values:

$$m_c \boldsymbol{w}_c^{n+1} = m_c \boldsymbol{w}_c^n + \Delta t \sum_{p(c)} \boldsymbol{F}_p^{n+\frac{1}{2}} \cdot \boldsymbol{n}_{pc}^{n+\frac{1}{2}}, \tag{4.11}$$

where

$$\boldsymbol{n}_{pc}^{n+\frac{1}{2}} = \frac{1}{2}\left(\boldsymbol{n}_{pc+}^{n+\frac{1}{2}} + \boldsymbol{n}_{pc-}^{n+\frac{1}{2}}\right)$$

is the corner outward vector and $\boldsymbol{n}_{pc+}^{n+\frac{1}{2}}, \boldsymbol{n}_{pc-}^{n+\frac{1}{2}}$ are normal vectors to edges adjacent to cell c and point p. After the corrector is performed, the mesh is again advanced by point velocities $\boldsymbol{U}_p^{n+\frac{1}{2}}$:

$$\boldsymbol{x}_p^{n+1} = \boldsymbol{x}_p^{n+\frac{1}{2}} + \frac{\Delta t}{2}\boldsymbol{U}_p^{n+\frac{1}{2}}. \tag{4.12}$$

4.2.3 GCL proof

Here we will prove that the LW scheme preserves Geometric Conservation Law (2.48). This proof is taken from [64]. Let us start by writing the corrector (4.9) only for a specific volume η, which discretizes the equation (2.48):

$$m_c \eta_c^{n+1} = m_c \eta_c^n + \frac{\Delta t}{2} \sum_{p(c)} \boldsymbol{U}_p^{n+\frac{1}{2}} \cdot \boldsymbol{n}_{pc}^{n+\frac{1}{2}}. \tag{4.13}$$

We want to prove that if the nodal positions are advanced using (4.8) and (4.12), then

$$V_c^{n+1} = V_c^n + \frac{\Delta t}{2} \sum_{p(c)} \boldsymbol{U}_p^{n+\frac{1}{2}} \cdot \boldsymbol{n}_{pc}^{n+\frac{1}{2}}. \tag{4.14}$$

The following proof of (4.14) holds for any arbitrary choice of nodal velocity $\boldsymbol{U}_p$. We will repeatedly use the following identity:

$$a_2 b_2 - a_1 b_1 = \frac{1}{2}\left[(a_1 + a_2)(b_2 - b_1) + (a_2 - a_1)(b_2 + b_1)\right]. \tag{4.15}$$

The cell volume (area) update from time level n to $n+1$ is given by:

$$\begin{aligned}
V_c^{n+1} - V_c^n &= \frac{1}{4}\left(\sum_{p(c)} \boldsymbol{x}_p^{n+1} \cdot \boldsymbol{n}_{pc}^{n+1} - \sum_{p(c)} \boldsymbol{x}_p^n \cdot \boldsymbol{n}_{pc}^n\right) \\
&= \frac{1}{8} \sum_{p(c)} \left[(\boldsymbol{x}_p^{n+1} - \boldsymbol{x}_p^n) \cdot (\boldsymbol{n}_{pc}^{n+1} + \boldsymbol{n}_{pc}^n) + (\boldsymbol{x}_p^{n+1} + \boldsymbol{x}_p^n) \cdot (\boldsymbol{n}_{pc}^{n+1} - \boldsymbol{n}_{pc}^n)\right] \qquad (4.16) \\
&= \frac{1}{4} \sum_{p(c)} \Delta t \boldsymbol{U}_p \cdot \boldsymbol{n}_{pc}^{n+\frac{1}{2}} + \frac{1}{2}(V_c^{n+1} - V_c^n) + \frac{1}{8} \sum_{p(c)} (\boldsymbol{x}_p^n \cdot \boldsymbol{n}_{pc}^{n+1} - \boldsymbol{x}_p^{n+1} \cdot \boldsymbol{n}_{pc}^n).
\end{aligned}$$

And so:

$$V_c^{n+1} - V_c^n = \frac{1}{2}\sum_{p(c)} \Delta t \boldsymbol{U}_p \cdot \boldsymbol{n}_{pc}^{n+\frac{1}{2}} + \frac{1}{4}A, \tag{4.17}$$

where

$$A = \sum_{p(c)} (\boldsymbol{x}_p^n \cdot \boldsymbol{n}_{pc}^{n+1} - \boldsymbol{x}_p^{n+1} \cdot \boldsymbol{n}_{pc}^n). \tag{4.18}$$

To prove (4.14), we have to show that A is zero. Now, we also have that A can be written as a sum over edges:

$$A = \sum_{e(c)} (\boldsymbol{x}_e^n \cdot \boldsymbol{n}_e^{n+1} - \boldsymbol{x}_e^{n+1} \cdot \boldsymbol{n}_e^n). \tag{4.19}$$

where $\boldsymbol{x}_e$ is the sum of the endpoint positions, and $\boldsymbol{n}_e$ is the edge normal. With and $\boldsymbol{x} = (x, y)$,

$$A = \sum_{e(c)} \left[x_e^n \Delta(y^{n+1})_e - y_e^n \Delta(x^{n+1})_e - x_e^{n+1}\Delta(y^n)_e + y_e^{n+1}\Delta(x^n)_e \right]. \tag{4.20}$$

The first and the last terms can be combined as

$$A_1 = \sum_{e(c)} [x_e^n \Delta(y^{n+1})_e + y_e^{n+1}\Delta(x^n)_e]. \tag{4.21}$$

From (4.15), A_1 is a collapsing sum,

$$A_1 = 2\sum_{e(c)} \Delta(y^{n+1}x^n)_e = 0. \tag{4.22}$$

A similar result holds for the middle two terms, thus proving that $A = 0$ and hence we have proved (4.14), which means that if we set

$$m_c \eta_c^0 = V_c^0$$

then for all n

$$m_c \eta_c^n = V_c^n,$$

which implies that (4.13) is irrelevant and

$$\varrho_c^{n+1} = m_c / V_c^{n+1}.$$

4.3 Artificial dissipation in 2D

The artificial dissipation in 2D is again based on HLL approximate Riemann solver, in this case, used on the cell edges. Assuming that the finite element V corresponds with the cell c, the Equation (2.52) can be discretized as:

$$m_c \boldsymbol{w}_c^{n+1} = m_c \boldsymbol{w}_c^n + \Delta t \sum_{e(ca)\in e(c)} \boldsymbol{f}_{e(ca)}^n |\boldsymbol{n}_{e(ca)}^n|, \tag{4.23}$$

where the projection to the edge normals has been moved into the fluxes

$$\boldsymbol{f}^n_{e(ca)} = \frac{\sigma_{ac}\boldsymbol{F}^n_c + \sigma_{ca}\boldsymbol{F}^n_a}{\sigma_{ac} + \sigma_{ca}} \cdot \frac{\boldsymbol{n}^n_{e(ca)}}{|\boldsymbol{n}^n_{e(ca)}|} + \frac{\sigma_{ac}\sigma_{ca}}{\sigma_{ac} + \sigma_{ca}}(\boldsymbol{w}^n_a - \boldsymbol{w}^n_c). \tag{4.24}$$

These fluxes can be obtained from the modified solution of the 1D Riemann problem on the symmetry normal $\boldsymbol{n}^n_{e(ca)}$ to the edge $e(ca)$. If we apply the HLL approximate Riemann solver (3.48) with the HLL signal speeds $-\sigma_{ca}$ and σ_{ac} in the Lagrangian frame moving with the edge $e(ca)$, we obtain (4.24). The $\boldsymbol{f}^n_{e(ca)}$ flux consists of two parts. The first part on the right-hand side of (4.24) represents the physical fluxes, while the second term represents numerical dissipation. Similarly to 1D, the Lagrangian LW+n scheme can be written as

$$m_c\boldsymbol{w}^{d,n+1}_c = m_c\boldsymbol{w}^n_c + \Delta t \sum_{e(c)} \left(\boldsymbol{F}^{n+\frac{1}{2}}_{e(ca)} \cdot \boldsymbol{n}^{n+\frac{1}{2}}_{e(ca)} + \mathcal{D}_\tau \cdot \boldsymbol{D}^d_{e(ca)} \right), \tag{4.25}$$

where $\boldsymbol{F}^{n+\frac{1}{2}}_{e(ca)}$ is the LW flux obtained from (4.10),

$$\mathcal{D}_\tau = \begin{pmatrix} \tau_d & 0 & 0 & 0 \\ 0 & \tau_u & 0 & 0 \\ 0 & 0 & \tau_u & 0 \\ 0 & 0 & 0 & \tau_e \end{pmatrix}, \tag{4.26}$$

and

$$\boldsymbol{D}^d_{e(ca)} = \frac{\sigma_{ac}\sigma_{ca}}{\sigma_{ac} + \sigma_{ca}}(\boldsymbol{w}^n_a - \boldsymbol{w}^n_c).$$

The modified acoustic impedances in 2D on edge $e(ca)$ write:

$$\sigma_{ac} = \varrho_c \left(c^s_c + \frac{|(\boldsymbol{U}_c - \boldsymbol{U}_a) \cdot \boldsymbol{n}_{e(ca)}|}{|\boldsymbol{n}_{e(ca)}|} \right), \qquad \sigma_{ca} = \varrho_a \left(c^s_a + \frac{|(\boldsymbol{U}_c - \boldsymbol{U}_a) \cdot \boldsymbol{n}_{e(ca)}|}{|\boldsymbol{n}_{e(ca)}|} \right). \tag{4.27}$$

Let us end the 2D scheme description with a remark that for $\boldsymbol{n}_{e(ca)} = (\pm 1, 0)^t$ the 2D scheme (4.25) becomes the 1D version (3.51).

4.3.1 Limiting artificial dissipation in multiple dimensions

Multidimensional artificial dissipation limiters are taken from [9]. Because LW+n dissipative fluxes are defined on the edges, their limiters should also be edge-based.

Let us start by the computation of the velocity divergence over the edge $e(ca)$. The control volume for the divergence calculation is quadrilateral given by cell centers a, c adjacent to the edge, and its endpoints $p-$ and $p+$. The velocity divergence over the edge e is given by:

$$\begin{aligned}(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{e(ca)} = \frac{1}{2V_{e(ca)}} & \left[(u_{p+} - u_{p-})(y_a - y_c) - (\bar{u}_a - \bar{u}_c)(y_{p+} - y_{p-}) \right. \\ & \left. - (v_{p+} - v_{p-})(x_a - x_c) + (\bar{v}_a - \bar{v}_c)(x_{p+} - x_{p-}) \right],\end{aligned} \tag{4.28}$$

where node velocities $u_{p\pm}, v_{p\pm}$ are components of nodal velocity estimate $\boldsymbol{U}_{p\pm}^{n+\frac{1}{2}}$ due to predictor (4.7), and cell values are simple averages of nodal values

$$\bar{\boldsymbol{U}}_c = (\bar{u}_c, \bar{v}_c)^t = \frac{1}{n} \sum_{p(c)} \boldsymbol{U}_p^{n+\frac{1}{2}},$$

$V_{e(ca)}$ is the volume of the edge $e(ca)$ control volume, computed using (4.3), x, y are components of node or cell center positions, and u, v are components of the velocity vector. Subscripts a, c stand for cell centers adjacent to the edge $f = e(ca)$, while $p-, p+$ denote the edge endpoints. The situation is illustrated in Figure 4.2.

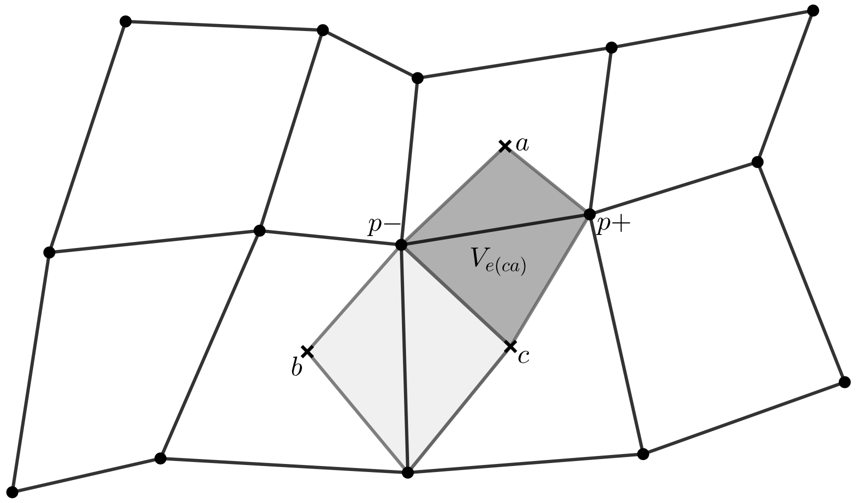


Figure 4.2: The control volumes used for the computation of velocity divergence.

When edge-based velocity divergences are known, we can advance to the computation of limiters. The limiter $\psi_{e(ca)}$ as proposed in [9] uses 4 neighbor edges (5 edges stencil) and is defined as the minimum of 1D limiters in logically horizontal and vertical directions (denoted by h and v subscripts respectively):

$$\psi_{e(ca)} = \min(\psi_h, \psi_v), \tag{4.29}$$

where

$$\psi_h = \max\left[0, \min((r_w + r_e)/2, 2r_w, 2r_e, 1)\right], \tag{4.30a}$$

$$\psi_v = \max\left[0, \min((r_n + r_s)/2, 2r_n, 2r_s, 1)\right]. \tag{4.30b}$$

The divergence fractions r are subscribed by compass points (n – North, w – West, s – South, e – East) with respect to their position logically relative to the central edge $f = e(ca)$ as depicted in Fig. 4.3 and write:

$$r_{\mathrm{n}} = \frac{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{n}}}{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{f}}}, \quad r_{\mathrm{w}} = \frac{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{w}}}{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{f}}}, \quad r_{\mathrm{s}} = \frac{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{s}}}{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{f}}}, \quad r_{\mathrm{e}} = \frac{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{e}}}{(\boldsymbol{\nabla} \cdot \boldsymbol{U})_{\mathrm{f}}}. \tag{4.31}$$

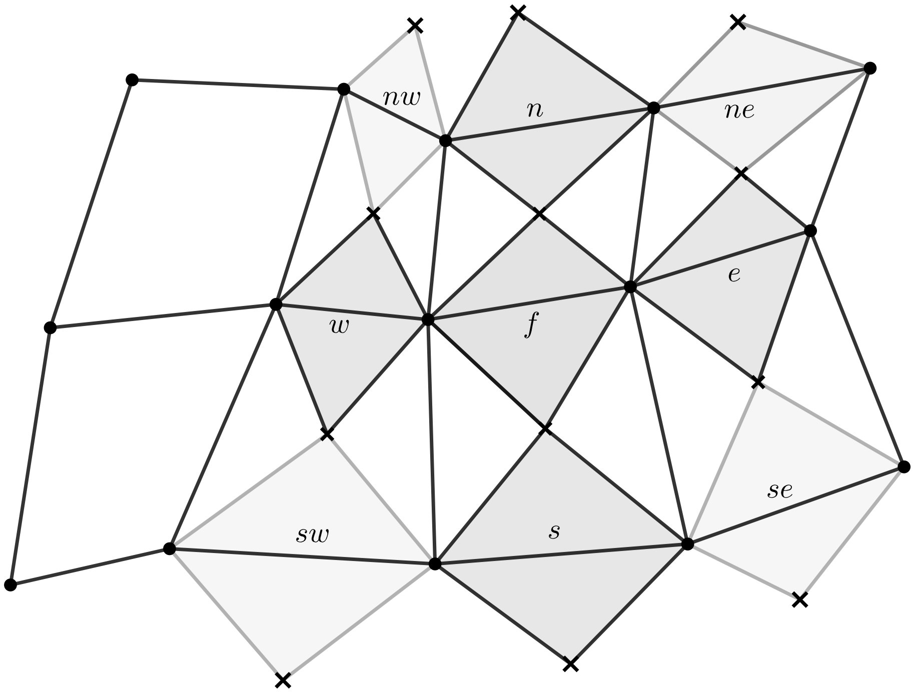


Figure 4.3: Illustration of cell indexing used for limiter computation. Dark dual cells are used for 5 edges stencil, the light cells are added to the 9 edges stencil.

To improve the symmetry on nonpolar meshes, we can modify the limiter (4.29) by extending the neighbors to 8 edges (9 edges stencil), taking into account also the ratios (defined again using compass points and illustrated in Figure 4.2) $r_{\mathrm{nw}}, r_{\mathrm{se}}$ and $r_{\mathrm{ne}}, r_{\mathrm{sw}}$ that form the logically diagonal limiters:

$$\psi_{13} = \max\left[0, \min((r_{\mathrm{ne}} + r_{\mathrm{sw}})/2, 2r_{\mathrm{ne}}, 2r_{\mathrm{sw}}, 1)\right], \tag{4.32a}$$

$$\psi_{24} = \max\left[0, \min((r_{\mathrm{nw}} + r_{\mathrm{se}})/2, 2r_{\mathrm{nw}}, 2r_{\mathrm{se}}, 1)\right]. \tag{4.32b}$$

And the final 9 edges stencil limiter will be

$$\psi_{e(ca)} = \min(\psi_h, \psi_v, \psi_{13}, \psi_{24}). \tag{4.33}$$

The comparison of different limiters will be presented in Subsection 4.5.2, dealing with the 2D Noh test.

4.4 Time step control

The time step Δt is controlled mainly by the Courant-Friedrichs-Levy (CFL) condition. At the time level n, the CFL time step is given by:

$$\Delta t^n_{\mathrm{CFL}} = C_{\mathrm{CFL}} \min_c \frac{l^n_c}{(c_s)^n_c}, \tag{4.34}$$

where $C_{\mathrm{CFL}} \in \langle 0, 1\rangle$ is Courant-Friedrichs-Lewy constant, l_c^n is the minimal edge of cell c, and $(c_s)_c^n$ is nothing else than sound speed defined by EOS (2.57).

To prevent volumes from changing too fast, the time step (4.34) is additionally restricted to increase no more than by 10% and set it to $\Delta t/2$ if the volume in any cell changes more than that:

$$\Delta t^{n+1} = \min(1.1\Delta t^n, t^{n+1}); \qquad \Delta t = t^{n+1}/2 \quad \text{if} \quad V^{n+1}/V^n \notin \langle 0.9, 1.1\rangle. \tag{4.35}$$

4.5 Numerical results in 2D Cartesian geometry

All the presented results were computed on quadrilateral (logically rectangular) meshes. In the case of polar mesh, the cells near the origin are treated as degenerated quadrilaterals.

4.5.1 Kidder

This test is taken from [29], [44], and deals with an isentropic compression of a shell filled by perfect gas characterized by $\gamma = 1 + \frac{2}{\nu}$, where $\nu = 1$ for planar, $\nu = 2$ for cylindrical, and $\nu = 3$ for spherical symmetry. Here we assume cylindrical symmetry and therefore set $\gamma = 2$. This problem has an analytical solution; hence it can be used to assess the accuracy order of the scheme. Let us define radius $R = \sqrt{x^2 + y^2}$ at time $t = 0$, the radius of a fluid particle at time $t > 0$ initially located at radius $R \in \langle R_{\mathrm{i}}, R_{\mathrm{o}}\rangle$ can be written as [29]:

$$\mathcal{R}(R, t) = h(t)R, \tag{4.36}$$

where

$$h(t) = \sqrt{1 - \left(\frac{t}{t_{\mathrm{f}}}\right)^2}$$

and t_{f} is the focusing time of the shell (i.e., the time when the shell collapses into itself) given by

$$t_{\mathrm{f}} = \sqrt{\frac{\gamma - 1}{2} \frac{R_{\mathrm{o}}^2 - R_{\mathrm{i}}^2}{a_{\mathrm{o}}^2 - a_{\mathrm{i}}^2}},$$

where

$$a = \sqrt{s\gamma\varrho^{\gamma-1}}$$

is an isentropic sound speed, and R_{i}, R_{o} are the initial inner and outer radii of the shell.

Let arbitrary physical quantity q in position R_{i} be noted q_{i} and similarly, $q(R_{\mathrm{o}}) = q_{\mathrm{o}}$. Let $\mathcal{R}(R, t) = h(t)R$ be the radius of a fluid particle at time $t > 0$ initially located at radius $R \in \langle R_{\mathrm{i}}, R_{\mathrm{o}}\rangle$. The self-similar analytical solution for density ϱ, radial velocity

mesh	$L_1(\varrho)$	$L_1(P)$	$L_1(U)$	NOC(ϱ)	NOC(P)	NOC(U)
45x25	5.9e-03	1.0e-02	3.9e-03	1.3	1.7	2.0
90x50	2.4e-03	3.1e-03	1.0e-03	1.6	1.7	1.8
180x100	7.9e-04	9.5e-04	2.9e-04	1.7	1.8	1.8
360x200	2.5e-04	2.7e-04	8.2e-05			

Table 4.1: Convergence rates for density, velocity, and pressure for Kidder problem.

mesh	$\mathcal{R}_\mathrm{i}$	$\mathcal{R}_\mathrm{o}$	$L_1(\mathcal{R}_\mathrm{i})$	$L_1(\mathcal{R}_\mathrm{o})$	NOC($\mathcal{R}_\mathrm{i}$)	NOC($\mathcal{R}_\mathrm{o}$)
45x25	0.4492	0.4994	1.1e-03	9.5e-04	2.0	2.0
90x50	0.4501	0.5001	2.6e-04	2.4e-04	2.0	2.0
180x100	0.4503	0.5003	6.7e-05	6.0e-05	2.0	2.0
360x200	0.4503	0.5003	1.7e-05	1.5e-05		

Table 4.2: Convergence rates for input radius $\mathcal{R}_\mathrm{i}$ and output radius $\mathcal{R}_\mathrm{o}$ for the Kidder problem.

component U, and pressure P has the following form:

$$\varrho(\mathcal{R}(R,t),t) = h(t)^{-\frac{2}{\gamma-1}}\varrho^0\left[\frac{\mathcal{R}(R,t)}{h(t)}\right] = h(t)^{-\frac{2}{\gamma-1}}\varrho^0(R), \tag{4.37}$$

$$U(\mathcal{R}(R,t),t) = \frac{\mathrm{d}}{\mathrm{d}t}h(t)\frac{\mathcal{R}(R,t)}{h(t)}, \tag{4.38}$$

$$P(\mathcal{R}(R,t),t) = h(t)^{-\frac{2\gamma}{\gamma-1}}P^0\left[\frac{\mathcal{R}(R,t)}{h(t)}\right] = h(t)^{-\frac{2\gamma}{\gamma-1}}P^0(R), \tag{4.39}$$

For any $R \in \langle R_\mathrm{i}, R_\mathrm{o}\rangle$ the initial density and pressure are:

$$\varrho^0(R) = \left(\frac{R_\mathrm{o}^2 - R^2}{R_\mathrm{o}^2 - R_\mathrm{i}^2}\varrho_\mathrm{i}^{\gamma-1} + \frac{R^2 - R_\mathrm{i}^2}{R_\mathrm{o}^2 - R_\mathrm{i}^2}\varrho_\mathrm{o}^{\gamma-1}\right)^{\frac{1}{\gamma-1}}, P^0(R) = s(\varrho^0(R))^\gamma.$$

The shell is initially at rest, so $U^0 = 0$. The boundary conditions are given by prescribed pressure:

$$P_\mathrm{i}(t) = P_\mathrm{i}^0 h(t)^{-\frac{2\gamma}{\gamma-1}}, \qquad P_\mathrm{o}(t) = P_\mathrm{o}^0 h(t)^{-\frac{2\gamma}{\gamma-1}}$$

The following constants are used (assuming cylindrical symmetry): $r_\mathrm{i} = 0.9, r_\mathrm{o} = 1.0, P_\mathrm{i} = 0.1, P_\mathrm{o} = 10, \varrho_\mathrm{o} = 0.01$, from which the following values can be computed: $\varrho_\mathrm{o} = 6.31 \times 10^{-4}, s = 2.15 \times 10^4$ and $t_\mathrm{f} = 7.26 \times 10^{-3}$. The final time was set to $t = (\sqrt{3}/2)t_\mathrm{f} = 6.29 \times 10^{-3}$, at which the outer radius $r_\mathrm{o} = 0.5$. The experimental convergence order for density, pressure, and radial velocity by pure LW method can be found in Table 4.1, while the convergence of inner and outer radius is presented in Table 4.2. Both tables verify the second order of accuracy for the scheme given by predictor (4.7) and corrector (4.11). Figure 4.4 then shows a pseudocolor plot of the density for a coarse mesh with 5 angular times 10 radial cells and finer mesh with 45 angular times 25 radial cells. For this test, Tables 4.1 and 4.2 show the numerical order of convergence by pure LW scheme slightly below 2. Figure 4.4 shows the reasonable performance of the method.

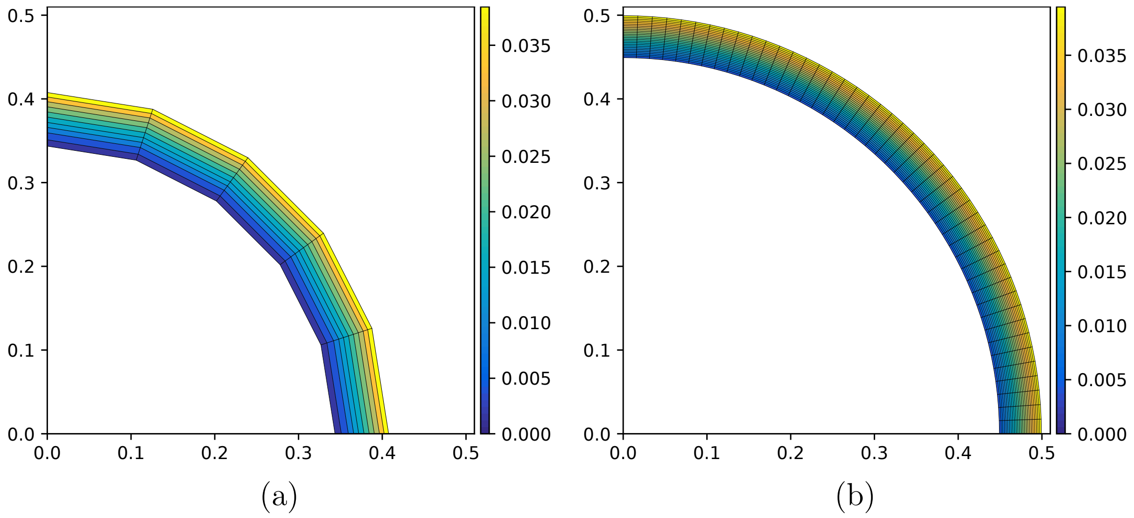


Figure 4.4: Pseudocolor plots of density for Kidder problem on equiangular mesh computed by pure LW; 5 × 10 cells (a), 45 × 25 cells (b).

4.5.2 Noh

Let us now present some results of the Noh test in 2D Cartesian geometry. For the sake of simplicity, we will now recall the initial conditions, exact solution, and scheme parameters specific for this geometry. For a detailed description, we refer the reader to Subsection 3.4.6. In 2D, the computational domain is initially equally spaced square $\langle 0,1\rangle^2$, or a single quarter of a circle with cells equally spaced in either radius or angle directions. For the initially equiangular mesh, the triangular cells adjacent to the origin are treated as degenerated quadrilaterals. The initial conditions are given by unit density, zero pressure (numerically we set $P_0 = 10^{-6}$), and unit velocity pointing towards the origin, that is for cell c given by

$$\boldsymbol{U}_c^0 = -\frac{\boldsymbol{x}_c^0}{|\boldsymbol{x}_c^0|} = \frac{1}{\sqrt{x_c^2 + y_c^2}}(-x_c, -y_c),$$

where $\boldsymbol{x}_c^0 = (x_c, y_c)$ is initial position of cell c center. The exact solution for $\gamma = 5/3$ is a circularly symmetric shock wave with

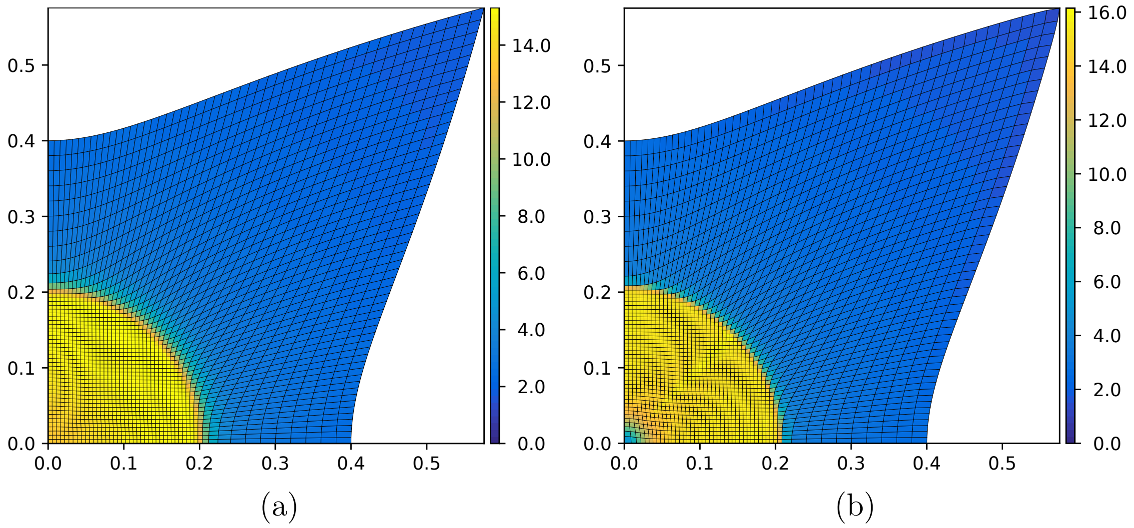
$$\varrho(t,\boldsymbol{x}) = \begin{cases} 16 & \text{if } |\boldsymbol{x}| < t/3 \\ 1 + \frac{t}{|\boldsymbol{x}|} & \text{otherwise} \end{cases}, \qquad P(t,\boldsymbol{x}) = \begin{cases} 16/3 & \text{if } |\boldsymbol{x}| < t/3 \\ 0 & \text{otherwise} \end{cases},$$

$$\boldsymbol{U}(t,\boldsymbol{x}) = \begin{cases} \boldsymbol{0} & \text{if } |\boldsymbol{x}| < t/3 \\ -\frac{\boldsymbol{x}}{|\boldsymbol{x}|} & \text{otherwise} \end{cases}.$$

We use $C_{\text{CFL}} = 0.2$. The final time is $T = 0.6$, which corresponds to the shock position at radius $R = \sqrt{x^2 + y^2} = 0.2$. Reflective boundaries are applied on the axes, while the exact velocity is imposed on the rest of the boundary.

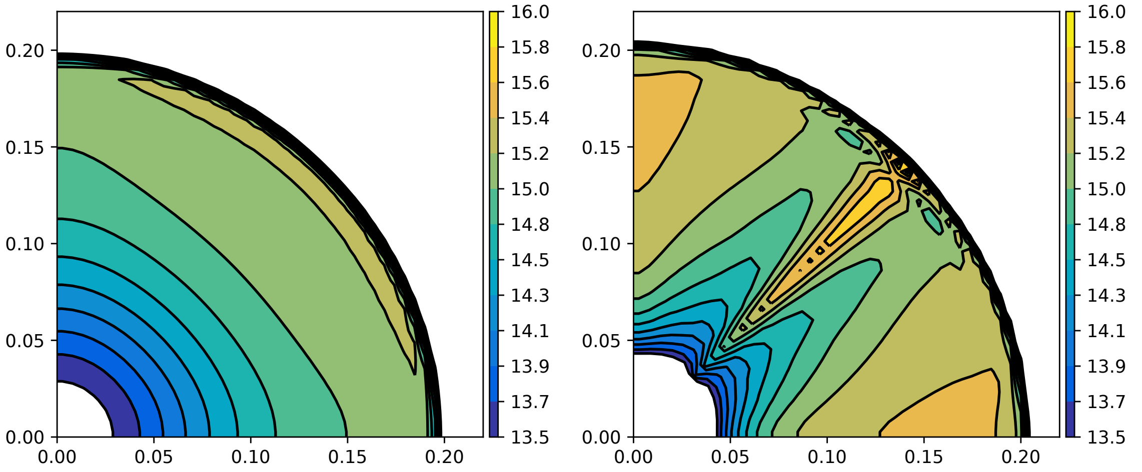
Let us start by comparison of LW+2 and staggered method with tensor artificial viscosity [9]. The pseudocolor plots computed on an initially orthogonal uniform coarse mesh of 50×50 cells can be seen in Figure 4.5. Figure 4.6 then shows the same comparison using the contour plot. As one can see, the unlimited LW+2 method clearly outperforms the Staggered scheme regarding the symmetry of the solution. The scatterplots for different mesh resolutions, namely 50×50, 100×100, and 200×200 are then presented in Figure 4.7(a). Figure 4.8 then shows the influence of different versions of unlimited LW+n schemes and different values of τ. The scatterplots for the Noh problem on the initially polar equiangular grid with mesh resolutions (angular × radial direction) 25×50, 50×100, and 100×200 are shown in Figure 4.7(b).

The symmetry of unlimited LW+2 is kept within the machine precision for the polar mesh, while it remains very nice for the initially rectangular case. The density dip near the origin is successfully mitigated by the artificial energy flux (i.e., LW+2 and LW+3 schemes) in comparison with the reference staggered method and LW+1.



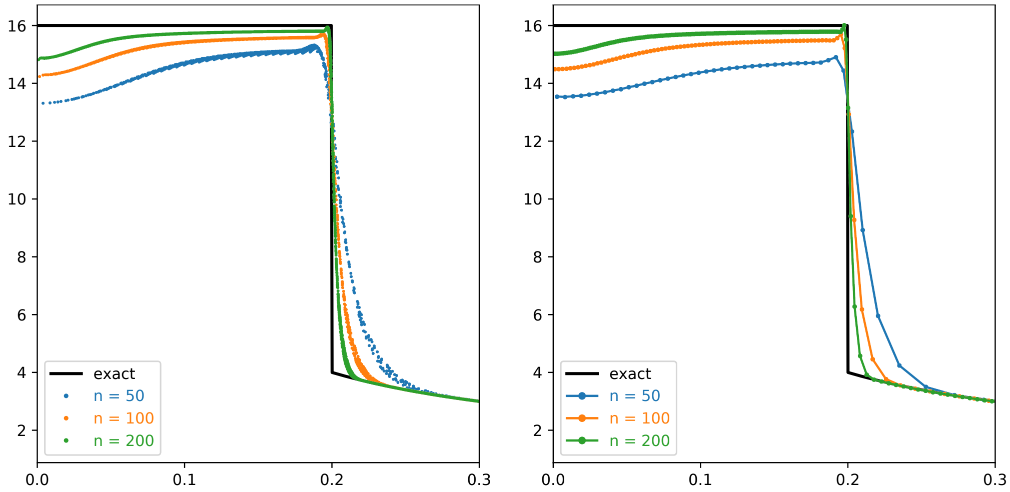
(a) (b)

Figure 4.5: Pseudocolor plots of Noh problem on initially rectangular 50×50 mesh. Unlimited LW+2 with $\tau = 1.5$(a) and Staggered scheme with tensor viscosity [9] (b).



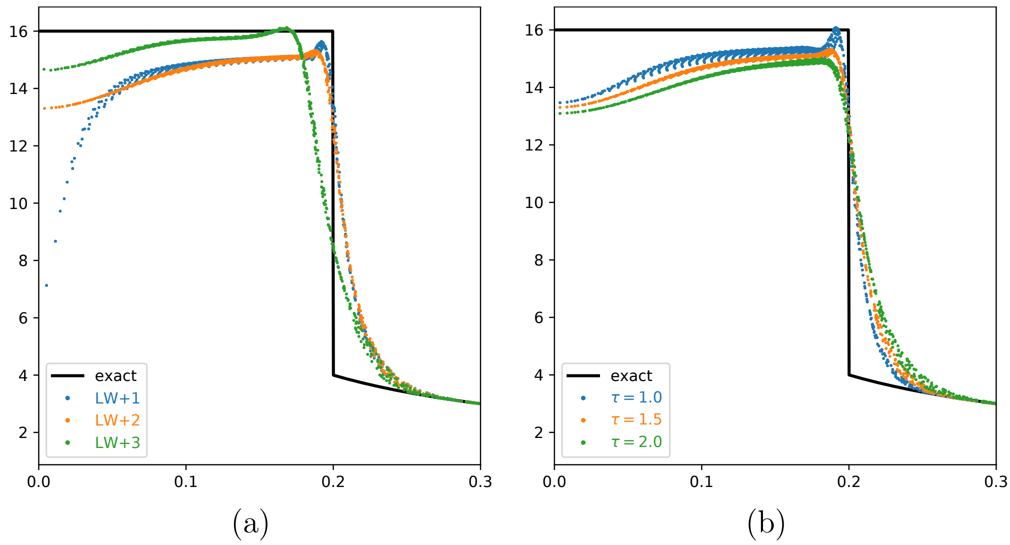
(a) (b)

Figure 4.6: Pseudocolor plots of Noh problem on initially rectangular 50 × 50 mesh. Unlimited LW+2 with $\tau = 1.5$ (a) and Staggered scheme with tensor viscosity [9] (b).



(a) (b)

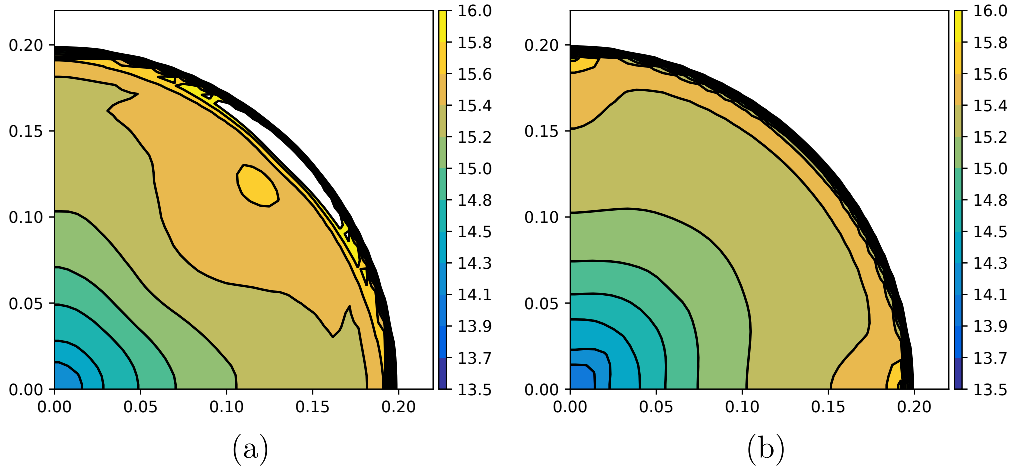
Figure 4.7: Scatter plots of Noh problem computed using unlimited LW+2 on different meshes. On initially rectangular 50 × 50 mesh. LW+2 with $\tau = 1.5$ (a), on polar meshes (b).



(a) (b)

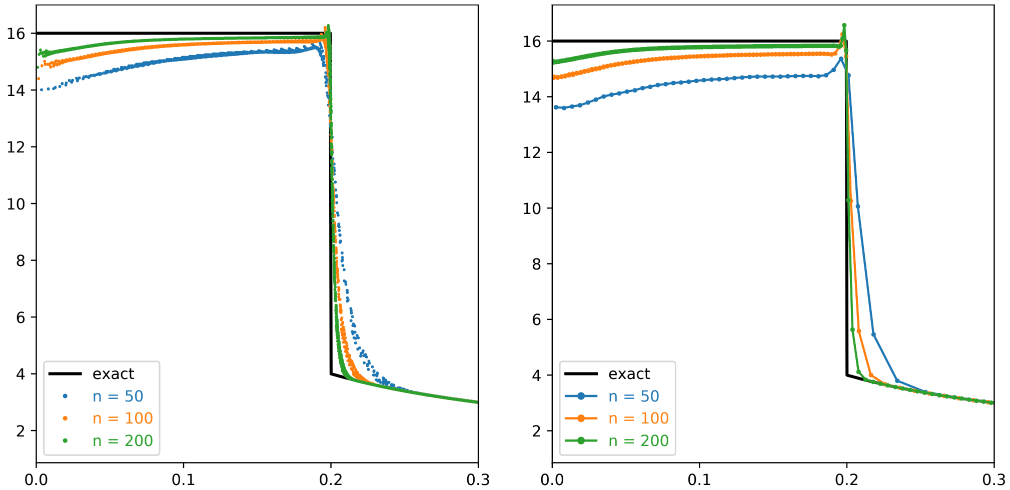
Figure 4.8: Scatter plots of Noh problem computed using different variants of unlimited LW+n scheme on coarse 50 × 50 initially rectangular mesh (a) and using LW+2 with different τ.

Figures 4.9 and 4.10 show results computed with the LW+2 scheme limited using 5 edge limiter (4.29), and 9 edges limiter (4.33). A symmetry on the initially rectangular mesh is worse than in unlimited cases presented in Fig. 4.6, 9 edges limiter (4.33) performs slightly better than the 5 edges one. On the other hand, the convergence due to LW+2 with 9 edges limiter for initially rectangular meshes presented in Figure 4.10 is regarding the height and shape of density plateau better than that due to unlimited LW+2. On polar meshes, the difference between unlimited and limited case is smaller; the overshot behind the shock is worse when a limiter is turned on.



(a) (b)

Figure 4.9: Contours for Noh on coarse initially rectangular 50×50 mesh computed using limited LW+2 with $\tau = 2.0$. The 5 edges stencil limiter (a); 9 edges stencil limiter(b).



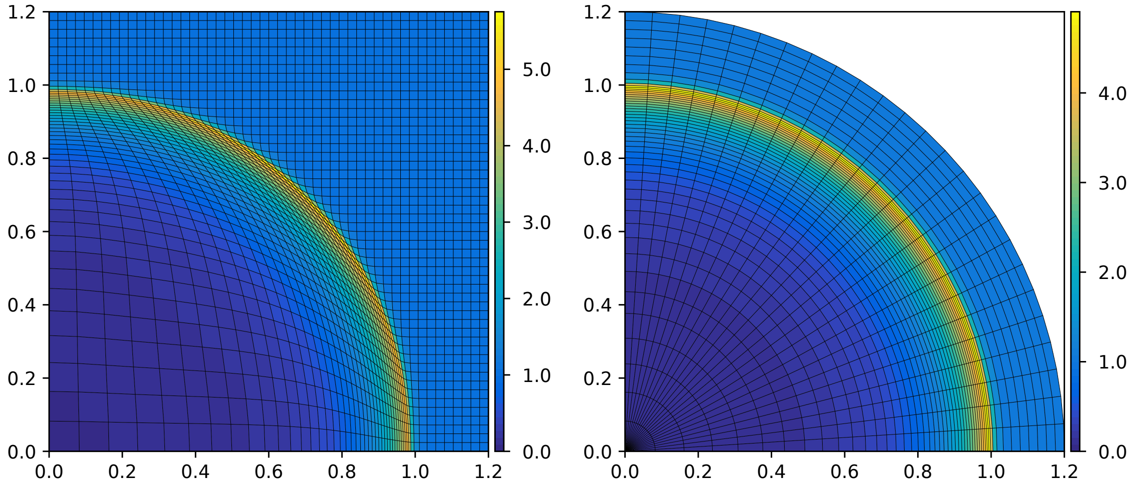
(a)

(b)

Figure 4.10: Scatter plots for Noh computed on different meshes using limited LW+2 with $\tau = 2.0$. On the initially rectangular grid with 9 point stencil (a); on the initially equiangular grid with limiter comparing only edges on a circle (b).

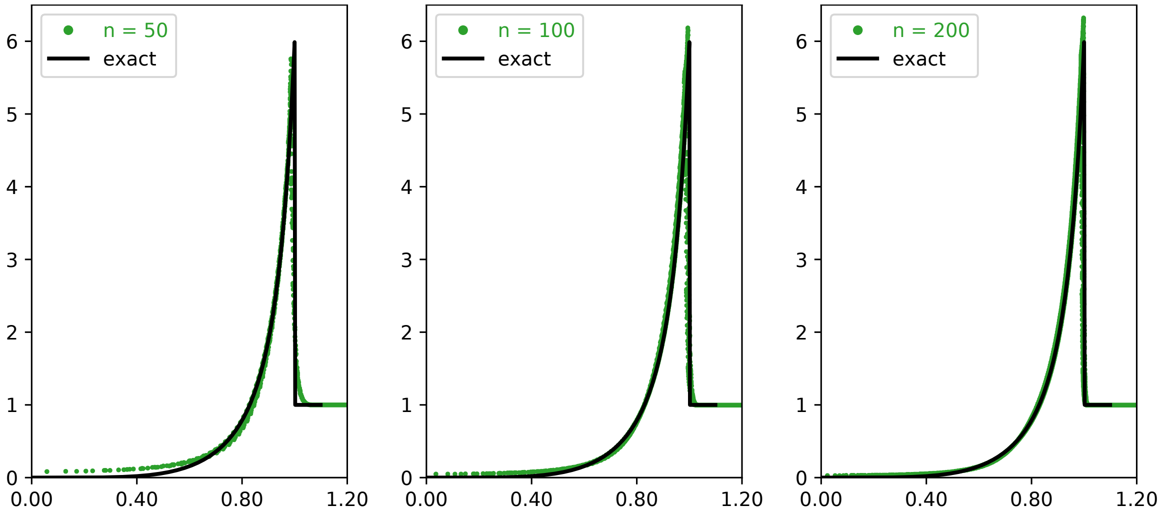
4.5.3 Sedov

Another well-known test case is the Sedov problem [87]. Similarly to the previous Noh test, we can again consider the symmetry and solve the problem only on a single quadrant. In 2D Cartesian geometry, the computational domain is a square $\langle 0, 1.2\rangle^2$ or a single quarter of circle defined by angle $\alpha \in \langle 0, \pi/2\rangle$ and radius $R = \sqrt{x^2+y^2} \in \langle 0, 1.2\rangle$. We consider ideal gas with $\gamma = 7/5$. The initial density is 1, and the pressure is 0 (numerically treated as 10^{-6}), and zero velocity. A single corner cell at the origin has the initial internal energy $\mathcal{E}_{\text{blast}} = 0.07783925$, which corresponds to the total blast energy $\mathcal{E}_{\text{blast}} = 0.311357$ for all four quadrants as in [88]. We use free boundary condition on the outer boundaries (North and East) and reflecting BC on axes. In Figure 4.11(a), we show the results for the initially rectangular coarse mesh of 50×50 cells, while in Figure 4.11(b), we present the result for the initially equiangular polar mesh with the resolution of 25×50 cells in angular and radial direction respectively. The convergence of numerical solutions computed on $50 \times 50, 100 \times 100$, and 200×200 initially rectangular cells to the exact solution computed by [88] is then shown in Figure 4.12, similarly to the convergence of solutions on initially equiangular polar meshes with $25 \times 50, 50 \times 100$ and 100×200 cells presented in Fig. 4.13. The effect of limiters can be seen in Figures 4.14, 4.15, and 4.16. Compared to unlimited cases, the density near the origin is closer to the analytical solution even though the τ was increased to a value of 2. The density overshot near the shockwave due to the 5-edge limiter is lower for the limited case, while due to the 9-edge limiter is comparable to the unlimited case. For polar meshes, we have limited only dissipation fluxes on circular edges to preserve the exact symmetry.



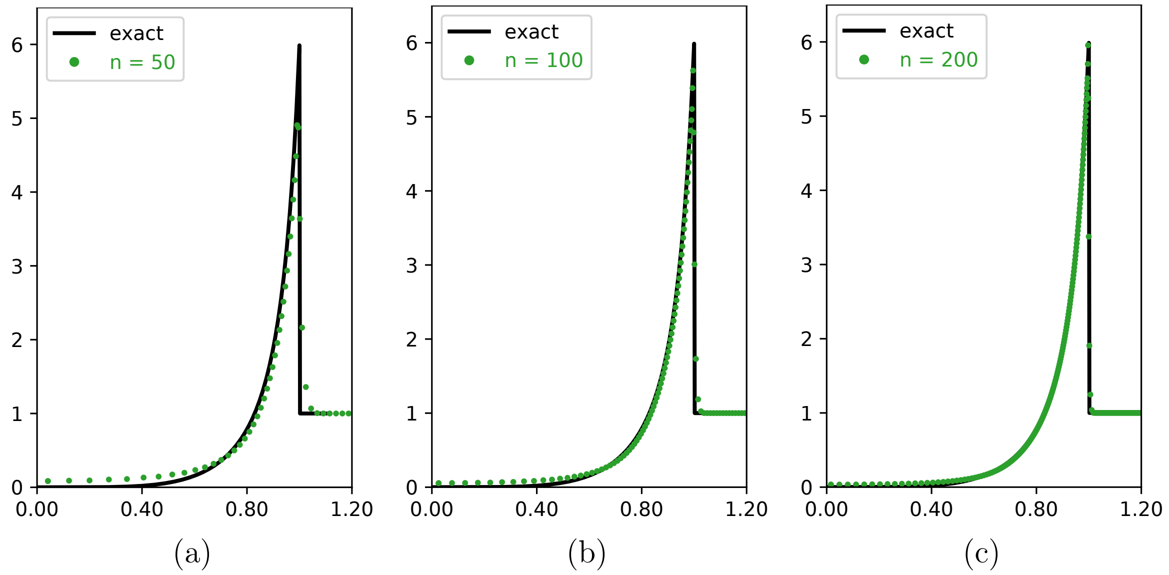
(a) (b)

Figure 4.11: Pseudocolor plots of the Sedov problem in Cartesian geometry computed using unlimited LW+2 with $\tau = 1.5$ on initially rectangular 50×50 grid (a); initially equiangular polar 25×50 grid (b).



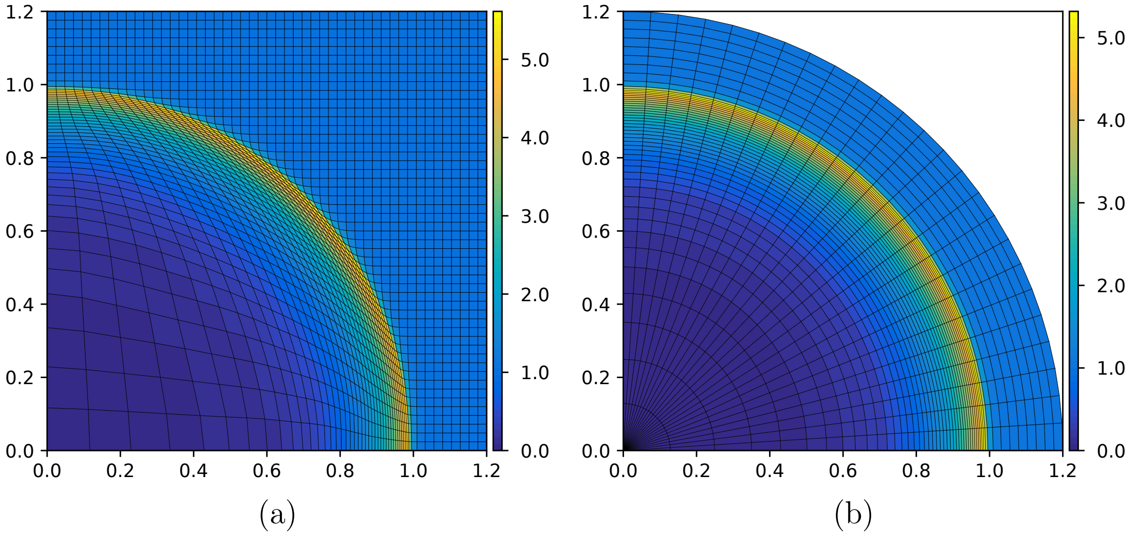
(a) (b) (c)

Figure 4.12: Scatter plots of the Sedov problem computed using unlimited LW+2 with $\tau = 1.5$ on initially rectangular meshes.



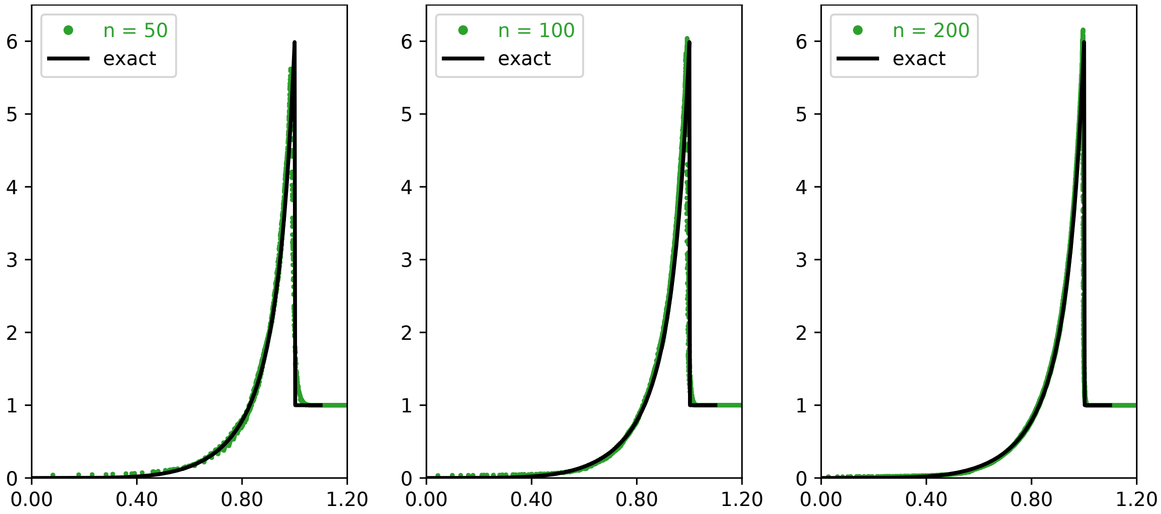
(a) (b) (c)

Figure 4.13: Scatter plots of the Sedov problem computed using unlimited LW+2 with $\tau = 1.5$ on polar meshes.



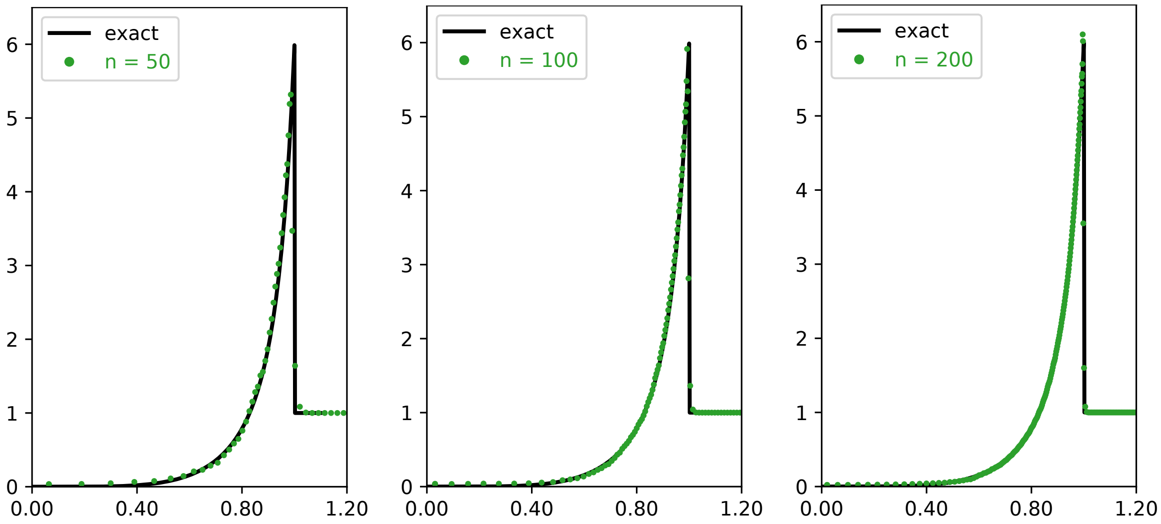
(a) (b)

Figure 4.14: Pseudocolor plots of the Sedov problem in Cartesian geometry computed using limited LW+2 with $\tau = 2.0$ on initially rectangular 50×50 grid (a); initially equiangular polar 25×50 grid (b).



(a) (b) (c)

Figure 4.15: Scatter plots of the Sedov problem computed using limited LW+2 with $\tau = 2.0$ on initially rectangular meshes.



(a) (b) (c)

Figure 4.16: Scatter plots of the Sedov problem computed using limited LW+2 with $\tau = 2.0$ on polar meshes.

4.5.4 Saltzman

This test is taken from [22], [38], [89], and it is basically the 2D piston-like problem on an initially skewed mesh. The computational region $\langle 0, 1\rangle \times \langle 0, 0.1\rangle$ is filled by an ideal gas with $\gamma = 1.4$ initially at rest. The initial density $\varrho^0 = 1$, the initial pressure is $P^0 = 10^{-6}$. The boundaries are treated as walls except for the left (West) edge, where we prescribe piston velocity $u_{\text{piston}} = 1$. The nodes of the equidistant rectangular mesh with 100×10

cells are initially displaced using the following formula:

$$x_{\text{skew}} = x + (0.1 - y)\sin(\pi x).$$

The analytical solution for this problem is characterized by a single shockwave moving from the left to the right with speed 4/3, so the East (right) boundary is reached at $t = 0.75$ when the shockwave is reflected. The density behind the unreflected shock (for $t < 0.75$) is 4, while the density behind the reflected shock is 10.

Similarly to [9], Figures 4.17 and 4.19 show the results in times $T = 0.7$ and $T = 0.85$ respectively computed by the unlimited LW+2 with $\tau = 1.5$. In Figures 4.18 and 4.20, one can see the result due to LW+2 with $\tau = 1.5$ and 9 edges limiter turned on. For this problem, the use of limiters reduces the density distortion, as visible mainly on the density scatter plots. The results are comparable to e.g. [9], [44].

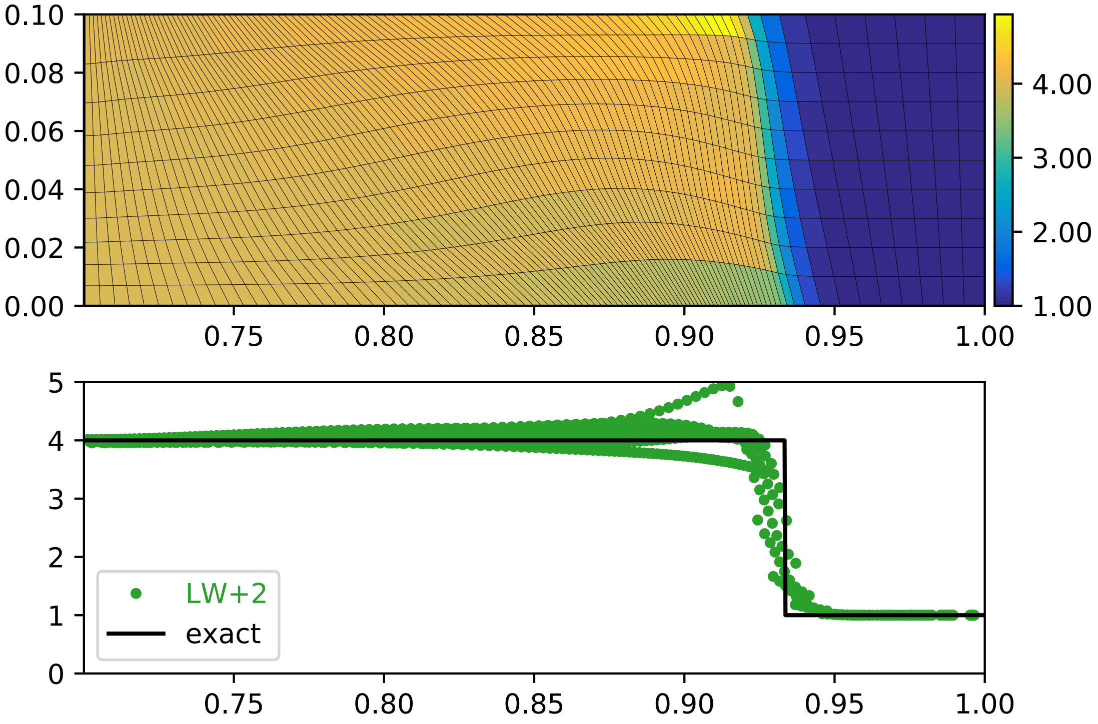


Figure 4.17: Density colormap and scatter plot for Saltzman problem computed by unlimited LW+2 with $\tau = 1.5$ in final time $T = 0.7$.

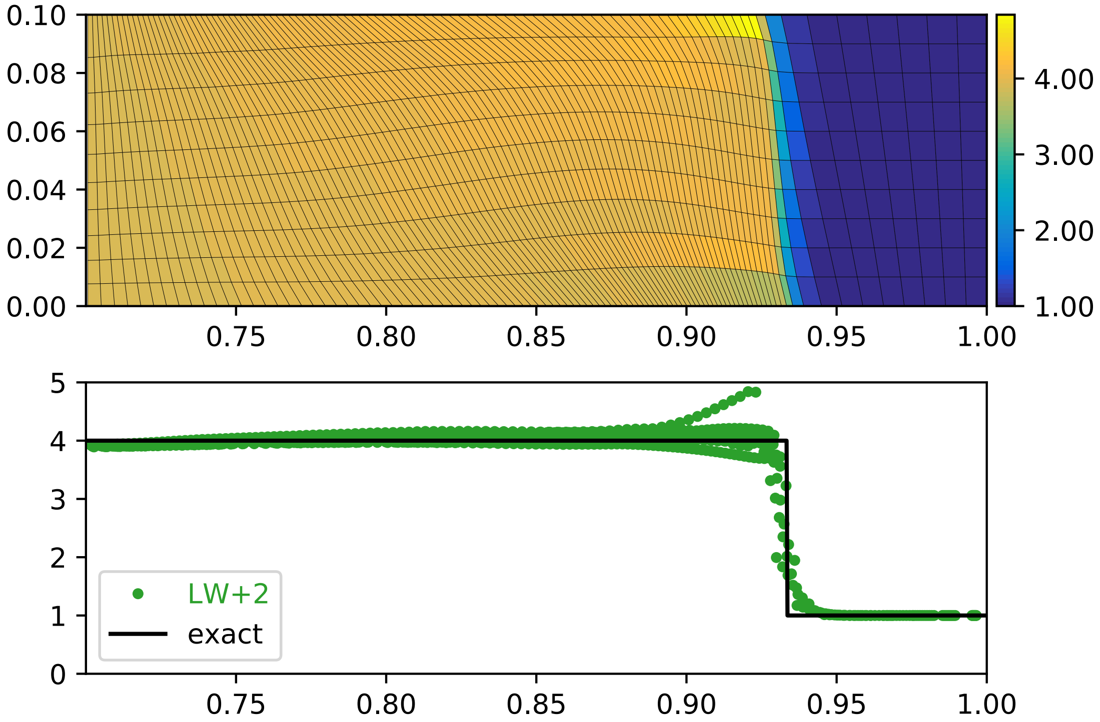


Figure 4.18: Density colormap and scatter plot for Saltzman problem computed by limited LW+2 with $\tau = 1.5$ in final time $T = 0.7$.

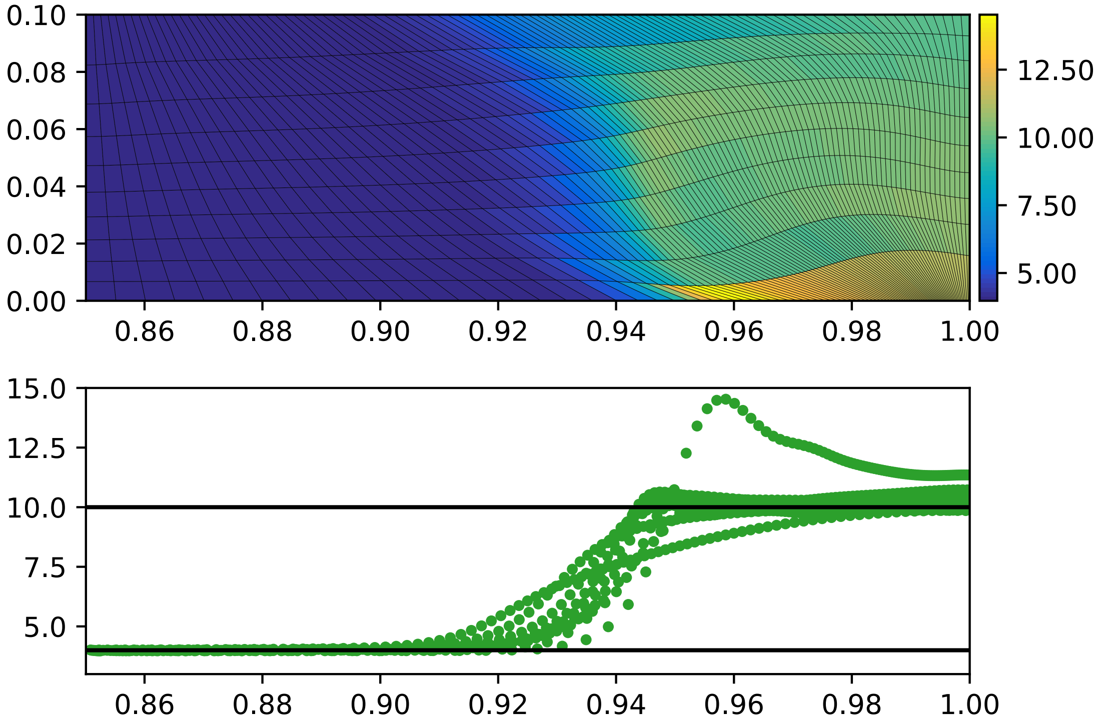


Figure 4.19: Density colormap and scatter plot for Saltzman problem computed by unlimited LW+2 with $\tau = 1.5$ in final time $T = 0.85$.

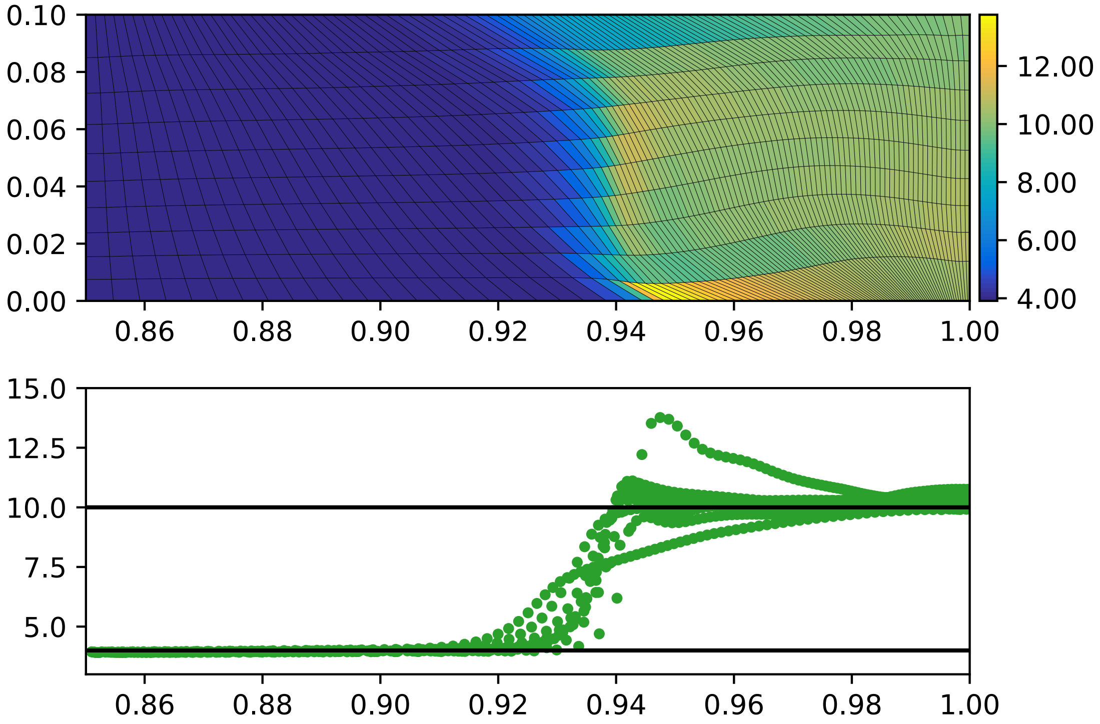


Figure 4.20: Density colormap, and scatter plot for Saltzman problem computed by limited LW+2 with $\tau = 1.5$ in final time $T = 0.85$.

4.5.5 Triple point

The Triple point test [90] is a two-dimensional Riemann problem that comprises three different states shown in Figure 4.21. The computational region $\langle 0, 7\rangle \times \langle 0, 3\rangle$ is initially divided into 70×30 square cells. Reflective boundary conditions are imposed everywhere. Several different variants with different choices of γ in each state exist; we consider the version from [40] with $\gamma = 1.4$ in all three states as proposed in [91]. The interface fix has been applied on all interface edges separating the individual states. The results at time $T = 2.7$ computed by the unlimited and limited LW+2 with $\tau = 1.5$ are shown in Figures 4.22 and 4.23 respectively. In this case, the use of limited use allowed a slightly better resolution of the interfaces between states and the vortex among the triple point. The shock wave generated by the left and bottom states is significantly better localized when the 9-edge limiter is used. The unlimited case is closer to the result due to the first-order scheme presented in [36], while the limited case is closer to that second order in the same place.

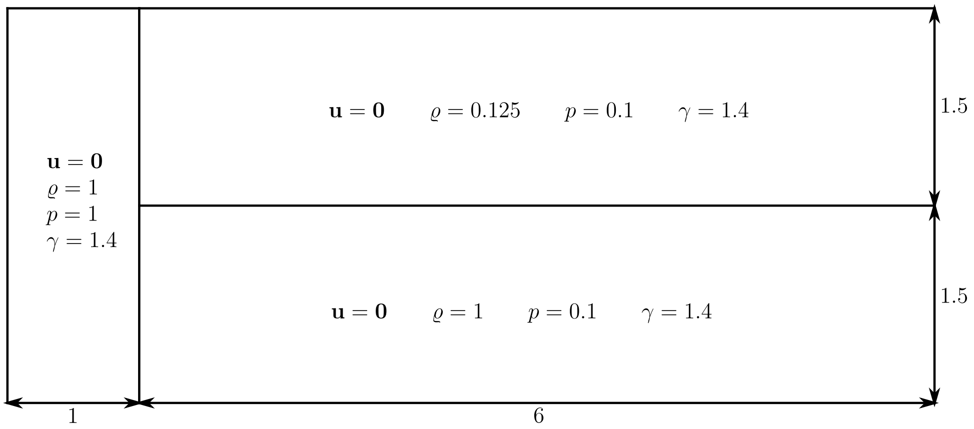


Figure 4.21: Initial conditions for thr Triple point problem.

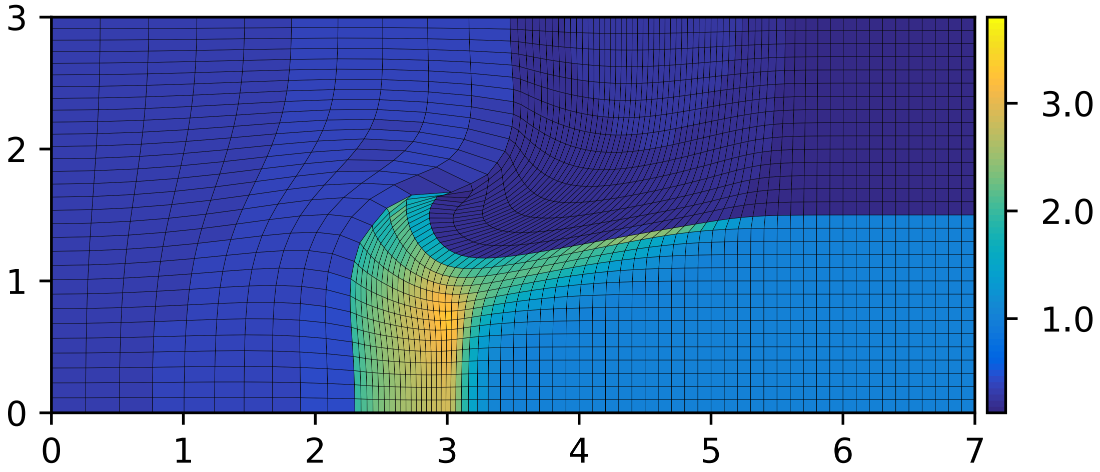


Figure 4.22: Density for the Triple point problem computed using unlimited LW+2 with $\tau = 1.5$ at time $t = 2.7$.

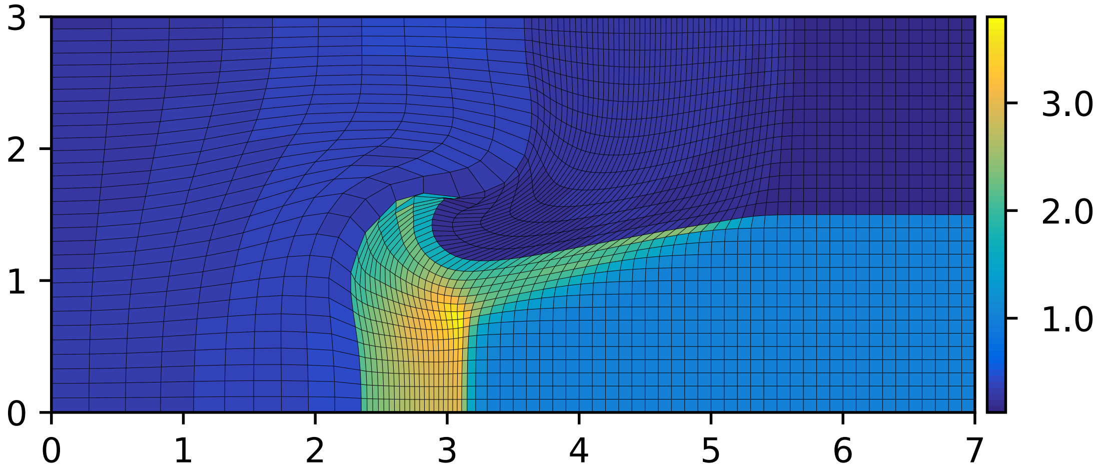


Figure 4.23: Density for the Triple point problem computed by limited LW+2 with $\tau = 1.5$ at time $t = 2.7$.

Chapter 5

Lagrangian hydrodynamics in 2D cylindrical coordinates

The primary motivation for the extension of the LW+n method to cylindrical geometry is the existence of cylindrically symmetric physical phenomena, mainly in the fields of inertial confinement fusion (ICF), astrophysics, or high energy density physics (HEDP).

In the literature, one can distinguish two families of numerical methods differing in a way how the extension to cylindrical geometry is performed. The first class of schemes is called *area weighed*, where one uses the Cartesian form of momentum equation in cylindrical coordinates system to preserve symmetry [3], [92]–[94]. Therefore, the integration is performed with respect to the area, not the cylindrical volume, which leads to the loss of energy conservation property [95]. This drawback was later corrected in [96] and more recently in [97].

The second family are the *control volume* (CV), or *volume weighted* methods [29], [98]–[102], where the momentum equation is integrated with respect to true cylindrical volume. Control volume methods conserve the total energy, but usually do not preserve spherical symmetry [96], which was later addressed in e.g. [101] by a different definition of average pressure in the geometrical source.

For a more detailed discussion of AW and CV differences, we refer the reader to e.g., [103], [104].

Let us now present the rz cylindrical geometry extension of our method, which also has these important properties. The governing equations will be again discretized in terms of Richtmyer form of Lax-Wendroff scheme with Wendroff-White weighting in the predictor, for which we will prove the symmetry-preserving property on equiangular meshes. This basic method will be complemented by the cylindrical version of our HLL-based artificial viscosity with a geometrical correction term, which construction has been inspired by [17].

5.1 Cylindrical 2D geometry

Let us briefly mention the differences from the Cartesian geometry. Similarly to the literature, the position vector components will be denoted by r, z, so that $\boldsymbol{x} = (r, z)$. Within the whole chapter, we will assume rotational symmetry around the z-axis, as shown in Figure 5.1.

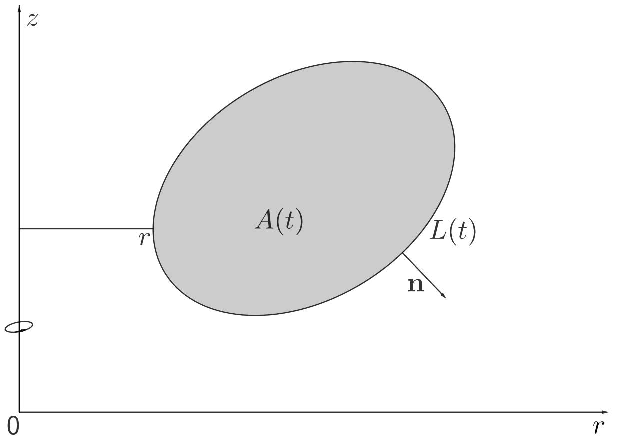


Figure 5.1: Illustration of cylindrical geometry and coordinates notation.

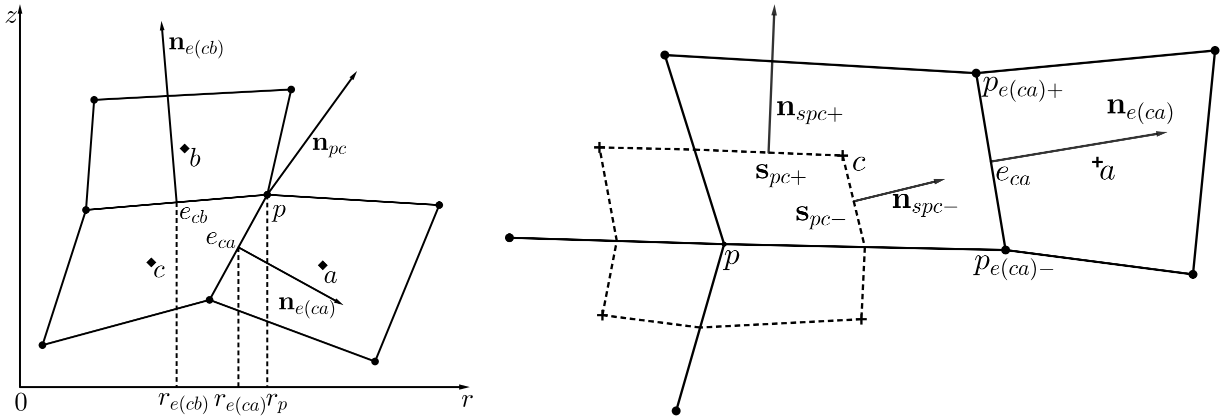


Figure 5.2: pc-notation in cylindrical geometry.

All the definitions remain similar as in the Cartesian geometry presented in Chapter 4.1 unless explicitly mentioned here. The sub-zonal volume is no longer defined by (4.3) but rather:

$$V_{pc} = \int_{\partial V} 1r\mathrm{d}r\mathrm{d}z. \tag{5.1}$$

Using the Green formula and assuming polygonal subzone pc

$$V_{pc} = \int_{\partial V} \frac{r^2}{2} = \sum_{e(c)} \int_e \frac{r^2}{2}\mathrm{d}z, \tag{5.2}$$

Using parametrization

$$r(s) = (1-s)r_q + sr_{q+1} \tag{5.3a}$$

$$z(s) = (1-s)z_q + sz_{q+1} \tag{5.3b}$$

$$\mathrm{d}z(s) = (z_{q+1} - z_q)\mathrm{d}s \tag{5.3c}$$

we can compute the integral over a single sub-zonal edge:

$$\begin{aligned}\int_e \frac{r^2}{2}\mathrm{d}z = \int_{(r,z)_q}^{(r,z)_{q+1}} \frac{r^2}{2}\mathrm{d}z = \int_0^1 \frac{1}{2}\left[(1-s)r_q + s r_{q+1}\right]^2 (z_{q+1}-z_q)\mathrm{d}s \\ = \frac{z_{q+1}-z_q}{2}\left[-\frac{(1-s)^3}{3}r_q^2 + 2\left(\frac{s^2}{2}-\frac{s^3}{3}\right) r_q r_{q+1} + \frac{s^3}{3}r_q^2\right]_0^1 \\ = \frac{1}{6}(z_{q+1}-z_q)(r_q^2 + r_q r_{q+1} + r_{q+1}^2),\end{aligned} \tag{5.4}$$

so finally:

$$V_{pc} = \sum_e \frac{1}{6}(z_{q+1}-z_q)(r_q^2 + r_q r_{q+1} + r_{q+1}^2). \tag{5.5}$$

Along with the cylindrical volume, let us also define the Cartesian area, which is given by (4.3):

$$A_{pc} = \frac{1}{2}\sum_e (y_{q+1}-y_q)(x_q - x_{q+1}), \qquad A_c = \sum_{p(c)} A_{pc}, \qquad A_p = \sum_{c(p)} A_{pc} \tag{5.6}$$

5.2 Differential operators in cylindrical geometry

Let us now recall the definitions of gradient and divergence operators needed for the derivation of governing equations in cylindrical coordinates. The gradient operator in 3D cylindrical coordinates applied to a scalar function $P(r,\varphi,z)$ is [105]:

$$\nabla(P(r,\varphi,z)) = \left(\frac{\partial P}{\partial r}, \frac{1}{r}\frac{\partial P}{\partial \varphi}, \frac{\partial P}{\partial z}\right)^t \tag{5.7}$$

And the cylindrical divergence applied to vector field $\boldsymbol{U}(r,\varphi,z) = (U_r, U_\varphi, U_z)^t$ is:

$$\nabla \cdot \boldsymbol{U}(r,\varphi,z) = \frac{1}{r}\frac{\partial(rU_r)}{\partial r} + \frac{1}{r}\frac{\partial U_\varphi}{\partial \varphi} + \frac{\partial U_z}{\partial z}. \tag{5.8}$$

Assuming cylindrical symmetry, neither P, nor $\boldsymbol{U}$ depends on the axial angle φ, and the partial derivatives with respect to φ are therefore zero, so they can be omitted. Due to that, in axisymmetric cylindrical coordinates:

$$\nabla(P(r,z)) = \left(\frac{\partial P}{\partial r}, \frac{\partial P}{\partial z}\right)^t. \tag{5.9}$$

Using the chain differentiation rule:

$$\nabla \cdot \boldsymbol{U}(r,z) = \frac{1}{r}\frac{\partial(rU_r)}{\partial r} + \frac{\partial U_z}{\partial z} = \frac{1}{r}\left(\frac{\partial(rU_r)}{\partial r} + \frac{\partial(rU_z)}{\partial z}\right) = \frac{\partial(U_r)}{\partial r} + \frac{U_r}{r} + \frac{\partial(U_z)}{\partial z}, \tag{5.10}$$

that is, applied to the velocity $\boldsymbol{U} = (U_r, U_z)^t = (u, v)^t$:

$$\nabla \cdot \boldsymbol{U}(r, z) = \frac{1}{r}\frac{\partial(ru)}{\partial r} + \frac{\partial v}{\partial z} = \frac{1}{r}\left(\frac{\partial(ru)}{\partial r} + \frac{\partial(rv)}{\partial z}\right) = \frac{\partial u}{\partial r} + \frac{u}{r} + \frac{\partial v}{\partial z}, \tag{5.11}$$

The last useful identity we will mention here is:

$$\nabla \cdot (P\boldsymbol{U}) = (\nabla P) \cdot \boldsymbol{U} + P(\nabla \cdot \boldsymbol{U}). \tag{5.12}$$

5.3 Governing equations in cylindrical geometry

In this section, we will rewrite the system (2.50) in cylindrical rz coordinates. By considering that $\mathrm{d}V = r\mathrm{d}A$:

$$\begin{aligned}\int_V \nabla \cdot \boldsymbol{U}\mathrm{d}V &= \int_A \frac{1}{r}\left(\frac{\partial(ru)}{\partial r} + \frac{\partial(rv)}{\partial z}\right) r\mathrm{d}A \\ &= \int_A \left(\frac{\partial(ru)}{\partial r} + \frac{\partial(rv)}{\partial z}\right) \mathrm{d}A \\ &= \int_L \boldsymbol{U} \cdot (r\boldsymbol{n})\mathrm{d}L,\end{aligned} \tag{5.13}$$

where the last equality is due to the Green theorem. By integration of (5.12) over the control volume V we have:

$$\int_V \boldsymbol{U} \cdot \nabla P\mathrm{d}V = \int_L P\boldsymbol{U} \cdot \boldsymbol{n}r\mathrm{d}L - \int_A P\nabla \cdot \boldsymbol{U}r\mathrm{d}A, \tag{5.14}$$

which, under the assumption that $\boldsymbol{U}$ is constant, leads to:

$$\int_V \nabla P\mathrm{d}V = \int_L P(r\boldsymbol{n})\mathrm{d}L - \boldsymbol{r}\int_A P\mathrm{d}A, \tag{5.15}$$

where $\boldsymbol{r} = (r, 0)^t$ remarks the fact, the geometrical source term appears only in the first component of momentum. Let us note that the last term of (5.15) is a source term due to cylindrical geometry. Assuming P constant over the volume V, the equation (5.15) leads into the following identity:

$$\int_L r\boldsymbol{n}\mathrm{d}L = \boldsymbol{r}\int_A \mathrm{d}A, \tag{5.16}$$

which will be used later in the scheme description. Now we can summarize the governing equations. The system (2.50) in cylindrical rz geometry writes [29]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V(t)} \varrho \mathrm{d}V = 0 \tag{5.17a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V(t)} \varrho \boldsymbol{U} \mathrm{d}V + \int_L P(r\boldsymbol{n})\mathrm{d}L = \boldsymbol{r}\int_A P\mathrm{d}A \tag{5.17b}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V(t)} \varrho E \mathrm{d}V + \int_L P\boldsymbol{U}\cdot(r\boldsymbol{n})\mathrm{d}L = 0 \tag{5.17c}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V(t)} \mathrm{d}V - \int_L \boldsymbol{U}\cdot(r\boldsymbol{n})\mathrm{d}L = 0 \tag{5.17d}$$

Using the identity (5.16), the momentum equation (5.17b) can be further rewritten in the form

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{V(t)} \varrho \boldsymbol{U} \mathrm{d}V + \int_L (P-\bar{P})(r\boldsymbol{n})\mathrm{d}L = \mathbf{0}, \tag{5.18}$$

where

$$\bar{P} = \frac{1}{A}\int_A P\mathrm{d}A \tag{5.19}$$

is the average of pressure P over the surface A. Knowing the governing equations, we can advance to the description of the numerical method.

5.4 Lax-Wendroff Scheme in 2D cylindrical coordinates (LWrz)

The predicted nodal velocity $\boldsymbol{U}_p^{n+\frac{1}{2}}$ is computed using a Cartesian form of the predictor with Wendroff-White averaging (4.7):

$$\boldsymbol{U}_p^{n+\frac{1}{2}} = \frac{\sum_{c(p)} \boldsymbol{U}_c/A_{cp}}{\sum_{c(p)} 1/A_{cp}} - \frac{\Delta t}{2\sum_{c(p)} \varrho_c^n A_{cp}^n}\sum_{c(p)} P_c^n(\boldsymbol{n}_{spc+}^n + \boldsymbol{n}_{spc-}^n), \tag{5.20}$$

while the predicted nodal energy $E_p^{n+\frac{1}{2}}$ can be written as:

$$E_p^{n+\frac{1}{2}} = \frac{\sum_{c(p)} E_c/A_{cp}}{\sum_{c(p)} 1/A_{cp}} - \frac{\Delta t}{2m_p}\sum_{c(p)} P_c^n\boldsymbol{U}_c^n\cdot(r_{spc+}^n\boldsymbol{n}_{spc+}^n + r_{spc-}^n\boldsymbol{n}_{spc-}^n), \tag{5.21}$$

where $\boldsymbol{n}_{spc\pm}$ stand for normal vectors to separators of the subzone pc and $r_{spc\pm}$ are r coordinates of separator centers as illustrated in Figure 5.2. After the predicted velocity $\boldsymbol{U}_p^{n+\frac{1}{2}}$ is known, the mesh is moved by $\Delta t/2$:

$$\boldsymbol{x}_p^{n+\frac{1}{2}} = \boldsymbol{x}_p^n + \frac{\Delta t}{2}\boldsymbol{U}_p^{n+\frac{1}{2}}, \tag{5.22}$$

The nodal specific volume and density are computed from geometry. The corrected velocity $\boldsymbol{U}_c^{n+1}$ given by discretization of (5.17b) writes:

$$\boldsymbol{U}_c^{n+1} = \boldsymbol{U}_c^n - \frac{\Delta t}{m_c} \sum_{e(c)} \left((P_e^{n+\frac{1}{2}} - \bar{P}_c^{n+\frac{1}{2}}) r_e^{n+\frac{1}{2}} \boldsymbol{n}_e^{n+\frac{1}{2}} \right), \tag{5.23}$$

where

$$\bar{P}_c^{n+\frac{1}{2}} = \frac{1}{N(c)} \sum_{p(c)} P_p^{n+\frac{1}{2}} \tag{5.24}$$

is the averaged cellular pressure at the time $t^{n+\frac{1}{2}}$ and $N(c)$ is the number of nodes. Due to

$$\sum_{e(c)} r_e (\boldsymbol{n}_e)_z = 0 \qquad \text{and} \qquad \sum_{e(c)} r_e (\boldsymbol{n}_e)_r = A_c, \tag{5.25}$$

the velocity corrector (5.23) can be rewritten into the following two separate equations for components u, v of velocity $\boldsymbol{U}$:

$$\begin{aligned} u_c^{n+1} &= u_c^n - \frac{\Delta t}{m_c} \left[\sum_{e(c)} r_e^{n+\frac{1}{2}} P_e^{n+\frac{1}{2}} (\boldsymbol{n}_e^{n+\frac{1}{2}})_r - \bar{P}_c^{n+\frac{1}{2}} A_c \right] \\ v_c^{n+1} &= v_c^n - \frac{\Delta t}{m_c} \sum_{e(c)} r_e^{n+\frac{1}{2}} P_e^{n+\frac{1}{2}} (\boldsymbol{n}_e^{n+\frac{1}{2}})_z \end{aligned} \tag{5.26}$$

The new energy is computed using

$$E_c^{n+1} = E_c^n - \frac{\Delta t}{m_c} \sum_{e(c)} (r_e P_e \boldsymbol{U}_e \cdot \boldsymbol{n}_e)^{n+\frac{1}{2}}. \tag{5.27}$$

When the new cellular velocities $\boldsymbol{U}_c^{n+1}$ and energies E_c^{n+1} are known, the mesh is moved again:

$$\boldsymbol{x}_p^{n+1} = \boldsymbol{x}_p^{n+\frac{1}{2}} + \frac{\Delta t}{2} \boldsymbol{U}_p^{n+\frac{1}{2}} \tag{5.28}$$

and the new cell-specific volume and density are computed from geometry. Finally, we can update the cell pressures and sound speeds using (2.55) and (2.57) respectively.

5.4.1 GCL Compatibility – using Simpson's rule

Since volume formulas in Cartesian and cylindrical coordinates differ, the proof 4.2.3 is no longer valid for cylindrical geometry. Let us now study the conditions under which GCL is satisfied in 2D cylindrical geometry. We will start from the equation (5.17d). Assuming the polygonal cell, the volume difference for $\Delta t = t^{n+1} - t^n$ can be rewritten as:

$$V_c^{n+1} - V_c^n = \int_{t^n}^{t^{n+1}} \frac{\mathrm{d}V_c}{\mathrm{d}t} \mathrm{d}t = \int_{t^n}^{t^{n+1}} \sum_{e(c)} F_{ec}^{\nu} \mathrm{d}t, \tag{5.29}$$

where

$$F_{ec}^{\nu} = \int_e (r\boldsymbol{u}_e) \cdot \boldsymbol{n}_e \mathrm{d}s. \tag{5.30}$$

Let c be a polygonal cell with nodes p moving with the velocities $\boldsymbol{U}_p = (u_p, v_p)$. Its volume V_c is then a function of nodal positions $\boldsymbol{x}_p = (r_p, z_p)$:

$$V_c^{n+1} - V_c^n = \int_{t^n}^{t^{n+1}} \frac{\mathrm{d}V_c}{\mathrm{d}t}\mathrm{d}t = \int_{t^n}^{t^{n+1}} \sum_{p(c)} \left(\frac{\partial V_c}{\partial r_p} u_p + \frac{\partial V_c}{\partial z_p} v_p \right) \mathrm{d}t, \tag{5.31}$$

which holds for any differentiable function $V_c(t)$ satisfying $V_c(t^n) = V_c^n$ and $V_c(t^{n+1}) = V_c^{n+1}$. Let us assume $\boldsymbol{U}_p^{n+\frac{1}{2}}$ be a constant within the interval $\langle t^n, t^{n+1} \rangle$ and set

$$\boldsymbol{x}_p(t) = \boldsymbol{x}_p^n + (t - t^n)\boldsymbol{U}_p^{n+\frac{1}{2}}, \tag{5.32}$$

then

$$V_c^{n+1} - V_c^n = \int_{t^n}^{t^{n+1}} \sum_{p(c)} \left(\frac{\partial V_c}{\partial r_p} u_p^{n+\frac{1}{2}} + \frac{\partial V_c}{\partial z_p} v_p^{n+\frac{1}{2}} \right) \mathrm{d}t. \tag{5.33}$$

The cell volume V_c can be expressed as:

$$V_c = \frac{1}{6} \sum_{e(c)} (r_p^2 + r_p r_{p+1} + r_p^2) n_{p,p+1}^r, \tag{5.34}$$

or

$$V_c = \frac{1}{6} \sum_{e(c)} [z_p r_{p+1} + z_{p+1} rp + 2(z_p r_p + z_{p+1} r_{p+1})] n_{p,p+1}^z, \tag{5.35}$$

where $\boldsymbol{n}_{p,p+1} = (n_{p,p+1}^r, n_{p,p+1}^z)$ is the normal vector to the edge defined by endpoints p and $p+1$. Because $n_{p,p+1}^r = z_{p+1} - z_p$ depends only on z-coordinates and $n_{p,p+1}^z = r_p - r_{p+1}$ depends only on r coordinates. From (5.34) one obtains:

$$\frac{\partial V_c}{\partial r_p} = \frac{1}{6} [(2r_p + r_{p+1}) n_{p,p+1}^r + (2rp + r_{p-1}) n_{p,p-1}^r] \tag{5.36}$$

and from (5.35)

$$\frac{\partial V_c}{\partial z_p} = \frac{1}{6} [(2r_p + r_{p+1}) n_{p,p+1}^z + (2rp + r_{p-1}) n_{p,p-1}^z]. \tag{5.37}$$

Then:

$$\begin{aligned} &\sum_{p(c)} \left(\frac{\partial V_c}{\partial r_p} u_p^{n+\frac{1}{2}} + \frac{\partial V_c}{\partial z_p} v_p^{n+\frac{1}{2}} \right) = \\ &= \frac{1}{6} \left[\sum_{p(c)} (2r_p + r_{p+1}) \boldsymbol{u}_p \cdot \boldsymbol{n}_{p,p+1} + \sum_{p(c)} (2r_p + r_{p-1}) \boldsymbol{u}_p \cdot \boldsymbol{n}_{p,p-1} \right]. \end{aligned} \tag{5.38}$$

After shifting index p to $p+1$ we can write:

$$V_c^{n+1} - V_c^n = \int_{t^n}^{t^{n+1}} \frac{1}{6} \sum_{e(c)} [(2r_p + r_{p+1})\boldsymbol{u}_p + (2r_{p+1} + r_p)\boldsymbol{u}_{p+1})] \cdot \boldsymbol{n}_e \mathrm{d}t, \tag{5.39}$$

which means that for computing the flux function (5.30), the following expression has to be used:

$$f_{ec}^{\eta} = \frac{1}{6}[(2r_p + r_{p+1})\boldsymbol{u}_p + (2r_{p+1} + r_p)\boldsymbol{u}_{p+1})] \cdot \boldsymbol{n}_e, \tag{5.40}$$

which corresponds to the Simpson's rule for spatial quadrature for (5.30). Let us note that the time integration in (5.31) has to be also discretized using Simpson's quadrature on rr and rz products. If done so, the new geometric volume and the volume obtained from (5.29) would be equal.

5.4.2 Artificial dissipation in cylindrical coordinates

Similarly to [64], [66], after each LW step, the artificial dissipation is added. The artificial viscosity and energy update has the following form:

$$\boldsymbol{U}_c^{n+1} = \boldsymbol{U}_c^{n+1,LW} + \tau_{\boldsymbol{u}} \frac{\Delta t}{m_c} \left[\sum_{e \in e(c)} (r_e \boldsymbol{D}_e^{\boldsymbol{u}}) + \begin{pmatrix} C^{AV} \\ 0 \end{pmatrix} \right] \tag{5.41a}$$

$$E_c^{n+1} = E_c^{n+1,LW} + \tau_E \frac{\Delta t}{m_c} \sum_{e \in e(c)} \left(r_e \boldsymbol{D}_e^E \right) \tag{5.41b}$$

where the upper index LW refers to the Lax-Wendroff step given by (5.23) and (5.27). $\boldsymbol{D}_{e(ca)}^{\boldsymbol{U}}$, $\boldsymbol{D}_{e(ca)}^E$ are artificial viscosity and artificial energy fluxes over the edge $e(ca)$ respectively:

$$\boldsymbol{D}_{e(ca)}^{\boldsymbol{U}} = \frac{\sigma_{ca}\sigma_{ac}|\boldsymbol{n}_{e(ca)}^n|}{\sigma_{ca} + \sigma_{ac}} (\boldsymbol{U}_c^n - \boldsymbol{U}_a^n), \tag{5.42a}$$

$$\boldsymbol{D}_{e(ca)}^E = \frac{\sigma_{ca}\sigma_{ac}|\boldsymbol{n}_{e(ca)}^n|}{\sigma_{ca} + \sigma_{ac}} (E_c^n - E_a^n), \tag{5.42b}$$

where

$$\sigma_{ca} = \varrho_c^n \left[(c_s)_c^n + \frac{|(\boldsymbol{U}_c^n - \boldsymbol{U}_a^n) \cdot \boldsymbol{n}_{e(ca)}^n|}{|\boldsymbol{n}_{e(ca)}^n|} \right]$$

is modified acoustic impedance used here as HLL wave speed, and $(c_s)_c = \sqrt{\gamma P_c / \varrho_c}$ is the cell sound speed. $\tau_{\boldsymbol{u}}$ and τ_E are parameters. In rz we usually set $\tau_{\boldsymbol{u}} = \tau_E = 1.25$. The C_{AV} is AV correction term in Eq. (5.41a) and is added only to the r-component of velocity, which is denoted by the column vector:

$$C_c^{AV} = \frac{1}{2} \sum_{a \in \{e,g\}} C_{e(ac)}^{AV} = \frac{1}{2} \sum_{a \in \{e,g\}} \left(B_{ac} \frac{u_c}{r_c} \right), \quad B_{ac} = \sigma_{ac} |\boldsymbol{n}_{e(ac)}| |\boldsymbol{x}_a - \boldsymbol{x}_c|^2, \tag{5.43}$$

where the edges e and g lie on rays. The derivation of (5.43) is presented in Section B.7.

5.5 Numerical results in 2D cylindrical coordinates

Similarly to [64], [67], all the following tests are computed using LWrz+2, i.e. $\tau_d = 0, \tau_v = \tau_h = \tau$ if not stated otherwise.

5.5.1 Free expansion problem

The free expansion problem[36], [87], [100] deals with a free expansion of ideal gas with $\gamma = 5/3$ into the vacuum. The exact solution can be obtained, and hence it can be used to assess the accuracy of the scheme. The computational domain is a two-dimensional interval in polar coordinates $(r, z) \in \langle 0, 1\rangle \times \langle 0, \pi/2\rangle$ on the logically quadrilateral polar mesh. The initial conditions are:

$$\varrho = 1, \qquad \boldsymbol{U} = \boldsymbol{0}, \qquad P(R) = 1 - R^2,$$

where $R = \sqrt{r^2 + z^2}$. The analytical solution has the following form:

$$\begin{aligned}
R_{\mathrm{o}}(t) &= \sqrt{1 - 2t^2}, & \varrho(t) &= \frac{1}{R_{\mathrm{o}}^3(t)},\\
U(\boldsymbol{r}, t) &= |\boldsymbol{U}(\boldsymbol{r}, t)| = \frac{2t}{1 + 2t^2} R, & P(\boldsymbol{r}, t) &= \frac{1}{R_{\mathrm{o}}^5(t)} \left(1 - \frac{R^2}{R_{\mathrm{o}}^2(t)}\right),
\end{aligned}$$

where R_{o} is the outer radius. The boundary condition on the outer boundary is characterized by $P_{\mathrm{o}} = 0$ (numerically we set 10^{-6}). Similarly to [100], all errors in convergence for this problem are computed only on sub-region $\langle \frac{3}{10} N_r + 1, \frac{7}{10} N_r\rangle \times \langle 1, N_\varphi\rangle$, where N_r, N_φ are numbers of cells in radial and axial directions respectively, to exclude the effects of boundary conditions. This is a smooth problem, so it is computed using pure LW. The solution was computed on meshes with $20 \times 20, 40 \times 40, 80 \times 80$ and 160×160 until the final time $T = 0.4$. The convergence rate for the free expansion problem can be found in Table 5.1, the LW scheme achieves numerically second order of convergence for the free expansion test case.

n	$L_1(\varrho)$	$L_1(U)$	$L_1(P)$	NOC(ϱ)	NOC(U)	NOC(P)
20	9.20e-04	2.75e-03	9.04e-04	2.0	2.0	1.9
40	2.34e-04	6.70e-04	2.39e-04	2.0	2.0	1.9
80	5.90e-05	1.67e-04	6.23e-05	2.0	2.0	1.9
160	1.50e-05	4.26e-05	1.63e-05			

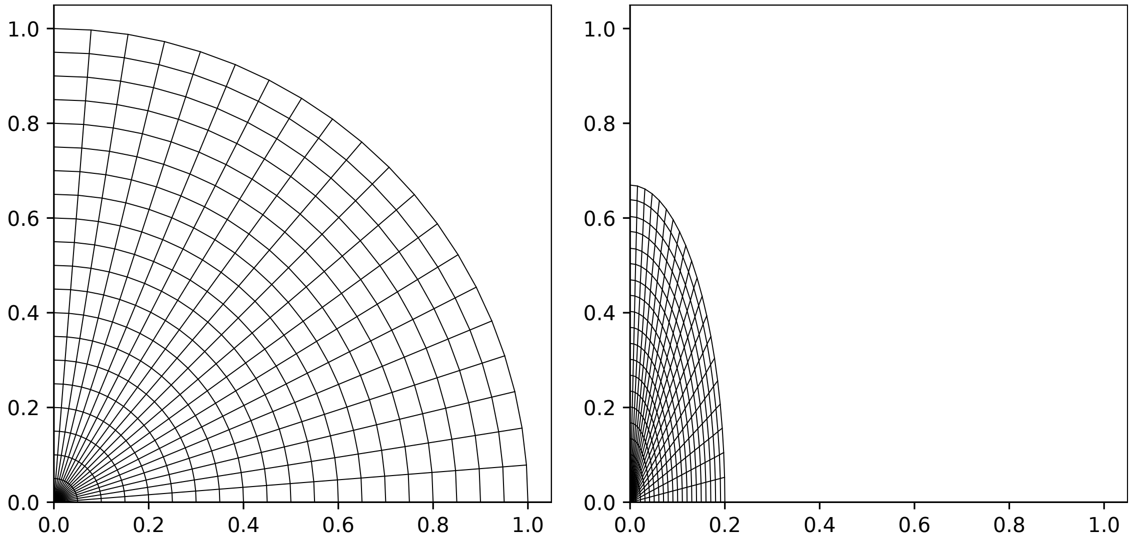
Table 5.1: Convergence rates for density ϱ, radial velocity U and pressure P for Free expansion problem by pure LW.

5.5.2 Coggeshall problem

The next test is the Coggeshall problem, also known as Coggeshall-Meyer-ter Vehn problem [18], [101], [106]. It describes a simple adiabatic compression. The exact solution at time $t \in \langle 0, 1 \rangle$ is

$$\begin{aligned} u &= \frac{-r}{1-t}, \quad v = \frac{-z}{4(1-t)}, \quad \varrho = (1-t)^{\frac{-9}{4}}, \\ \varepsilon &= \left(\frac{3z}{8(1-t)} \right), \quad P = \frac{3}{32}\frac{z^2}{(1-t)^{\frac{17}{4}}}, \quad S = \frac{3}{32}\frac{z^2}{(1-t)^{\frac{1}{2}}} \end{aligned} \tag{5.44}$$

The reflective boundary conditions are applied on the axes. Similarly to [18], we prescribe exact velocities u, v as the boundary conditions on the nodes that initially formed the outer circle. The second option would be to prescribe the exact pressure. With velocity BC, the scheme better preserves elliptical shape. This is a continuous problem, so it was computed with $\tau = 0$. The convergence rates presented in Table 5.2 show that the numerical order of convergence for this test is between 1 and 2. The computational meshes of 20×20 cells at initial time $t = 0$ and final time $T = 0.8$ are shown in Figure 5.3. Regarding the shape of the solution, the results due to the LW scheme are comparable to those in [18], [107], and closer to the exact solution than those presented in [8], [101]



(a) (b)

Figure 5.3: Meshes with 20×20 cells for the Coggeshall problem. At $T = 0$ (a), at $T = 0.8$ (b).

5.5.3 Kidder

The next presented test is the Kidder problem in spherical symmetry. This test is similar to the version assuming cylindrical symmetry presented in Subsection 4.5.1, so we will reduce this description only to the spherical symmetry specifics. We refer the reader to the

n	$L_1(\varrho)$	$L_1(u)$	$L_1(v)$	$L_1(P)$	NOC(ϱ)	NOC(u)	NOC(v)	NOC(P)
20	2.9e-02	1.1e-02	2.9e-02	3.7e-02	1.2	1.3	1.1	1.3
40	1.3e-02	4.4e-03	1.3e-02	1.5e-02	1.4	1.4	1.3	1.4
80	4.9e-03	1.7e-03	5.4e-03	5.7e-03	1.5	1.5	1.4	1.5
160	1.7e-03	5.9e-04	2.1e-03	2.1e-03				

Table 5.2: Convergence rates for density, u and v velocity components, and pressure for the Coggeshall problem by pure LW.

mesh	$L_1(\varrho)$	$L_1(P)$	$L_1(U)$	NOC(ϱ)	NOC(P)	NOC(U)
45x25	4.8e-03	8.0e-03	3.7e-03	1.3	1.6	2.0
90x50	1.9e-03	2.6e-03	9.4e-04	1.4	1.5	1.8
180x100	7.2e-04	9.2e-04	2.7e-04	1.4	1.5	1.7
360x200	2.7e-04	3.3e-04	8.2e-05			

Table 5.3: Convergence rates for density, velocity, and pressure for the Kidder problem.

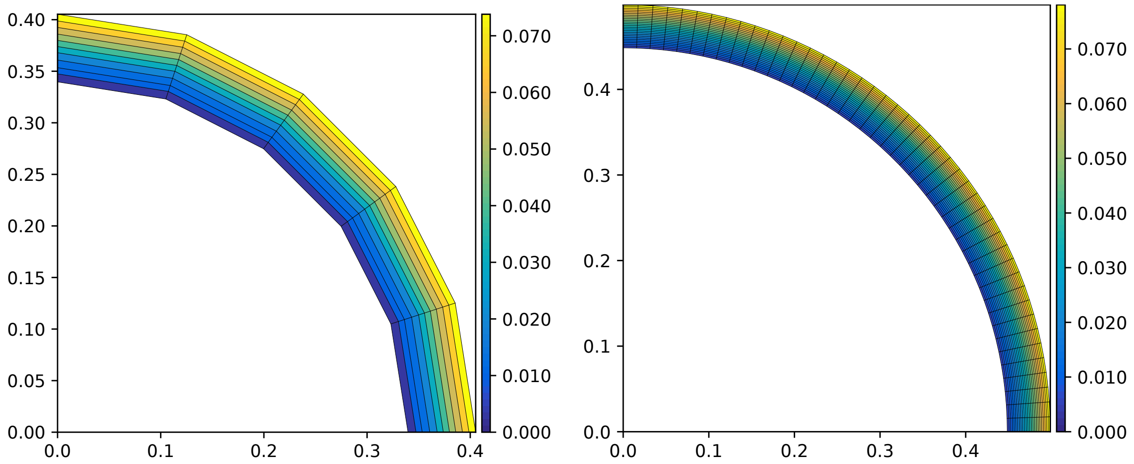
version in Cartesian 2D geometry for the detailed description. For a spherical symmetry, we set $\nu = 3$ and hence $\gamma = \frac{5}{3}$ and by R we understand spherical radius defined by $R = \sqrt{r^2 + z^2}$. The only other change is the focusing time $t_\mathrm{f} = 6.72 \times 10^{-3}$, so the final time at which the outer radius $R_\mathrm{o} = 0.5$ is $t = (\sqrt{3}/2)t_\mathrm{f} = 5.82 \times 10^{-3}$. The Kidder test deals with a smooth flow, so it is computed by pure LW.

Similarly to the Cartesian geometry, we present the numerical order of convergence. Tables 5.3 and 5.4 show that the rz version of LW scheme with predictor (5.20-5.21) and corrector (5.23) can achieve second order of accuracy.

The pseudocolor plots of density computed on meshes with 5×10 and 45×25 cells are presented in Figure 5.4.

mesh	$\mathcal{R}_\mathrm{i}$	$\mathcal{R}_\mathrm{o}$	$L_1(\mathcal{R}_\mathrm{i})$	$L_1(\mathcal{R}_\mathrm{o})$	NOC($\mathcal{R}_\mathrm{i}$)	NOC($\mathcal{R}_\mathrm{o}$)
45x25	0.4486	0.4988	1.3e-03	1.0e-03	2.0	2.0
90x50	0.4496	0.4996	3.3e-04	2.6e-04	2.0	2.0
180x100	0.4498	0.4998	8.3e-05	6.5e-05	2.0	2.0
360x200	0.4499	0.4999	2.1e-05	1.6e-05		

Table 5.4: Convergence rates for inner radius $\mathcal{R}_i$ and outer radius $\mathcal{R}_\mathrm{o}$ for the Kidder problem.



(a) (b)

Figure 5.4: Pseudocolor plots of the Kidder problem on polar mesh computer by pure LW; 5 × 10 cells (a), 45 × 25 cells (b).

5.5.4 Noh

The next test is the Noh problem in rz geometry. Similarly to the Noh problem in xy geometry presented in Subsection 4.5.2. This test deals with a spherically symmetric shockwave of ideal gas characterized by $\gamma = 5/3$ reflecting from the origin. The initial conditions remain the same as in the xy case, which means unit density, zero pressure (numerically we set $P_0 = 10^{-6}$), and unit velocity pointing towards the origin. The computational domain is again the square $\langle 0, 1\rangle^2$ for the case with initially rectangular mesh, or a single quarter of circle given by angle $\varphi \in \langle 0, \pi/2\rangle$, and radius $r \in \langle 0, 1\rangle$. Concerning symmetry, the boundary conditions are reflective on the axes, and the imposed velocity BC is used otherwise. Assuming spherical symmetry, the exact solution is:

$$\varrho(t, \boldsymbol{X}) = \begin{cases} 64 & \text{if } |\boldsymbol{X}|< t/3 \\ (1 + \frac{t}{|\boldsymbol{X}|)^2} & \text{otherwise} \end{cases}, \qquad P(t, \boldsymbol{X}) = \begin{cases} 64/3 & \text{if } |\boldsymbol{X}|< t/3 \\ 0 & \text{otherwise} \end{cases}.$$

$$\boldsymbol{U}(t, \boldsymbol{x}) = \begin{cases} \boldsymbol{0} & \text{if } |\boldsymbol{x}|< t/3 \\ -\frac{\boldsymbol{x}}{|\boldsymbol{x}|} & \text{otherwise} \end{cases}.$$

The final time remains $T = 0.6$, which corresponds to the shock wave position at $R = 0.2$. The results have been computed using an unlimited LW+2 scheme with $\tau = 1.25$ and a limited scheme with $\tau = 1.75$. Let us first present a comparison of LW and classical staggered method with tensor viscosity [9]. The pseudocolor plots of density are shown in Figure 5.5 and the contour plots in Fig. 5.6. Similarly to the xy case, the symmetry of the solution due to LW is significantly better. The solution on the polar meshes remains exactly symmetric. The scatter plots for subsequently refined initially rectangular meshes with resolution 50 × 50, 100 × 100, and 200 × 200 cells can be seen in Figure 5.7(a), the results on polar grids with 25 × 50, 50 × 100, and 100 × 200 are presented in Figure 5.7(b).

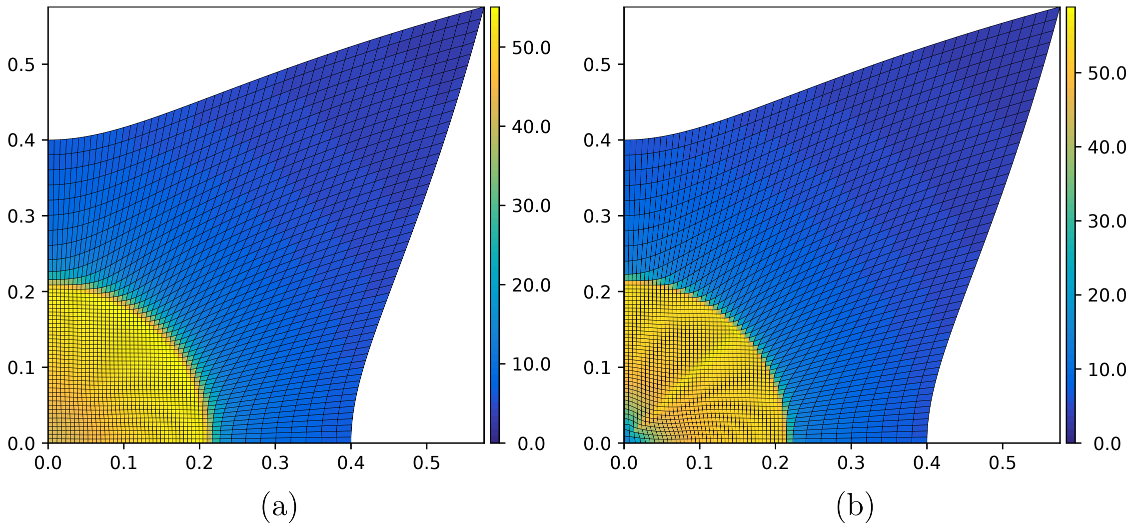


Figure 5.5: Pseudocolor plots of Noh problem on an initially rectangular mesh. Unlimited LW+2 with $\tau = 1.25$ (a) and Staggered scheme with tensor viscosity [9] (b).

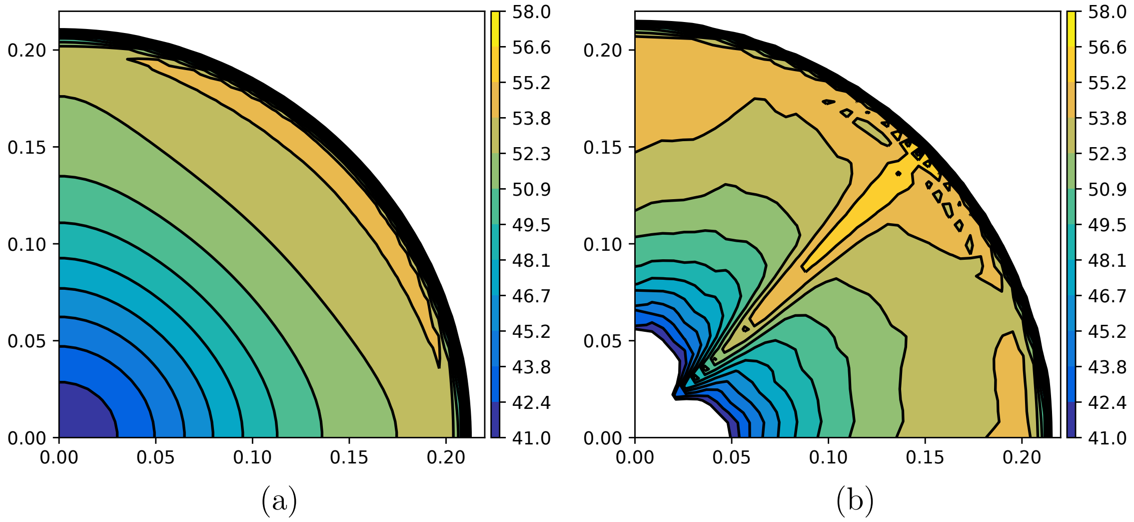
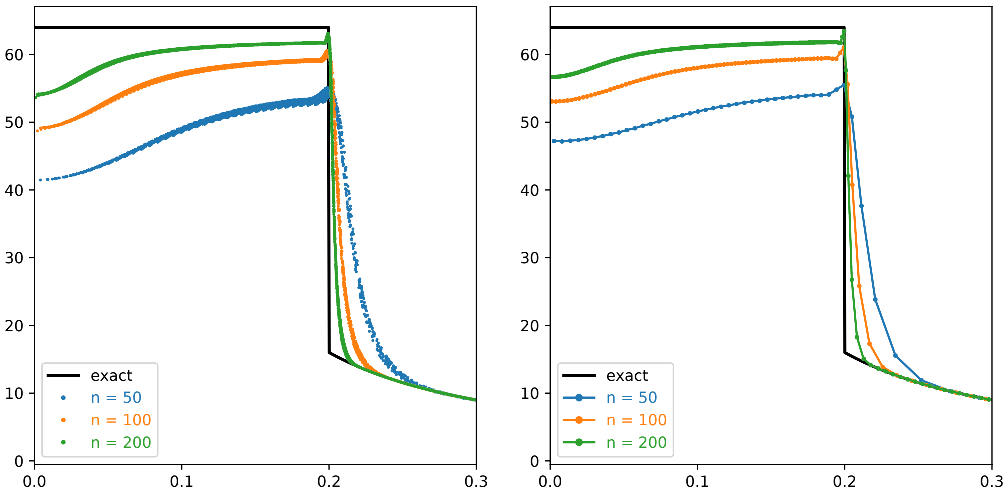


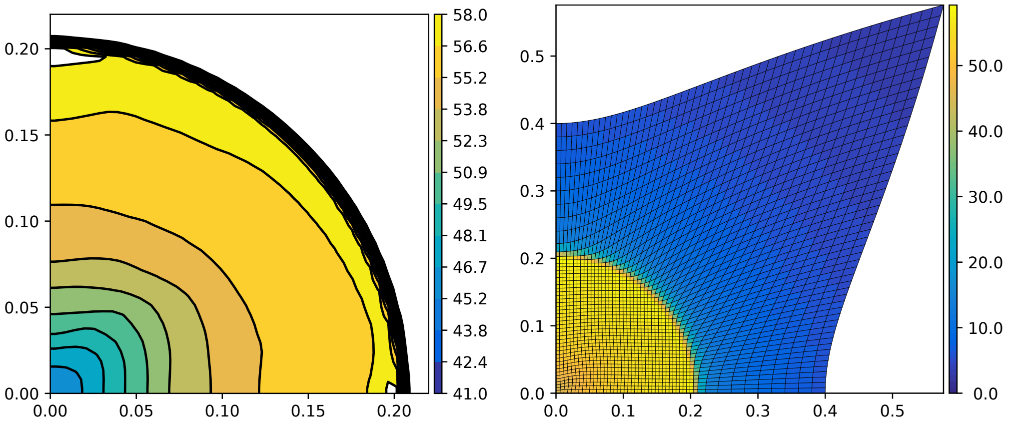
Figure 5.6: Contour plots of Noh problem on an initially rectangular mesh. Unlimited LW+2 with $\tau = 1.25$ (a) and Staggered scheme with tensor viscosity [9] (b).



(a) (b)

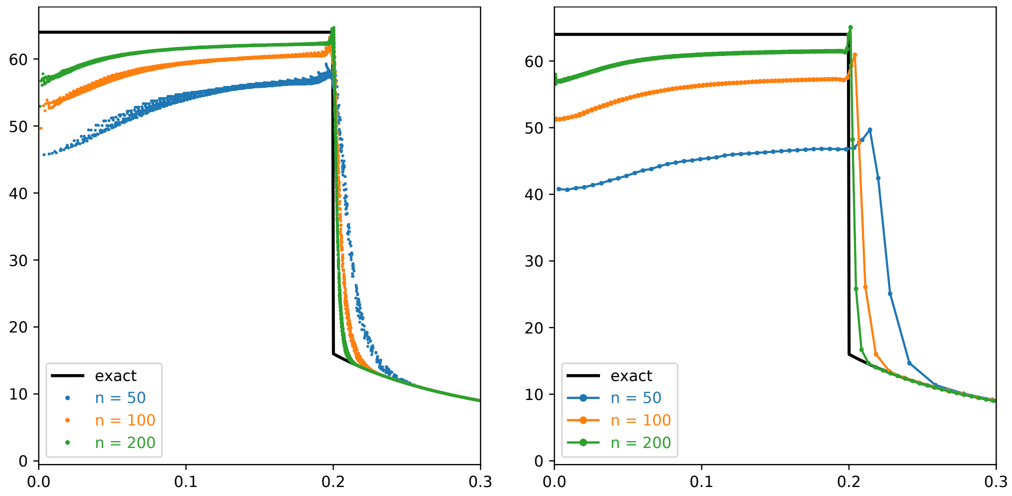
Figure 5.7: Scatter plots of Noh problem computed using the unlimited LW+2 with $\tau = 1.25$; on an initially rectangular mesh (a), on polar mesh (b).

For the sake of completeness, we will now present also some results computed using the limited LW+2 with $\tau = 1.75$. The contour and pseudocolor plots for density on initially orthogonal mesh with 50×50 cells can be found in Fig. 5.8, while the density scatter plots are presented in Figure 5.9. The symmetry on the initially rectangular mesh is worse but still reasonable. On the other hand, the shock wave position and the plateau behind the shock are closer to the exact solution.



(a) (b)

Figure 5.8: Contour (a) and pseudocolor plot (b) for density for Noh problem in rz geometry. Limited LW+2 with $\tau = 1.75$ on coarse 50×50 initially rectangular grid.



(a) (b)

Figure 5.9: Scatter plots of Noh problem computed using the limited LW+2 with $\tau = 1.75$; on initially rectangular meshes (a), on polar meshes (b).

5.5.5 Spherical Sedov

The spherical Sedov test shares all initial and boundary conditions except the value of blast energy with its Cartesian version. We again consider square $\langle 0, 1.2\rangle^2$ or a single quarter of circle given by angle $\varphi \in \langle 0, \pi/2\rangle$ and radius $r \in \langle 0, 1.2\rangle$ filled by an ideal gas characterized by $\gamma = 1.4$, initially at rest. The initial density is unit, and the initial pressure is set to zero (numerically set to 10^{-6}) everywhere except the cells having some vertex at the origin (a single cell for square or entire circle of cells for polar mesh), where we set the initial internal blast energy $\mathcal{E}_{\text{blast}} = 0.851072$. This value corresponds to the shock wave position $r = 1$ at the final time $T = 1$.

Pseudocolor plots of density computed on coarse grids 50×50 for the square case and 25 angular times 50 radial cells in polar case computed using the unlimited version of LW+2 with $\tau = 1.25$ are presented in Figure 5.10. Figures 5.11 and 5.12 then show convergences to the exact solution taken from [88].

In Figures 5.13, 5.14, and 5.15, we present similar plots of results due to limited LW+2 with $\tau = 1.75$. The 9-edge stencil limiter has been used for square cases. As limiting ray edges on polar grids can break the symmetry, for polar meshes, only edges on mesh circles have been limited.

The symmetry is exact for polar grids while remaining very good on initially rectangular meshes. The smooth solution behind the shockwave is better resolved when the limiter is used.

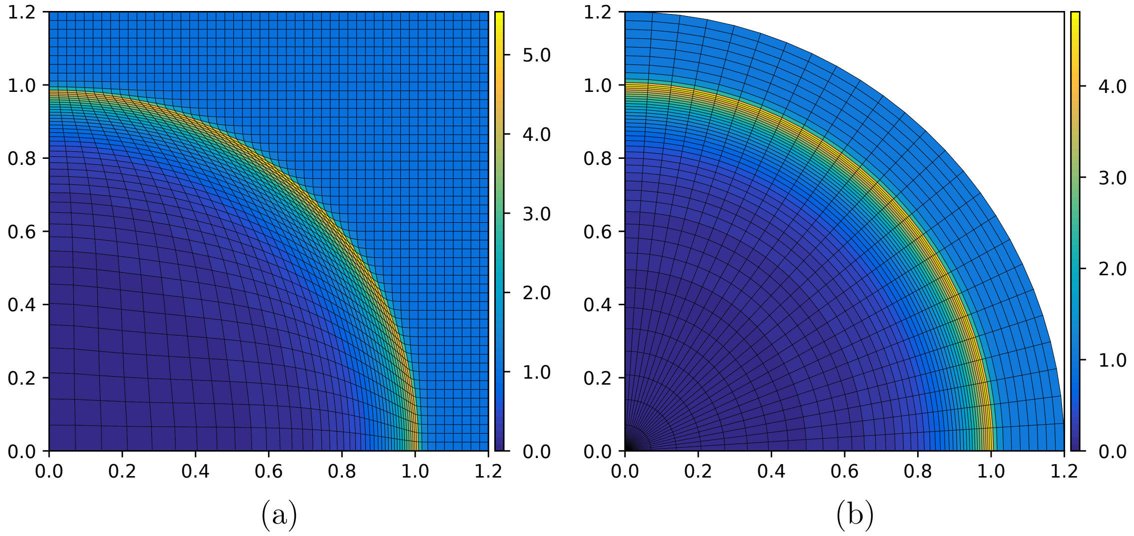


Figure 5.10: Pseudocolor plots of the Sedov problem in cylindrical geometry computed using the unlimited LW+2 $\tau = 1.25$ on initially rectangular grid (a) initially equiangular grid (b).

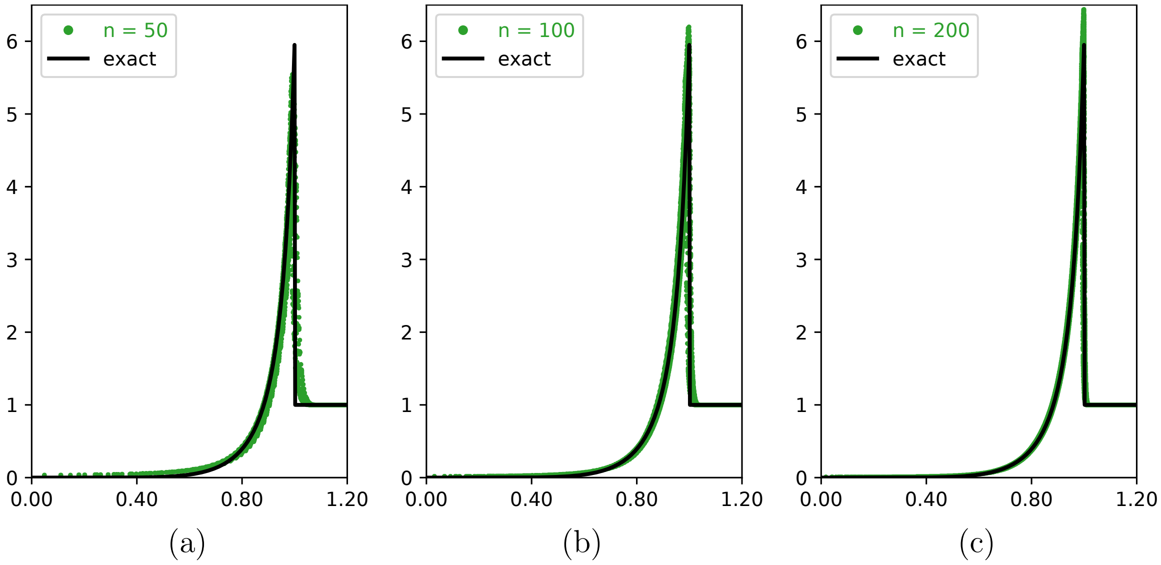
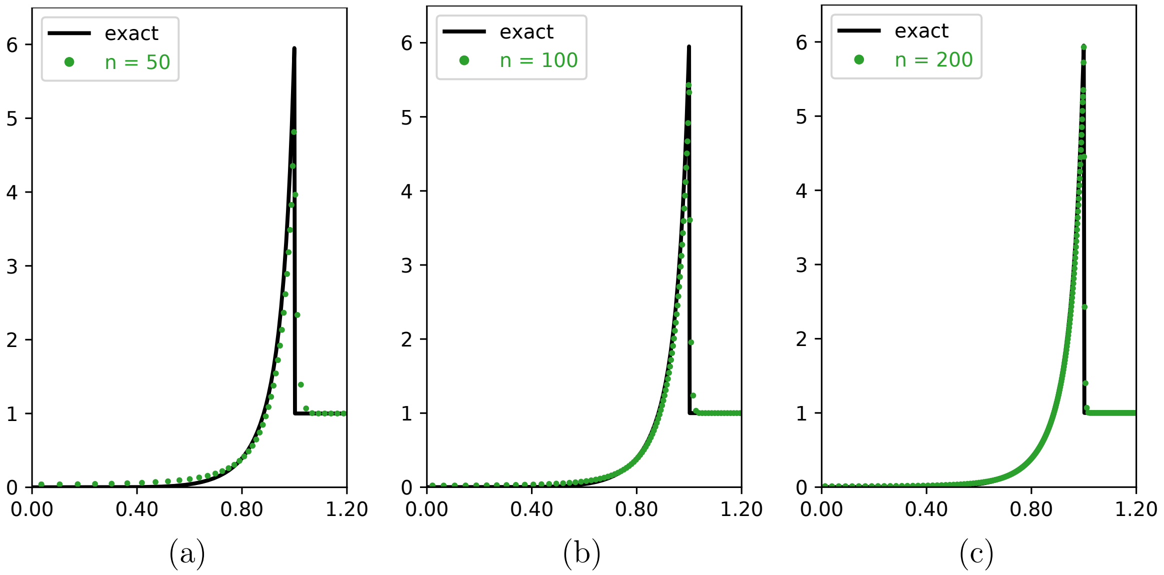
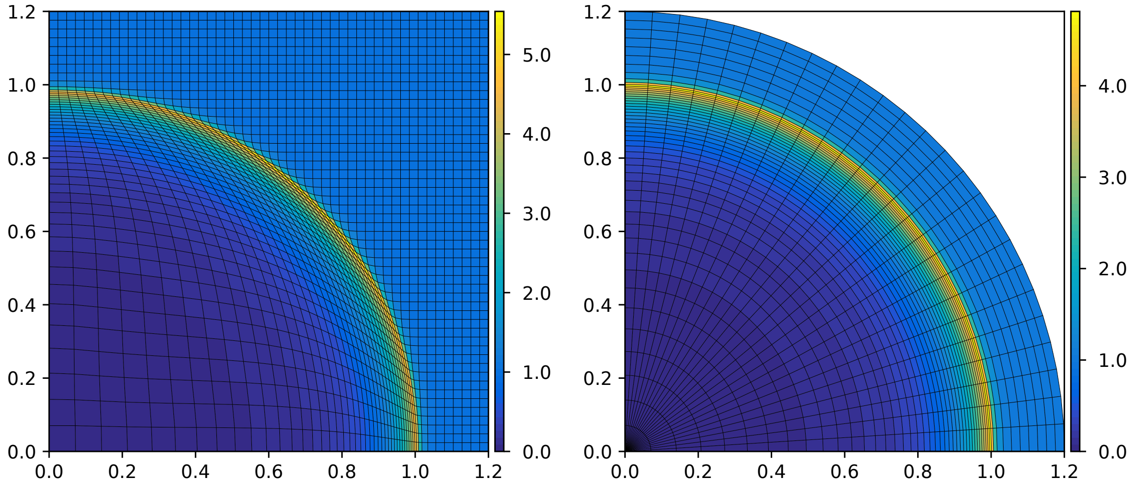


Figure 5.11: Scatter plots of the Sedov problem computed using the unlimited LW+2 with $\tau = 1.25$ on initially rectangular meshes.



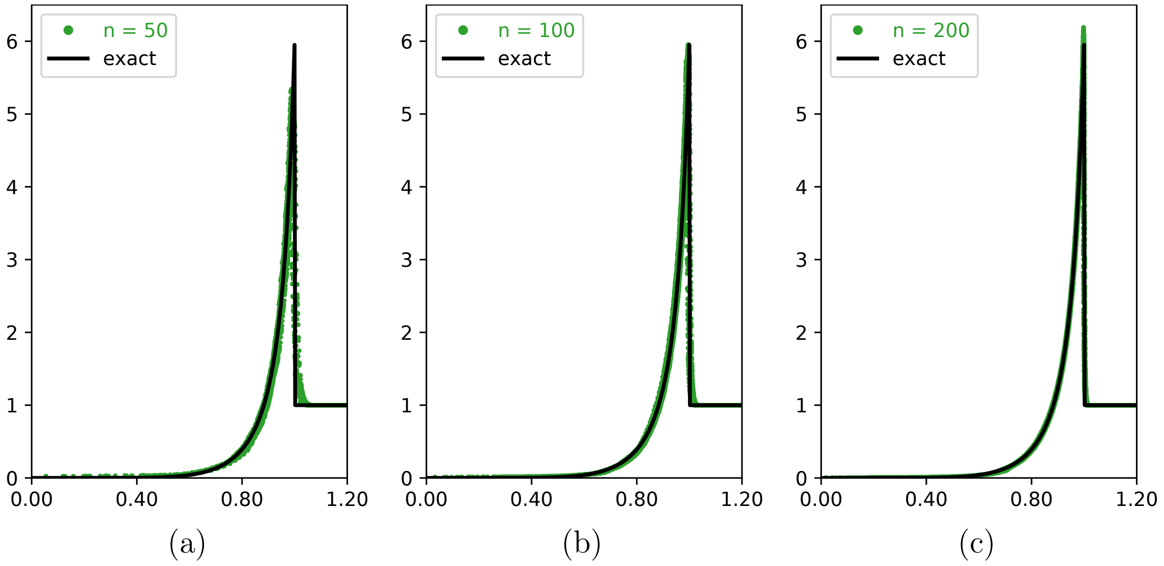
(a) (b) (c)

Figure 5.12: Scatter plots of the Sedov problem computed using unlimited LW+2 with $\tau = 1.25$ on polar meshes.



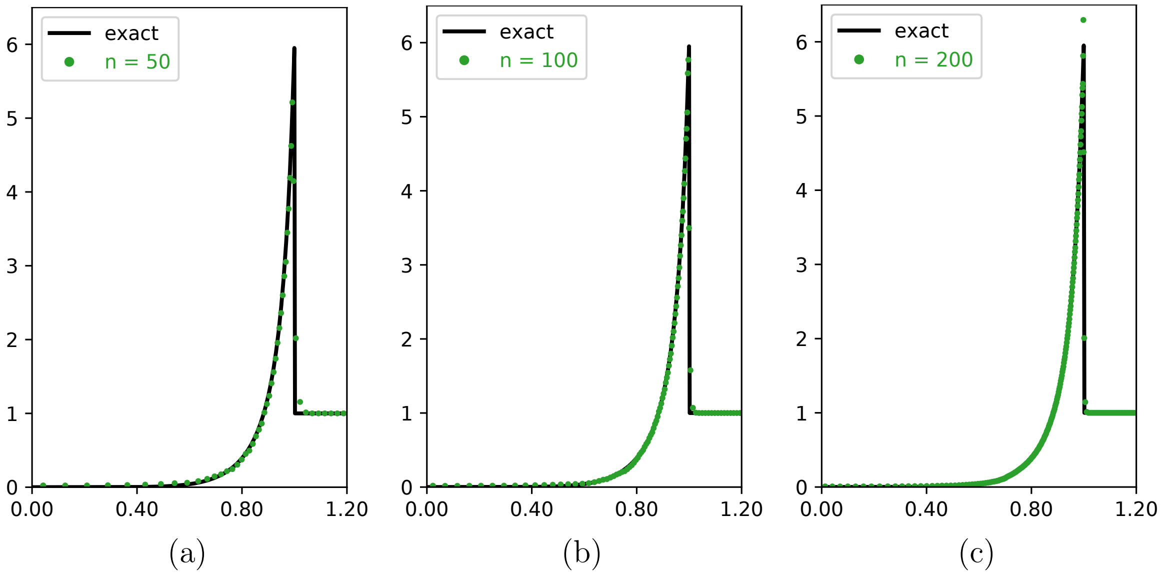
(a) (b)

Figure 5.13: Pseudocolor plots of the Sedov problem in cylindrical geometry computed using unlimited LW+2 with $\tau = 1.25$ on initially rectangular grid (a) initially equiangular grid (b).



(a) (b) (c)

Figure 5.14: Scatter plots of the Sedov problem computed using limited LW+2 with $\tau = 1.75$ on initially rectangular meshes.



(a) (b) (c)

Figure 5.15: Scatter plots of the Sedov problem computed using limited LW+2 with $\tau = 1.75$ on polar meshes.

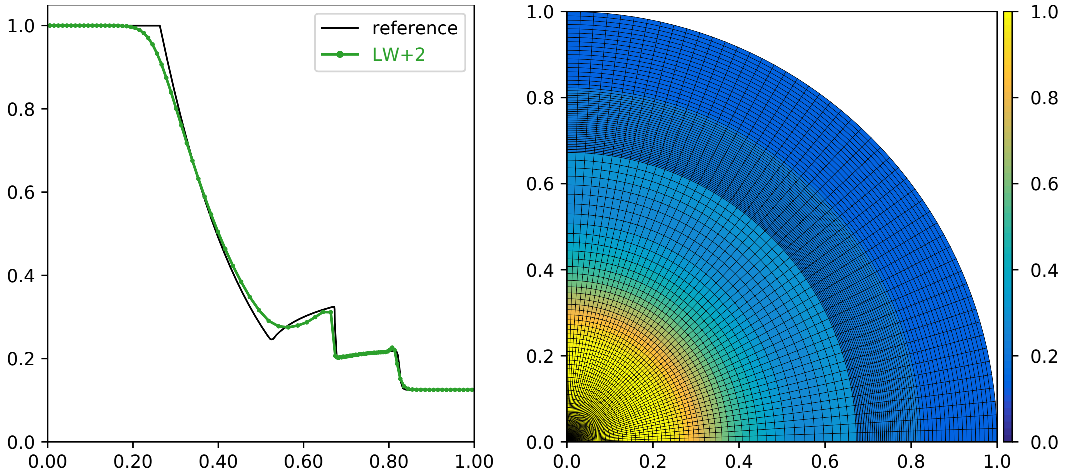
5.5.6 Spherical Sod

The next test is a generalization of the 1D Sod shock tube presented in Subsection 3.5.1 into spherical geometry [29]. The initial conditions of 1D Sod are copied to every ray of the polar mesh 25×100 cells covering a single quarter of the circle by angle $\alpha \in \langle 0, \pi/2 \rangle$, and radius $R \in \langle 0, 1 \rangle$. The reflective boundary conditions are imposed everywhere.

The initial conditions are given by:

$$(\varrho, \boldsymbol{U}, P) = \begin{cases} (1.0, \mathbf{0}, 1.0) & \text{for} R \leqslant 0.5 \\ (0.125, \mathbf{0}, 0.1) & \text{for} R \geqslant 0.5 \end{cases} \qquad R = \sqrt{r^2 + z^2}$$

The final time is $T = 0.2$. Similarly to the 1D version, the results for this test case computed using limited LW+2 with artificial energy flux turned off on contact and *interface fix* enabled. The dissipation coefficient τ has been set to 1.25. The results presented in Figure 5.16 are comparable with other methods like e.g. [29], [43], [52], [101].



(a) (b)

Figure 5.16: Scatter and pseudocolor plots of spherical Sod problem at $T = 0.2$ computed using limited LW+2 with $\tau = 1.25$ on the polar mesh. 25×100 cells.

5.5.7 Axisymmetric Triple-Point

The last presented problem is a generalization of the Triple point problem defined in Subsection 4.5.5 into spherical geometry inspired by [15], [53]. The initial conditions are defined in Figure 5.17. The boundary conditions remain the same as in Cartesian geometry, i.e. reflecting BC's everywhere. The final time is again $T = 2.7$, while the τ has been set to 1.25. Figures 5.18 and 5.19 show the results computed using unlimited and limited versions of LW+2 respectively on 70×30 initially uniform orthogonal mesh.

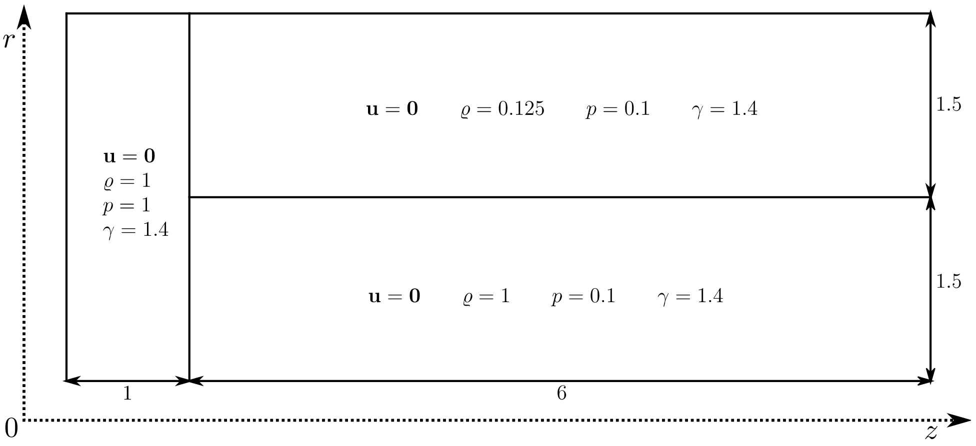


Figure 5.17: Initial conditions for the triple point problem in cylindrical geometry.

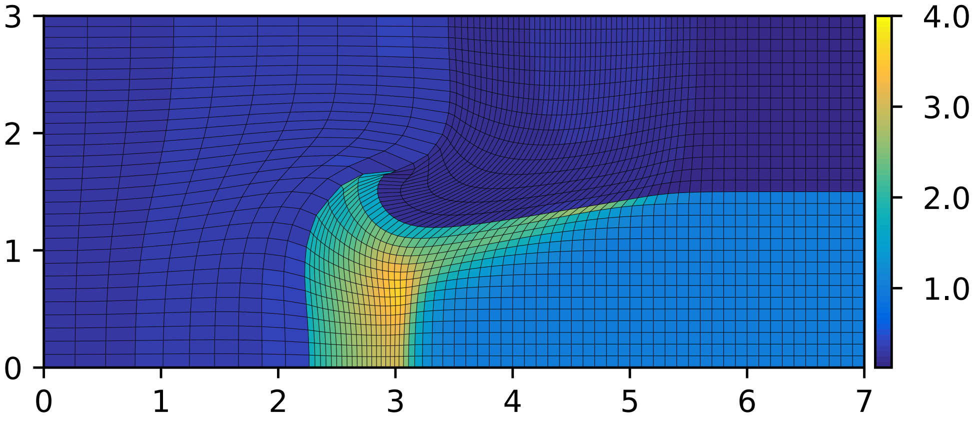


Figure 5.18: Axisymmetric triple point problem computed using the unlimited LW+2 scheme on 70 × 30 initially orthogonal equally spaced grid.

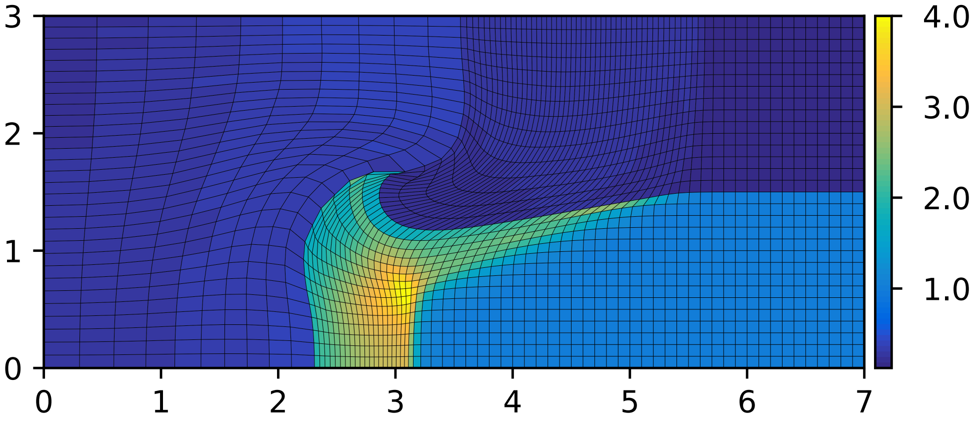


Figure 5.19: Axisymmetric triple point problem computed using the limited LW+2 scheme on 70 × 30 initially orthogonal equally spaced grid.

Similarly to the Cartesian geometry version of this test, using limiter enhances the resolution of the density peak near the triple point, and compared with that, the results are reasonable, although worse than these published in [15] since our method uses only straight lines on cell edges, which limits the vortex resolution.

Chapter 6

Wilkins model for elastoplasticity

The *hypoelastic*, also referred to as an *elastic-perfectly plastic*, material description of solid dynamics has been introduced by Wilkins in [3]. The model consists of a system of conservation laws coupled with an incremental constitutive law. The conservation laws part is expressed using momentum and energy fluxes that are dependent on Cauchy stress tensor comprising hydrostatic pressure and traceless deviatoric stress tensor. The incremental constitutive law describes the evolution of the deviatoric stress as a dependency of the velocity gradient. The principle of material frame indifference (independence on the reference frame) is fulfilled by means of Jaumann objective derivative, which involves the spin tensor (the antisymmetric part of the velocity gradient). Within the Wilkins model, the strain rate tensor is decomposed into the elastic part, and the plastic deviatoric part and von Mises yield criterion is used for modelling of plastic yielding, which works well for metals [32].

Historically, the first method for the hypoelastic model was the staggered method due to Wilkins [3], later followed by the Godunov method [108] and free-Lagrange [109] methods. Recently, the development of the Finite Element Method [110] has been proposed, and significant interest has been drawn in cell-centered methods [32], [33], [36], [111].

Looking away from the Wilkins model, a consistent non-dissipative hypoelastic model based on the 4D Lie derivative has been proposed in [112], [113].

Besides hypoelastic models, it is worth to mention also hyperelastic model [114]–[118], where the Cauchy stress tensor is derived from strain energy density function [119], and unified first-order hyperbolic formulation of continuum mechanics [120], which describes the fluid and solid mechanics using a single universal hyperbolic system.

Wilkins model suffers from several drawbacks, namely somehow arbitrary choice of objective derivative, the lack of thermodynamic consistency (the entropy is produced even in elastic regime), and the fact that due to the use of Jaumann derivative, the system cannot be expressed in a conservative form, which makes its mathematical analysis questionable [114].

Despite the existence of more rigorous models and the above-mentioned problems, the Wilkins model is still being used even in commercial computational codes, such as LS-DYNA [121], mainly due to its simplicity and good accuracy for metal dynamics.

6.1 Conservation laws

Within the Wilkins model, the elastic behaviour is described using Euler equations. Unlike hydrodynamics, Cauchy stress tensor no longer reduces to pressure, so in Lagrangian coordinates, we can write:

$$\varrho\frac{\mathrm{d}\eta}{\mathrm{d}t} - \boldsymbol{\nabla}\cdot\boldsymbol{U} = 0, \tag{6.1a}$$

$$\varrho\frac{\mathrm{d}\boldsymbol{U}}{\mathrm{d}t} - \boldsymbol{\nabla}\cdot\mathbb{T} = \boldsymbol{0}, \tag{6.1b}$$

$$\varrho\frac{\mathrm{d}E}{\mathrm{d}t} - \boldsymbol{\nabla}\cdot(\mathbb{T}\boldsymbol{U}) = 0, \tag{6.1c}$$

where P is the pressure given by EOS (6.17), $\mathbb{I}$ is the tensor identity, and $\mathbb{S}$ is deviatoric stress. The equation (6.1a) is the equation of continuity (the law of conservation of mass), (6.1b) is the law of conservation of momentum, and (6.1c) is the law of conservation of total energy. The integral form of system (6.1) together with GCL (2.48) is:

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho\mathrm{d}v = 0, \tag{6.2a}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\mathrm{d}v - \int_{\partial\omega}\boldsymbol{U}\cdot\boldsymbol{n}\mathrm{d}s = 0, \tag{6.2b}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho\boldsymbol{U}\mathrm{d}v - \int_{\partial\omega}\mathbb{T}\cdot\boldsymbol{n}\mathrm{d}s = \boldsymbol{0}, \tag{6.2c}$$

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{\omega}\varrho E\mathrm{d}v - \int_{\partial\omega}\mathbb{T}\boldsymbol{U}\cdot\boldsymbol{n}\mathrm{d}s = 0. \tag{6.2d}$$

The system (6.2) can be rewritten using a single vector equation (2.52) with the only difference in physical flux vector $\boldsymbol{F}$, which is now

$$\boldsymbol{F} = (\boldsymbol{U}, \mathbb{T}, \mathbb{T}\cdot\boldsymbol{U})^t. \tag{6.3}$$

6.2 Incremental constitutive law

For an elastic–perfectly plastic material, the evolution of $\mathbb{S}$ is given by incremental constitutive law. Within the whole description, we will follow the notations presented in [32]. Let us start by defining the strain rate tensor:

$$\mathbb{D} = (\nabla\boldsymbol{U} + (\nabla\boldsymbol{U})^t). \tag{6.4}$$

The strain rate tensor $\mathbb{D}$ is by definition symmetric, and its trace is equal to the velocity divergence:

$$\mathrm{tr}(\mathbb{D}) = \nabla\cdot\boldsymbol{U}. \tag{6.5}$$

Similarly, we can define antisymmetric spin tensor:

$$\mathbb{W} = (\nabla\boldsymbol{U} - (\nabla\boldsymbol{U})^t). \tag{6.6}$$

The deviatoric part of strain rate tensor $\mathbb{D}$ is

$$\mathbb{D}_0 = \mathbb{D} - \frac{1}{3}\mathrm{tr}(\mathbb{D})\mathbb{I}. \tag{6.7}$$

The strain rate tensor $\mathbb{D}$ can also be split into elastic ($\mathbb{D}^e$) and plastic ($\mathbb{D}^p$) parts:

$$\mathbb{D} = \mathbb{D}^e + \mathbb{D}^p. \tag{6.8}$$

The plastic part is deviatoric because

$$\mathrm{tr}(\mathbb{D}^p) = 0.$$

Let us define the inner product of two tensors

$$\mathbb{P} : \mathbb{Q} = \mathrm{tr}(\mathbb{P} \cdot \mathbb{Q}^t).$$

To incorporate plasticity, the magnitude of stress tensor $|\mathbb{S}| = \sqrt{\mathbb{S} : \mathbb{S}}$, is limited by von Mises yield criterion:

$$|\mathbb{S}| \leqslant \sqrt{\frac{2}{3}} Y_0 \qquad \Leftrightarrow \qquad f = |\mathbb{S}| - \sqrt{\frac{2}{3}} Y_0 \leqslant 0, \tag{6.9}$$

where Y_0 is the yield strength of the material under simple tension, and f is the function for yield surface. The plastic part of the strain rate tensor $\mathbb{D}^p$ reads

$$\mathbb{D}^p = (\mathbb{N}^p : \mathbb{D})\chi\mathbb{N}^p, \tag{6.10}$$

where

$$\mathbb{N}^p = \frac{1}{|\mathbb{S}|}\mathbb{S}$$

is the plastic flow direction and χ is the logical switch defined by

$$\chi = \begin{cases} 0 & \text{if} \quad f < 0 \quad \text{or} \quad f = 0 \wedge \mathbb{N}^p : \mathbb{D} \leqslant 0 \\ 1 & \text{if} \quad f = 0 \wedge \mathbb{N}^p : \mathbb{D} \geqslant 0. \end{cases} \tag{6.11}$$

Using the above-mentioned definitions, we can finally write the incremental constitutive law:

$$\frac{\mathrm{d}\mathbb{S}}{\mathrm{d}t} = 2\mu(\mathbb{D}_0 - \mathbb{D}^p) - (\mathbb{S} \cdot \mathbb{W} - \mathbb{W} \cdot \mathbb{S}), \tag{6.12}$$

where μ is the Lamé coefficient equal to the shear modulus. Finally, we introduce the Jaumann derivative

$$\overset{\circ}{\mathbb{S}} = \frac{\mathrm{d}S}{\mathrm{d}t} + \mathbb{S} \cdot \mathbb{W} - \mathbb{W} \cdot \mathbb{S}, \tag{6.13}$$

using that, the definition (6.12) can be rewritten as

$$\overset{\circ}{\mathbb{S}} = 2\mu(\mathbb{D}_0 - \mathbb{D}^p). \tag{6.14}$$

6.3 Mie-Grüneisen equation of state

For closing the system (6.2a-6.2d), which describes elastic-plastic flow, we will use Mie-Grüneisen equation of state [1], [32]. Pressure P can be expressed as a function of specific volume $\eta = 1/\varrho$ and specific internal energy ε:

$$P(\eta, \varepsilon) = \frac{a_0^2(\eta_0 - \eta)}{[\eta_0 - s(\eta_0 - \eta)]^2} + \frac{\Gamma(\eta)}{\eta}\left[\varepsilon - \frac{1}{2}\left(\frac{a_0(\eta_0 - \eta)}{\eta_0 - s(\eta_0 - \eta)}\right)^2\right], \tag{6.15}$$

where a_0 and s are empirically obtained coefficients connecting particle speed U_p and shock speed $U_s = a_0 + sU_p$, Γ is Grüneisen coefficient given by

$$\Gamma = \eta \left(\frac{\partial P}{\partial \varepsilon}\right)_V = \Gamma_0 \frac{\eta}{\eta_0}. \tag{6.16}$$

Zero subscribed values η_0, ϱ_0 represent specific volume and density at a reference state. Specific values for particular materials can be found in Table 6.1. For computational purposes, we will stick with the formula found in [32]:

$$P(\varrho, \varepsilon) = \varrho_0 a_0^2 f(\nu) + \varrho_0 \Gamma_0 \varepsilon, \tag{6.17a}$$

$$f(\nu) = \frac{(\nu - 1)\left[\nu - \Gamma_0(\nu - 1)/2\right]}{\left[\nu - s(\nu - 1)\right]^2} \tag{6.17b}$$

where $\nu = \varrho/\varrho_0$ is the ratio of densities of unstressed material and material under load. The sound speed is computed using:

$$a = \sqrt{a_0^2 f'(\nu) + \frac{\Gamma_0}{\varrho_0}\frac{P}{\nu^2}}, \tag{6.18}$$

where $f'(\nu)$ is a simple derivative of $f(\nu)$:

$$f'(\nu) = \frac{\mathrm{d}f}{\mathrm{d}\nu} = \frac{\nu + (s - \Gamma_0)(\nu - 1)}{\left[\nu - s(\nu - 1)\right]^3}. \tag{6.19}$$

6.3.1 Material parameters

Let us summarize the parameters necessary for the Mie-Grüneisen equation of state and constitutive model. The values for materials presented in numerical tests in this work can be found in Table 6.1.

	Mie-Grüneissen EOS parameters				Constitutive parameters	
Material	ϱ_0[kg/m^3]	a_0[m/s]	Γ_0	s	μ[Pa]	Y_0[Pa]
Cu	8930	3940	2	1.49	45×10^9	90×10^6
Al	2785	5328	2	1.338	27.6×10^9	300×10^6
Be	1845	7998	2	1.124	151.9×10^9	330×10^6

Table 6.1: Mie-Grüneissen equation of state and constitutive model parameters (the shear modulus μ and the yield strength Y^0) for Copper, Aluminum, and Beryllium.

Chapter 7

Method for 1D Elastic-Plastic flow

In this chapter, we will discuss the extension of the LW+2 for 1D elastoplasticity described by Wilkins model. Although on the first look, the system is very similar to that describing hydrodynamics, there are a few differences, which will be discussed in more detail.

Assuming 1D geometry, the system (6.1a-6.1c) simplifies to:

$$\varrho \frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{\partial U}{\partial x} = 0 \tag{7.1a}$$

$$\varrho \frac{\mathrm{d}U}{\mathrm{d}t} - \frac{\partial T}{\partial x} = 0 \tag{7.1b}$$

$$\varrho \frac{\mathrm{d}E}{\mathrm{d}t} - \frac{\partial (TU)}{\partial x} = 0, \tag{7.1c}$$

where $T = -P + S_{xx}$ is the Cauchy stress, P is pressure and S_{xx} is deviatoric stress. Using the mass coordinate (3.3) we can rewrite (7.1a-7.1c):

$$\frac{\mathrm{d}\eta}{\mathrm{d}t} - \frac{\partial U}{\partial s} = 0, \tag{7.2a}$$

$$\frac{\mathrm{d}U}{\mathrm{d}t} - \frac{\partial T}{\partial s} = 0, \tag{7.2b}$$

$$\frac{\mathrm{d}E}{\mathrm{d}t} - \frac{\partial (TU)}{\partial s} = 0. \tag{7.2c}$$

Similarly to Section 3.3, the system (7.2a-7.2c) can be compacted into a single vector equation (3.17) with the only difference in physical flux vector which is now

$$\boldsymbol{F} = (U, T, TU)^t. \tag{7.3}$$

The system is closed by Mie-Grünneisen EOS (6.17-6.19).

7.1 Incremental constitutive law in 1D

Assuming the fact that the Jaumann derivative (6.13) is the same as the material derivative (3.2) in 1D and that deviatoric stress is initially zero, the one-dimensional version of incremental constitutive law(6.14) can be then written as [32]:

$$\frac{\mathrm{d}S_{xx}}{\mathrm{d}t} = \frac{4}{3}\mu\frac{\partial U}{\partial x}, \tag{7.4a}$$

$$\frac{\mathrm{d}S_{yy}}{\mathrm{d}t} = -\frac{2}{3}\mu\frac{\partial U}{\partial x}, \tag{7.4b}$$

$$\frac{\mathrm{d}S_{zz}}{\mathrm{d}t} = -\frac{2}{3}\mu\frac{\partial U}{\partial x}. \tag{7.4c}$$

The plasticity is again incorporated using yield criterion. Since tensor $\mathbb{S}$ in 1D reads:

$$\begin{aligned}\mathbb{S} = \begin{pmatrix} S_{xx} & 0 & 0 \\ 0 & -\frac{1}{2}S_{xx} & 0 \\ 0 & 0 & -\frac{1}{2}S_{xx} \end{pmatrix} \Rightarrow |\mathbb{S}| &= \sqrt{\operatorname{tr}(\mathbb{S}:\mathbb{S})} \\ &= \sqrt{(S_{xx})^2 + \left(-\frac{1}{2}S_{xx}\right)^2 + \left(-\frac{1}{2}S_{xx}\right)^2} \\ &= \sqrt{\frac{3}{2}}S_{xx}.\end{aligned} \tag{7.5}$$

From this, the 1D yield criterion(6.9) becomes [32]:

$$f = |S_{xx}| - \frac{2}{3}Y_0 \leqslant 0. \tag{7.6}$$

Since either Cauchy or deviatoric stress tensor can be expressed using just its S_{xx} and T components, we will, for the sake of simplicity, omit their subscripts.

7.2 Numerical method

The basic step is again given by the Lax-Wendroff method with HLL based artificial dissipation discussed in Section 3.3 with a small difference that the numerical fluxes $F_i^n, F_{i\pm\frac{1}{2}}^{n+\frac{1}{2}}$ are given by (7.3). For the computation of fluxes, one needs the values of S, which are obtained from constitutive law (7.4a) discretized using the central difference. In a predictor, the nodal estimate of S in $i+\frac{1}{2}$-th node is given by:

$$S_{i+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{\Delta x_{i+1}^n S_i^n + \Delta x_i^n S_{i+1}^n}{\Delta x_i^n + \Delta x_{i+1}^n} + \frac{1}{2}\Delta t\frac{4}{3}\mu\frac{u_{i+\frac{1}{2}}^{n+\frac{1}{2}} - u_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x_{i+\frac{1}{2}}^n}, \tag{7.7}$$

where the first term of the right-hand side is the nodal mass average of the strain rate similar to the WW predictor (3.27), and the second term is a central difference approximation of (7.4a). The value of S in i-th cell is computed using corrector:

$$S_i^{n+1,tr} = S_i^n + \Delta t\frac{4}{3}\mu\frac{u_{i+\frac{1}{2}}^{n+\frac{1}{2}} - u_{i-\frac{1}{2}}^{n+\frac{1}{2}}}{\Delta x_i^n}. \tag{7.8}$$

7.2.1 Radial return algorithm

Formulas (7.7) and (7.8) give us the trial value of deviatoric stress S, which is denoted by the upper index tr. These values describe the evolution of the purely elastic flow. To incorporate plasticity, one has to enforce the von Mises yield criterion (7.6). For this purpose, we will use a simple radial return algorithm originally proposed by Wilkins [3]. The idea of the radial return algorithm is to limit the value of deviatoric stress, so the criterion (6.9) is not violated. In 1D, the new deviatoric stress is defined by:

$$S_i^{n+1} = \operatorname{sign}(S_i^{n+1,tr}) \max\left\{ |S_i^{n+1,tr}|, \frac{2}{3} Y_0 \right\}. \tag{7.9}$$

7.2.2 Timestep

The time step for 1D elastic-perfectly plastic flow is computed similarly to 1D hydrodynamics defined by (3.23) with the only difference that instead of sound speed (2.57), we use longitudinal elastic wave speed, which writes

$$a_i^2 = ((c_s)_i)^2 + \frac{4}{3} \frac{\mu}{\varrho_i}, \tag{7.10}$$

where $(c_s)_i$ is the sound speed (6.18) obtained from Mie-Grünneissen EOS. Therefore, the time step reads:

$$\Delta t^{n+1} = C_{\mathrm{CFL}} \min_{i \in \hat{N}} \left\{ \frac{\Delta x_i^n}{a_i^n} \right\}. \tag{7.11}$$

Similarly to 1D hydrodynamics, the time step (7.10) is additionally restricted by the maximum allowable change of the cell length.

7.2.3 Recapitulation of a single step

For the sake of simplicity, let us now shortly summarize a single step of the proposed method.

1. Starting with cell data at time level n, we compute a new timestep using (7.11).
2. Then the predicted nodal values are evaluated $\boldsymbol{w}_p^{n+\frac{1}{2}}$ using (3.27).
3. When we have predicted nodal velocity, we can compute nodal stress from (7.7).
4. Nodal thermodynamical quantities (c_s,p,ε) can be now updated by EOS (6.17) and (6.18).
5. Knowing all the nodal data at $t^{n+\frac{1}{2}}$, the new cell state values $\boldsymbol{w}_c^{n+1}$ are computed using corrector (3.21).
6. Trial cell stress is updated using incremental constitutive law (7.8).
7. The final cell stress computed from the trial one using the radial return algorithm (7.9).

8. Artificial dissipation and limiter terms are computed using (3.50) and (3.55) and applied to $\boldsymbol{w}^{n+1}$.
9. The nodal positions are advanced by Δt^{n+1} to time level t^{n+1} using nodal velocities $\boldsymbol{U}_p^{n+\frac{1}{2}}$, geometry structures (cell centers and volumes) are updated using (3.5 - 3.6).
10. Cell densities are recomputed from the new geometry using (3.7).
11. Cell thermodynamical quantities are updated from EOS.
12. Boundary conditions are applied.

7.3 Numerical results

7.3.1 Elastoplastic piston

The first test is taken from [32], [122], and it is similar to the classical hydrodynamic piston problem, unlike from which it consists of two waves – a small elastic precursor and a “main” plastic wave.

Let us consider a 1D copper wire initially at rest characterized by material parameters presented in Table 6.1. The wire is at the beginning 1 cm long, and the pressure at $t = 0$ is $P_0 = 10^5$ Pa. There is a piston-like moving boundary condition characterized by velocity $u_{\text{piston}} = 20$ m/s prescribed on the left, and the wall boundary on the right. The final time is $T = 150 \times 10^{-6}$ s.

This test was computed on initially equidistant meshes consisting of 100, 200, and 400 cells using the pure LW scheme, i.e. no artificial dissipation was included ($\tau_d = \tau_u = \tau_E = 0$). The results, together with the analytical solution, are presented in Figure 7.1. Several numerical experiments have shown that in this and the following test, the dissipation due to radial return algorithm is sufficient for the mitigation of oscillations caused by LW. For this reason, the τ is here set to zero. The resolution of the elastic precursor is similar to the methods [32], [111].

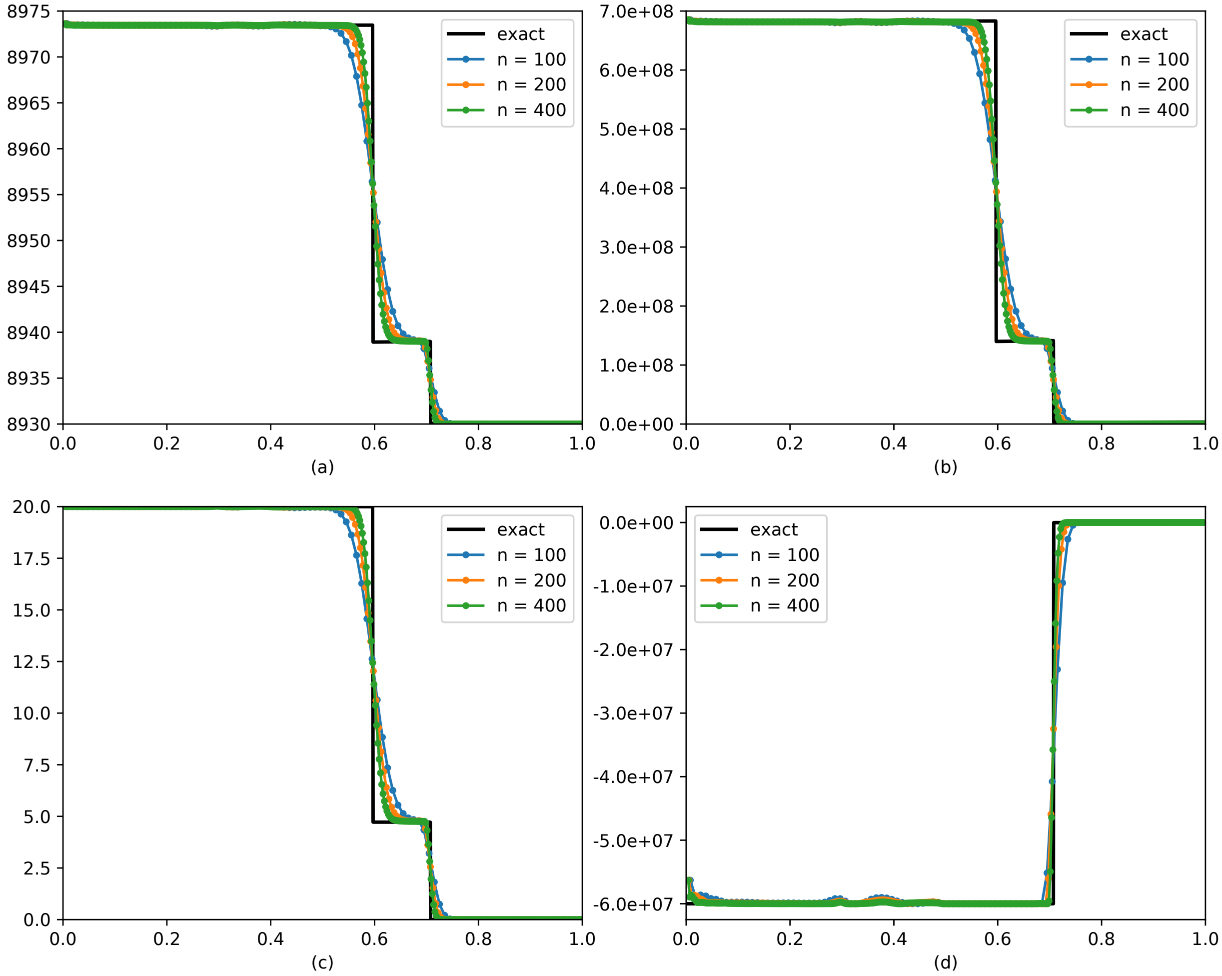


Figure 7.1: Convergence for the elastoplastic piston at $t = 150 \times 10^{-6}$ s for different resolutions; density (a); pressure (b); velocity (c); deviatoric stress (d).

7.3.2 Wilkins problem

The next problem was firstly introduced by Wilkins in [3]. It deals with a 1D thin flying Aluminium projectile that strikes the thick Aluminium target that is initially at rest. The material parameters for Aluminium can be again found in Table 6.1. At $t = 0$ the computational domain is given by interval $\langle 0, 0.05 \rangle$ cm divided at $x_0 = 0.005$ cm into two states. The left state represents the projectile with the initial velocity $u_0 = 800$ m/s and pressure $P_0 = 10^{-6}$ Pa. The right state representing the thick target has at the beginning zero velocity and pressure $P_0 = 10^{-6}$ Pa. The final time is $T = 5 \cdot 10^{-6}$ s. The free surface boundary conditions are prescribed on both ends. The result for a mesh consisting of 500 initially equidistant cells can be found in Figure 7.2. Similarly to the previous test, $\tau = 0$. Compared with [32], the results are reasonable.

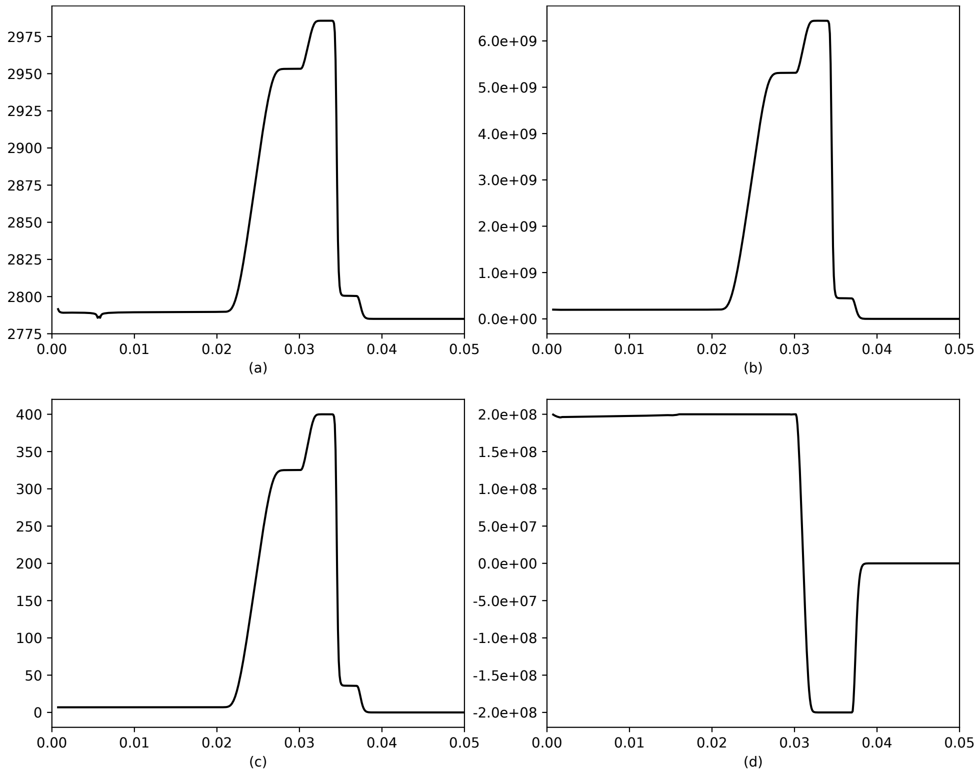


Figure 7.2: Wilkins problem by LW+1 scheme; density (a); pressure (b); velocity (c); deviatoric stress (d).

7.3.3 Purely elastic impact

This test is similar to the Wilkins problem presented in the previous paragraph 7.3.2, with the only difference that the radial return algorithm (7.9) has not been applied, which means that the flow remains purely elastic even when von Mises yield condition should apply. Turning the radial return algorithm off is equivalent to artificially setting Y_0 sufficiently high. In Figure 7.3, one can see that the plastic wave has not developed and, what is more important, that the oscillations due to the LW scheme occur behind the shock wave and even on the tail of rarefaction. When the artificial viscosity (i.e., LW+1 scheme) is turned on with $\tau = 0.2$, the oscillations due to LW dispersion are no longer present. From this rather synthetic test, we can conclude that the radial return algorithm can itself provide enough dissipation to prevent the oscillations due to LW.

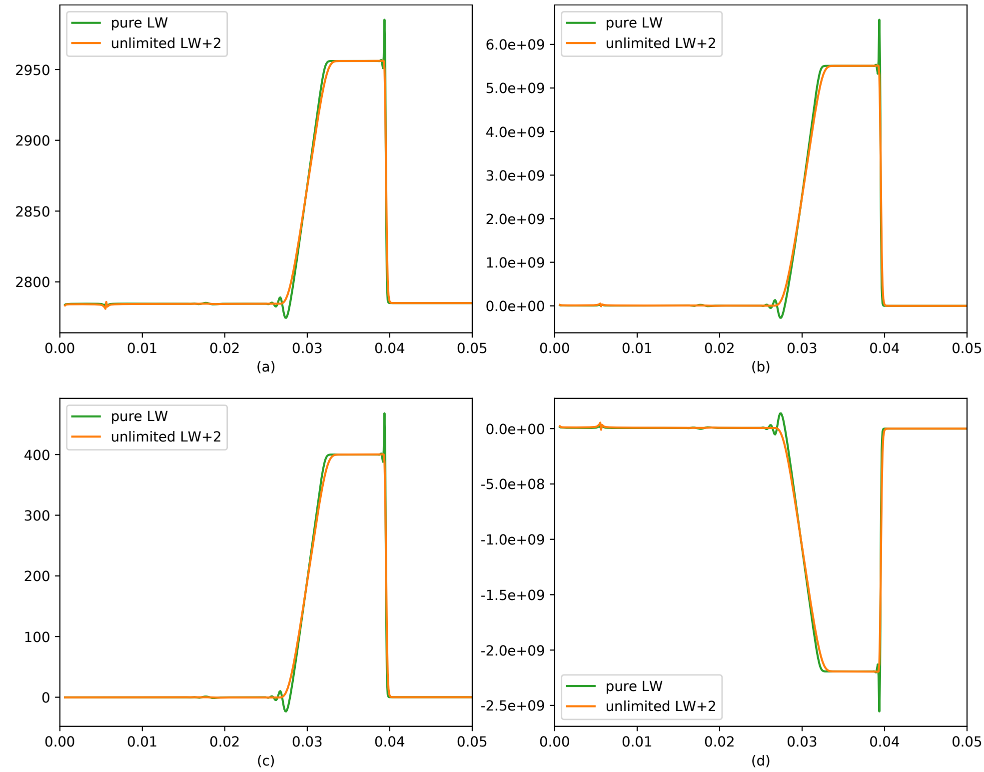


Figure 7.3: Wilkins problem without radial return computed by pure LW scheme and LW+1 with $\tau = 0.2$; density (a); pressure (b); velocity (c); deviatoric stress (d).

Chapter 8

Method for 2D Elastic-Plastic flow

The conservation laws part of the Wilkins model is described by (2.52) with physical fluxes (6.3), which are directly discretized using (4.7) and (4.9). The only specific is the deviatoric stress $\mathbb{S}$, which will be described in the following section.

8.1 Incremental constitutive law in 2D

Deviatoric stress tensor $\mathbb{S}$ in 2D geometry can be written as:

$$\mathbb{S} = \begin{pmatrix} S_{xx} & S_{xy} & 0 \\ S_{xy} & S_{yy} & 0 \\ 0 & 0 & -S_{xx} - S_{yy} \end{pmatrix}. \tag{8.1}$$

Because the stress tensor is no longer given by just a single component S_{xx}, as in the 1D case, the simplification described in the previous chapter is no longer applicable. Therefore we have to solve the equation (6.14) directly, and the method for 2D has to be different. The full discretization of the elastic step of incremental constitutive law is taken from [32]. We want to solve the following equation for $t \in \langle t^n, t^{n+k} \rangle$, for $k = \frac{1}{2}$ or 1:

$$\overset{\circ}{\mathbb{S}}_c = 2\mu \mathbb{D}_{0c} \tag{8.2}$$

with initial condition $\mathbb{S}_c = \mathbb{S}_c^n$, where $\mathbb{D}_{0c}$ is deviatoric part of strain rate tensor given by (6.7) in cell c. The equation (8.2) is, as shown in [108], equivalent to the ordinary differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t}(\Omega_c^t \mathbb{S}_c \Omega_c) = 2\mu_c \Omega_c^t \mathbb{D}_{0c} \Omega_c, \tag{8.3}$$

with the initial condition $\Omega_c(t^n) = \mathbb{I}$, where Ω_c is the rotation tensor given by the solution of ODE (8.3):

$$\frac{\mathrm{d}}{\mathrm{d}t}\Omega_c = \mathbb{W}_c \Omega_c \tag{8.4}$$

and $\mathbb{W}_c$ is the value of spin tensor (6.6) in cell c. The solution of this equation is

$$\Omega_c(t) = \exp[(t - t^n)\mathbb{W}_c^n]. \tag{8.5}$$

If we integrate (8.3) over the interval $\langle t^n, t^n + k\Delta t\rangle$, we obtain

$$\Omega_c^t(t^{n+k})\mathbb{S}_c^{n+k}\Omega_c(t^{n+k}) - \mathbb{S}_c^n = 2\mu_c \int_{t^n}^{t^n+k\Delta t} \Omega_c^t(t)\mathbb{D}_{0c}\Omega_c(t)\mathrm{d}t. \tag{8.6}$$

After substituting (8.5) into (8.3) and approximation of the integral by the midpoint formula, one gets:

$$\begin{aligned} &\exp\left(-k\Delta t\mathbb{W}_c\right)\mathbb{S}_c^{n+k}\exp\left(k\Delta t\mathbb{W}_c\right) - \mathbb{S}_c^n = \\ &= 2\mu k\Delta t \exp\left(-\frac{k\Delta t}{2}\mathbb{W}_c\right)\mathbb{D}_{0c}\exp\left(\frac{k\Delta t}{2}\mathbb{W}_c\right), \end{aligned} \tag{8.7}$$

from which we can extract the unknown $\mathbb{S}_c^{n+k}$:

$$\begin{aligned} \mathbb{S}_c^{n+k} &= \exp\left(k\Delta t\mathbb{W}_c\right)\mathbb{S}_c^n\exp\left(-k\Delta t\mathbb{W}_c\right) \\ &+ 2\mu k\Delta t \exp\left(\frac{k\Delta t}{2}\mathbb{W}_c\right)\mathbb{D}_{0c}\exp\left(-\frac{k\Delta t}{2}\mathbb{W}_c\right). \end{aligned} \tag{8.8}$$

The exponentials of tensors in (8.8) are approximated using Padé approximant $\mathcal{Q}[\mathbb{W}](t)$:

$$\exp(t\mathbb{W}_c) \approx \mathcal{Q}[\mathbb{W}_c](t) = \left(\mathbb{I} - \frac{t}{2}\mathbb{W}_c\right)^{-1}\left(\mathbb{I} + \frac{t}{2}\mathbb{W}_c\right). \tag{8.9}$$

By substituting approximation (8.9) into (8.8), one gets the final expression for $\mathbb{S}_c^{n+k}$ dependent only on known variables $k, \mathbb{S}_c^n, \mathbb{D}_{0c}$ and $\mathbb{W}_c$:

$$\begin{aligned} \mathbb{S}_c^{n+k}(k, \mathbb{S}_c^n, \mathbb{D}_{0c}, \mathbb{W}_c) &= \mathcal{Q}[\mathbb{W}_c]\left(k\Delta t\right)\mathbb{S}_c^n\mathcal{Q}^t[\mathbb{W}_c]\left(k\Delta t\right) \\ &+ 2\mu k\Delta t\mathcal{Q}[\mathbb{W}_c]\left(\frac{k\Delta t}{2}\right)\mathbb{D}_{0c}\mathcal{Q}^t[\mathbb{W}_c]\left(\frac{k\Delta t}{2}\right), \end{aligned} \tag{8.10}$$

which is used in both predictor and corrector substeps.

8.1.1 Discretization of conservation laws

The discretization of conservation laws for EP flow is the same as the 2D hydrodynamics described in Chapter 4 with a small difference in the definition of physical flux with is now governed by (6.3). Therefore the reader is kindly referred to Section 4 for the details.

8.1.2 Predictor for strain rate tensor

The values of the stress tensor $\mathbb{D}$ and spin tensor $\mathbb{W}$ in node p at time level n are obtained by integration over dual cell p using compatible discretization [32]:

$$\mathbb{D}_p^n = \frac{1}{V_p}\sum_{c(p)}\frac{1}{2}\left(\boldsymbol{U}_c^n \otimes \boldsymbol{n}_{spc}^n + \boldsymbol{n}_{spc}^n \otimes \boldsymbol{U}_c^n\right), \tag{8.11}$$

$$\mathbb{W}_p^n = \frac{1}{V_p}\sum_{c(p)}\frac{1}{2}\left(\boldsymbol{U}_c^n \otimes \boldsymbol{n}_{spc}^n - \boldsymbol{n}_{spc}^n \otimes \boldsymbol{U}_c^n\right), \tag{8.12}$$

where $\boldsymbol{n}_{spc}^n = \boldsymbol{n}_{spc+}^n + \boldsymbol{n}_{spc-}^n$. The deviatoric part of the strain rate tensor (8.11) is computed using

$$\mathbb{D}_{0p}^n = \mathbb{D}_p^n - \frac{1}{3}\mathrm{tr}(\mathbb{D}_p^n)\mathbb{I}, \tag{8.13}$$

where the trace of $\mathbb{D}_p^n$ is given by:

$$\mathrm{tr}(\mathbb{D}_p^n) = \frac{1}{V_p^n}\sum_{c(p)}\boldsymbol{U}_c^n \cdot \boldsymbol{n}_{spc}^n. \tag{8.14}$$

To advance the deviatoric stress in dual cell p, using (8.10), one has to interpolate cellular values $\mathbb{S}_c^n$ into node p, for which we again use Wendroff-White weighting:

$$\mathbb{S}_p^n = \frac{\sum_{c(p)} \mathbb{S}_c^n / A_{pc}^n}{\sum_{c(p)} 1/A_{pc}^n} \tag{8.15}$$

Then we can finally use (8.10) to advance $\mathbb{S}_p^n$:

$$\mathbb{S}_p^{n+\frac{1}{2}} = \mathbb{S}_p(\frac{1}{2}, \mathbb{S}_p^n, \mathbb{D}_{0p}^n, \mathbb{W}_p^n). \tag{8.16}$$

Let us remark that the Wilkins' radial return algorithm is not applied to $\mathbb{S}_p^{n+\frac{1}{2}}$ in the predictor phase because it would violate the evaluation of the trial elastic value in the corrector.

8.1.3 Corrector for strain rate tensor

The corrector phase for $\mathbb{S}_c^{n+1}$ is very similar to the predictor. The stress tensor $\mathbb{D}$ and spin tensor $\mathbb{W}$ in cell c at time $t^{n+\frac{1}{2}}$ are:

$$\mathbb{D}_c^{n+\frac{1}{2}} = \frac{1}{V_c}\sum_{p(c)}\frac{1}{4}\left(\boldsymbol{U}_p^{n+\frac{1}{2}} \otimes \boldsymbol{n}_{pc}^{n+\frac{1}{2}} + \boldsymbol{n}_{pc}^{n+\frac{1}{2}} \otimes \boldsymbol{U}_p^{n+\frac{1}{2}}\right), \tag{8.17}$$

$$\mathbb{W}_c^{n+\frac{1}{2}} = \frac{1}{V_c}\sum_{p(c)}\frac{1}{4}\left(\boldsymbol{U}_p^{n+\frac{1}{2}} \otimes \boldsymbol{n}_{pc}^{n+\frac{1}{2}} - \boldsymbol{n}_{pc}^{n+\frac{1}{2}} \otimes \boldsymbol{U}_p^{n+\frac{1}{2}}\right), \tag{8.18}$$

where

$$\boldsymbol{n}_{pc}^{n+\frac{1}{2}} = (\boldsymbol{n}_{e(ca)}^{n+\frac{1}{2}} + \boldsymbol{n}_{e(cb)}^{n+\frac{1}{2}})/2$$

is the sum of vectors normal to edges $e(ca)$ and $e(cb)$ of cell c adjacent to node p. The deviatoric part of stress tensor $\mathbb{D}_c^n$ is given by:

$$\mathbb{D}_{0c}^{n+\frac{1}{2}} = \mathbb{D}_c^{n+\frac{1}{2}} - \frac{1}{3}\mathrm{tr}(\mathbb{D}_c^{n+\frac{1}{2}})\mathbb{I}, \tag{8.19}$$

where

$$\mathrm{tr}(\mathbb{D}_c^{n+\frac{1}{2}}) = \frac{1}{V_c^{n+\frac{1}{2}}} \sum_{c(p)} \boldsymbol{U}_p^{n+\frac{1}{2}} \cdot \boldsymbol{n}_{pc}^{n+\frac{1}{2}}. \tag{8.20}$$

Now we can evaluate the trial elastic value of strain rate tensor by applying (8.10):

$$\mathbb{S}_c^{n+1,tr} = \mathbb{S}_c^{n+1}(1, \mathbb{S}_c^n, \mathbb{D}_{0c}^{n+\frac{1}{2}}, \mathbb{W}_c^{n+\frac{1}{2}}). \tag{8.21}$$

The final value for strain rate tensor $\mathbb{S}_c^{n+1}$ which takes into account the plasticity described by von Mises criterion in the form

$$f(\mathbb{S}_c^{n+1}) = |\mathbb{S}_c^{n+1}| - \sqrt{\frac{2}{3}} Y_0$$

is computed by the radial return algorithm, which is given by the following simple limiter applied to elastic trial strain rate:

$$\mathbb{S}_c^{n+1} = \begin{cases} \mathbb{S}_c^{n+1,tr} & \text{if} f(\mathbb{S}_c^{n+1,tr} \leqslant 0), \\ \sqrt{\frac{2}{3}} Y_0 \frac{\mathbb{S}_c^{n+1,tr}}{|\mathbb{S}_c^{n+1,tr}|} & \text{otherwise.} \end{cases} \tag{8.22}$$

8.1.4 Timestep

The main restriction of the time step for 2D elastic plastic flow is given by the CFL condition (4.34) and the definition (7.10):

$$\Delta t_{\mathrm{CFL}}^n = C_{\mathrm{CFL}} \min_c \frac{l_c^n}{a_c^n}, \tag{8.23}$$

where

$$a_c^2 = (c_s)_c^2 + \frac{4}{3} \frac{\mu}{\varrho_c}$$

is again the longitudinal elastic wave speed.

8.2 Numerical results

In this section, we will present numerical results for 2D elastic-plastic flow. In contrast with [65], all here presented results were computed using predictor (4.7).

Let us first define a dimensionless coefficient called *plasticity threshold* similarly to [32]:

$$p_t = \sqrt{\frac{3}{2}} \frac{|\mathbb{S}|}{Y_0}, \tag{8.24}$$

which refers to the von Mises yield criterion (6.9). In the regions where the deformations are plastic, its value is 1, and in regions of elastic deformations, its value is less than 1.

8.2.1 Elastic vibration of Beryllium plate

This test has been firstly introduced in PAGOSA [123] and deals with purely elastic oscillations of a thin Beryllium plate. The material constants can be again found in Table 6.1, except that the yield strength Y_0 is artificially set to 10^{11} Pa to keep the flow purely elastic. The computational domain is a 2D region $\langle -5, 5\rangle \times \langle -30, 30\rangle$ mm. The x-component of velocity is initially zero, and the y-component is given by:

$$v(0, x) = A\omega \left\{ g_1[\sinh(\xi) + \sin(\xi)] - g_2[\cosh(\xi) + \cos(\xi)] \right\},$$

where $\xi = \Omega(x + 0.03)$, $\Omega = 78.83401241\text{m}^{-1}$ is the first nonzero root of equation $\cosh(0.06\Omega)\cos(0.06\Omega) = 1$. The coefficients g_1, g_2, A are

$$g_1 = \cosh(0.06\Omega) - \cos(0.06\Omega) \approx 56.63685154, \tag{8.25a}$$

$$g_2 = \sinh(0.06\Omega) - \sin(0.06\Omega) \approx 57.64552048, \tag{8.25b}$$

$$A = \frac{y_{max}}{2g_2} \approx 0.04336850425\,\text{mm}, \tag{8.25c}$$

$y_{max} = 5$ mm is the maximum displacement in the y-direction, and

$$\omega = \Omega^2 \Delta y \sqrt{\frac{E}{12\varrho(1 - \nu^2)}}\,\text{Hz}$$

is the frequency-dependent on elastic modulus $E = 3.182656 \times 10^{11}$ Pa, Poisson ratio $\nu = 0.053896$, and the thickness of the plate $\Delta y = 0.01$ m. Initially, the pressure P^0 and the strain tensor $\mathbb{S}^0$ are set to zero. The analytical solution for a thin plate gives a time period for the first flexural moment $2\pi/\omega \approx 26.6\,\mu$s [40]. However, the exact solution for the thick plate does not exist. The final time of the simulation is $T_f = 100 \times 10^{-6}$ s. The computational domain is discretized by 100×16 initially rectangular, equally spaced cells. Figure 8.1 shows the time evolution (history) of the y-coordinate of the slab center, its velocity, and the conversion between internal and kinetic energy. The observed period is approximately $30\,\mu$s, which is in good agreement with the results presented in [32], [40]. The meshes at different fractions of the observed period T_p can be seen in Figure 8.2.

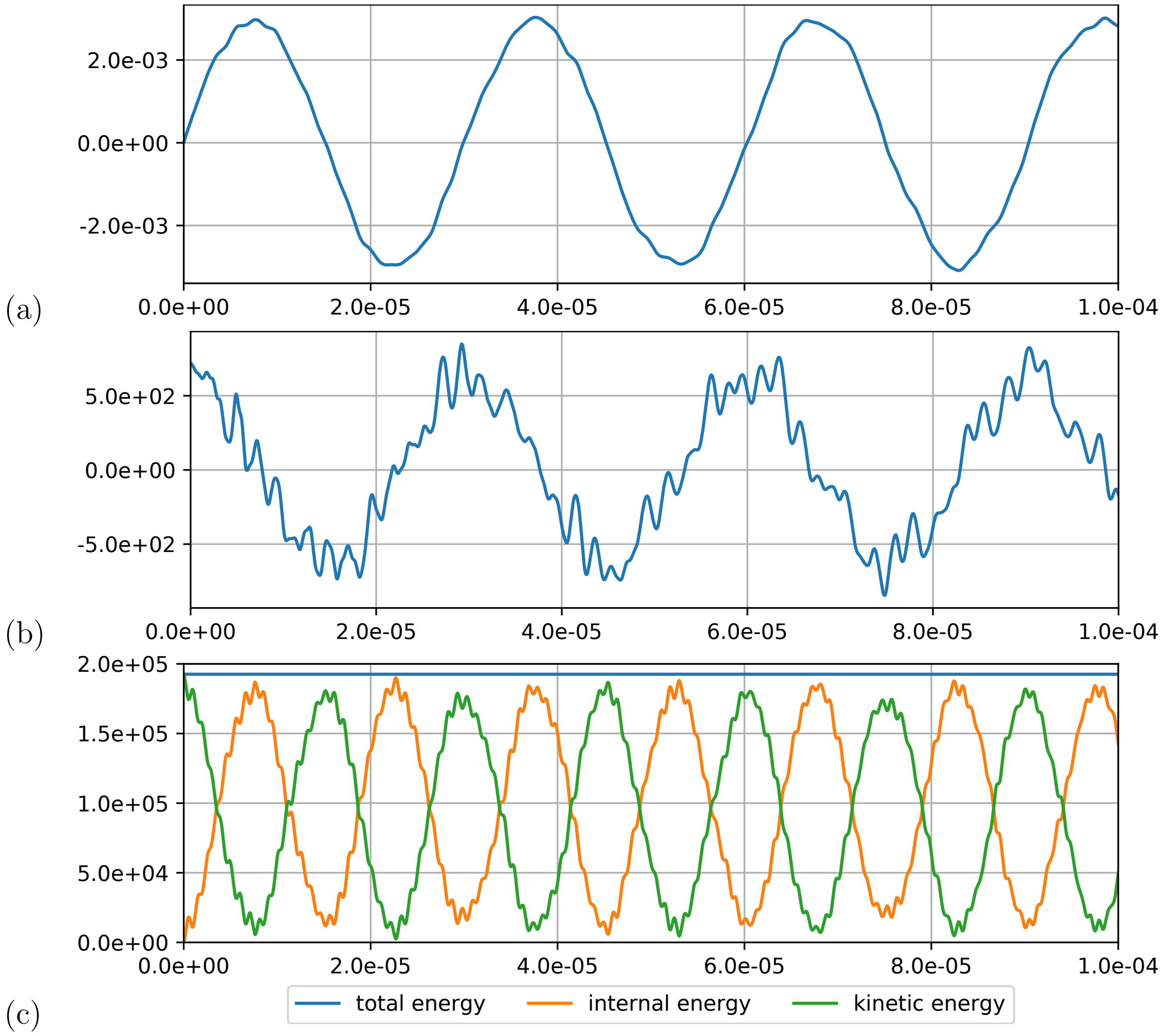


Figure 8.1: Time evolution of the vibrating Beryllium plate: the y-position of the plate center (a); the y-velocity component of the plate center (b); conversion between kinetic and internal energies (c).

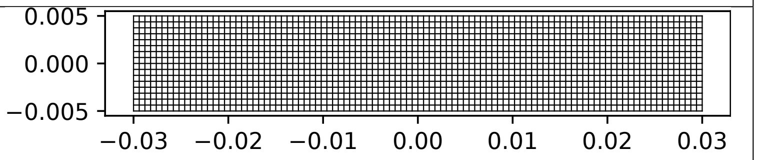
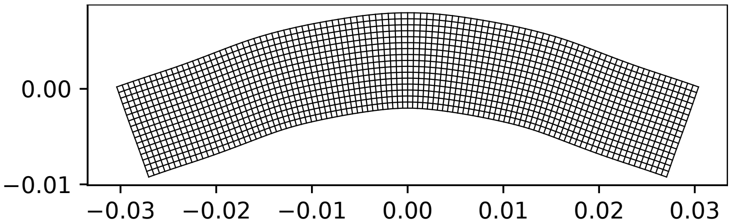
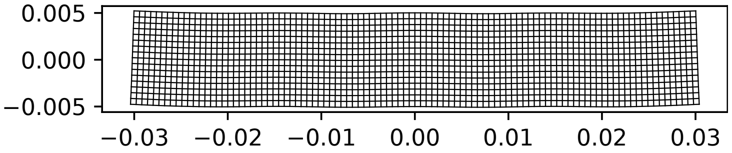
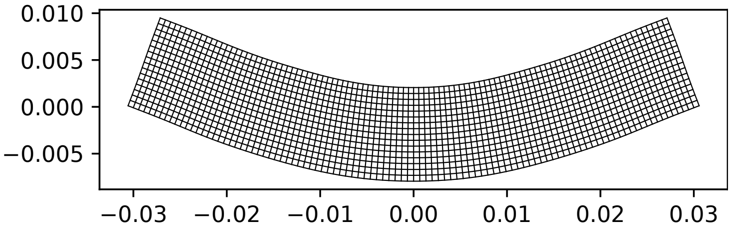
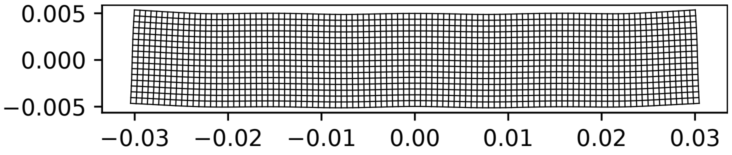
time t	fraction of T_p	mesh
0 s	0	
7.7×10^{-6} s	1/4	
15.2×10^{-6} s	1/2	
21.7×10^{-6} s	3/4	
29.9×10^{-6} s	1	

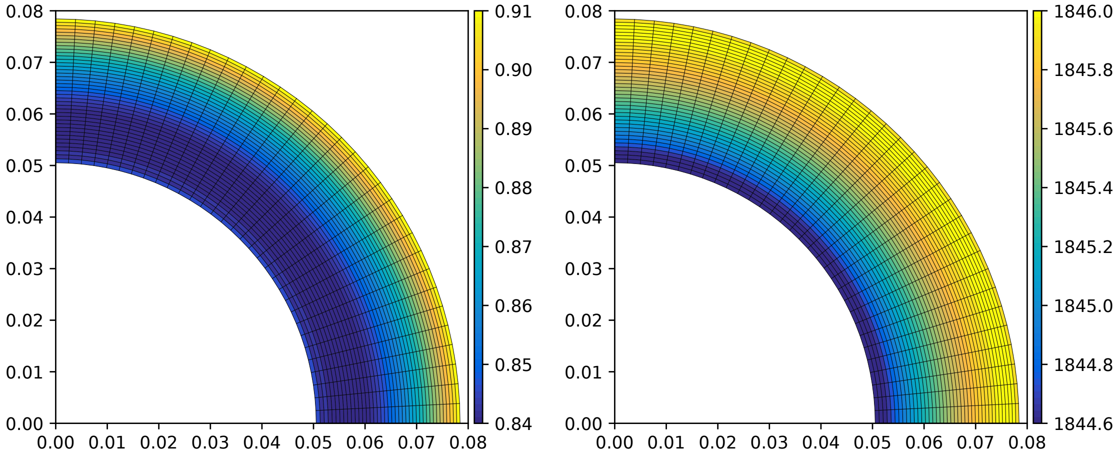
Figure 8.2: The evolution of the mesh for the Beryllium plate problem at different fractions of observed period T_p.

8.2.2 Collapse of Beryllium shell

This test taken from [109], [124] deals with a collapse of the Beryllium shell. Although the originally proposed version used different equations of state, the use of Mie-Grünneisen EOS has also been studied, showing similar results [32]. The Beryllium shell with the inner radius $R_\mathrm{i}^0 = 0.08$ m, and outer radius $R_\mathrm{o}^0 = 0.1$ m with the initial state defined by the density ϱ^0 from Table 6.1, pressure P^0, and velocity field $\boldsymbol{U}^0$ depending on radius $r = \sqrt{x^2 + y^2}$ defined by

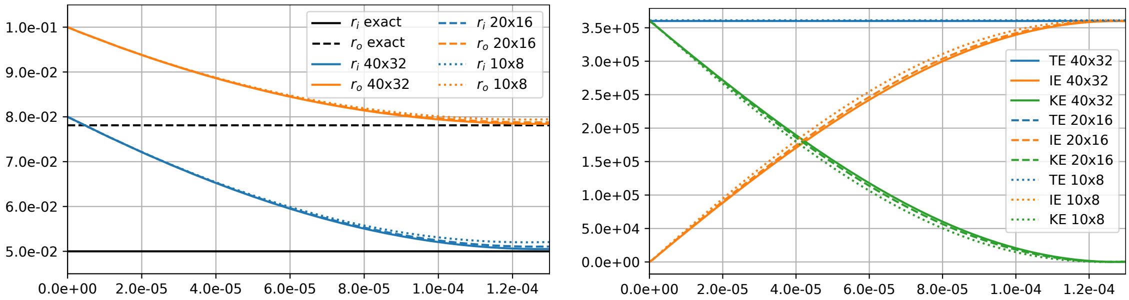
$$\boldsymbol{U}^0(r) = -U_\mathrm{i}\frac{R_\mathrm{i}}{r^2}\boldsymbol{r},$$

where $\boldsymbol{r}$ is the position vector from the origin and $U_\mathrm{i} = 417.1$ m/s is an axial velocity reached at the inner radius R_i. Assuming incompressibility, one can obtain analytic solutions for the inner and the outer radii, which are at time $T = 150 \times 10^{-6}$ s, $R_\mathrm{i} = 0.05$ m and $R_\mathrm{o} = 0.0781$ m. The problem was computed on a single quarter of the shell with free surface boundary conditions on inner and outer circle and with reflecting boundary conditions on x- and y- axes. The resolutions in circular × angular directions were $8 \times 10, 16 \times 20$, and 32×40 cells. The plasticity threshold map (8.24) for the fine grid can be seen in Figure 8.3, while the history of radii positions and energy conversion are presented in Figure 8.4. The radii positions converge to the exact ones, while the total energy is well conserved.



(a) (b)

Figure 8.3: Collapse of Beryllium shell: the plasticity threshold (a) and the density (b) on mesh with 32×40 cells.



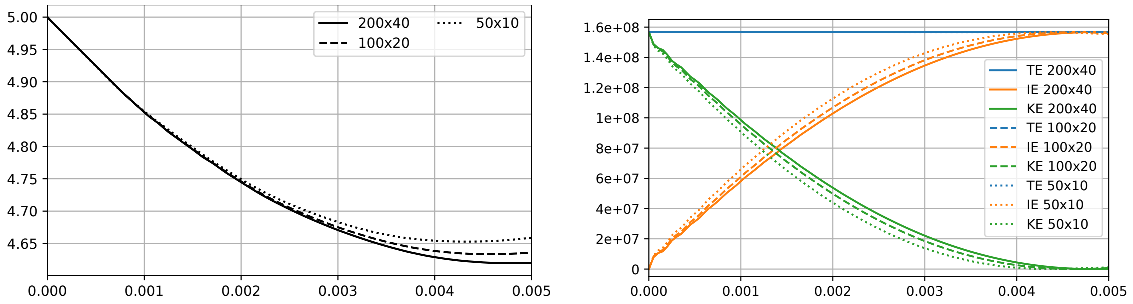
(a) (b)

Figure 8.4: Evolution of radius (a) and energy (b) for the collapse of Beryllium shell.

8.2.3 Cartesian Taylor bar impact

This test, which is a Cartesian analogy of Taylor bar impact [125], has been proposed in [40]. It models a 2D rectangular Aluminium projectile, which is initially a plate 5m long and 1m high. The initial computational domain is hence given by a 2D region $\langle 0, 5\rangle \times \langle -0.5, 0.5\rangle$ m filled with Aluminium given by the density ϱ^0, which can be found in Table 6.1 and pressure $P^0 = 10^{-6}$ Pa. The final time is $T = 5$ ms. Free boundaries are prescribed everywhere except the left boundary, which is treated as a wall. The problem has been computed on initially rectangular meshes of $50 \times 10, 100 \times 20, 200 \times 40$ cells with $C_{\mathrm{CFL}} = 0.3$ by pure LW scheme and LW+1 with $\tau = 0.1$.

Figure 8.6 shows the plasticity threshold map for pure LW, limited and unlimited LW+1. In Figure 8.5, one can find the time evolution of bar length and kinetic-to-internal energy conversion history. The result computed by pure LW in Fig. 8.6 contains spurious checkerboard-like oscillations near yielding level $x \approx 0.75$ and near the left (West) boundary, which means that the dissipation due to the radial return algorithm is no longer sufficient to mitigate the LW dispersion. The mesh near the origin is, in this case, denser as this region is at the final time fully plastic, which is different from the results due to [32]. Figure 8.6(b) then shows the results due to unlimited LW+1 with very low $\tau = 0.1$, which is enough to mitigate spurious checkerboard pattern. The region near the origin is, in this case, not fully yielded, which is closer to the reference. The last Figure 8.6(c) shows the result due to LW+1 with $\tau = 0.1$ and 9 edges limiter. As one can see, the region near the origin is not fully yielded, but the value of the plasticity threshold there is higher. The higher modes (at approximately at $x \in \langle 0.75, 2.2\rangle$ m) of elastic waves are closer to the pure LW case. The features have more details for pure LW and fewer details for unlimited LW+1. Due to the lack of an analytical solution for this test and the observed fast evolution of these features, it is hard to say which are closer to the physical reality. Both limited and unlimited cases are somehow comparable to the results due to EUCCLHYD[32].



(a) (b)

Figure 8.5: Evolution of length for the Taylor bar impact (a) and evolution of energy (b) for unlimited LW+1.

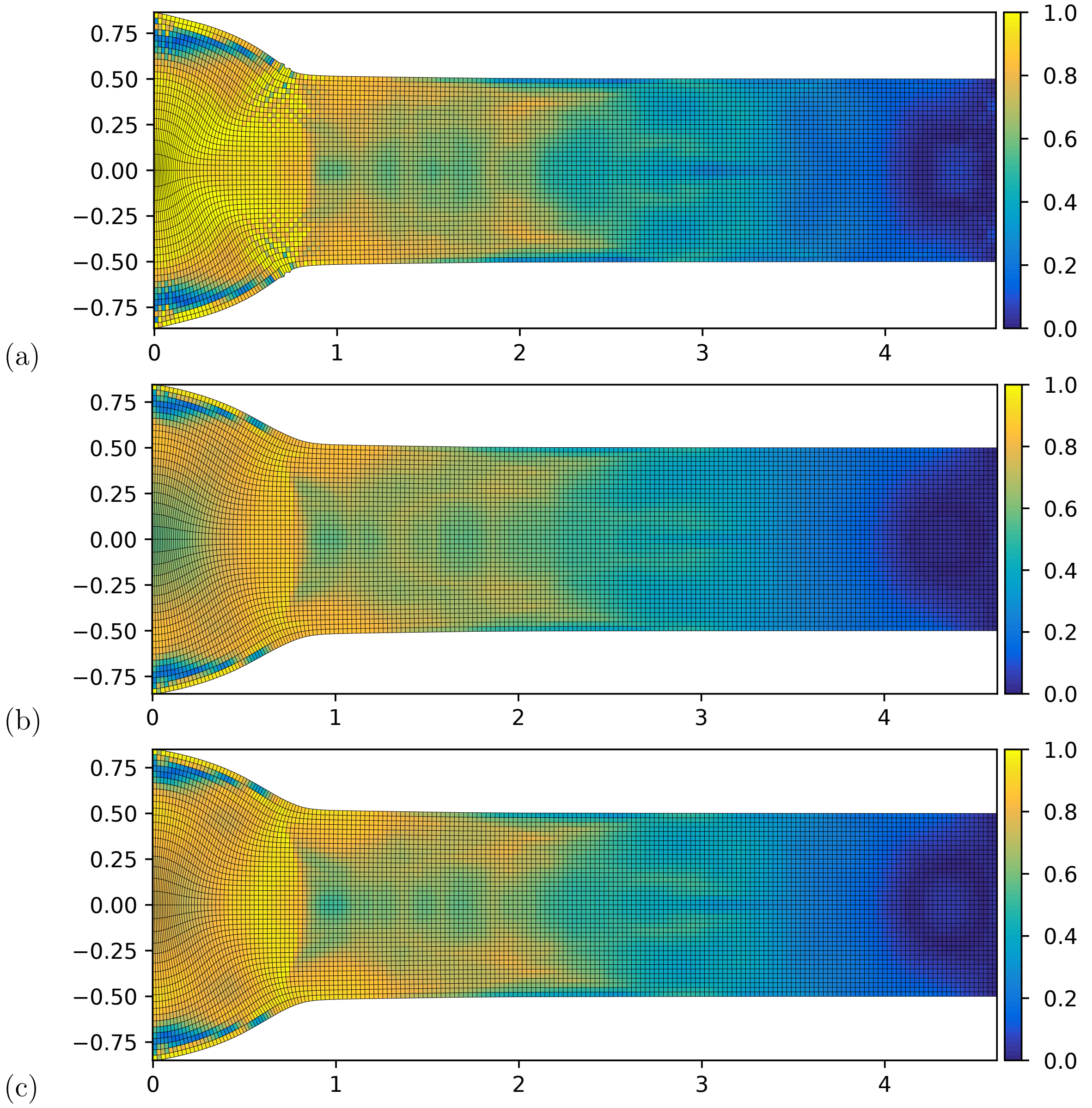


Figure 8.6: Taylor bar problem on the 200 × 40 cells mesh; the plasticity threshold by pure LW (a), unlimited LW+1 with $\tau = 0.1$ (b), limited LW+1 with $\tau = 0.1$ (c).

Chapter 9

Conclusion

The main objective of this thesis was to develop and study a new method for simulations of systems of conservation laws, specifically hydrodynamics, described using the system of Euler equations for inviscid compressible fluids closed by an ideal gas equation of state and solid dynamics, which is described by the Wilkins model.

We have presented a novel cell-centered Lagrangian method originally published in [64], later applied to elastoplasticity in [65], implemented on unstructured meshes [68], and most recently extended to cylindrical rz geometry in [67]. The scheme consists of Richtmyer's finite volume formulation of the Lax-Wendroff scheme used to discretize physical fluxes, and Harten-Lax-van Leer (HLL) based artificial dissipation.

The proposed artificial dissipative flux can be applied not only to the momentum equation, forming artificial viscosity in the traditional sense but also to the energy and specific volume equations. The addition of artificial energy flux into the energy equation reduces the effect of well-known wall heating phenomena and improves the symmetry. The application of artificial mass flux violates the GCL law and requires an alternative definition of density to keep the GCL error low. The amount of dissipation being applied to each equation can be changed using a single parameter τ. For the hydrodynamics, adding of artificial viscosity and energy flux, called LW+2, has been observed to perform best, while adding a significantly lower amount of only artificial viscosity has worked fine for solid dynamics.

The original method [66] has been further improved in several directions, namely the inverse area weighting of conserved quantities in predictor inspired by [76], which has significantly improved the behavior on highly nonuniform meshes, simplified *interface fix* recovering material interfaces, and contact discontinuities, and finally, the introduction of limiters, that helps to reduce the amount of dissipation being added in regions with expanding cells or smooth solution.

Concerning the solid dynamics, the elastic evolution of deviatoric stress is computed by the method due to Maire et al. [29], while the plastic behavior is computed by Wilkins' radial return algorithm [1] enforcing von Mises yield criterion.

Good overall performance has been demonstrated on several tests. Namely, on the 1D problem with initially smooth data [44], we have shown that the LW scheme achieves second order of accuracy and demonstrated the effect of Wendroff-White averaging of conserved quantities in the predictor. On the Noh problem, we have studied the effect of different choices of τ and different versions of the LW+n methods family. The Sod shock tube helped us to illustrate the effects of the *interface fix* and the dissipation limiting, and the Woodward-Colella test demonstrated the ability of the scheme to cope with interactions and reflections of multiple shock waves.

In 2D hydrodynamics, the numerical order of accuracy of LW has been assessed on the Kidder test. The performance of the method with artificial dissipation has been then tested on Noh and Sedov tests, where we focused mainly on symmetry, and later on Saltzman and Triple point problems assessing the mesh distortion due to artificial dissipation and the effects of the *interface fix* in multiple dimensions. In the cylindrical rz geometry, we have studied the symmetry and order of accuracy on Free expansion, Coggeshall, and Kidder problems and later demonstrated the artificial viscosity performance on spherical Noh, Sedov, Sod, and triple point problems.

The studied method performed well for most of the presented hydrodynamics test cases. The symmetry in 2D Cartesian geometry is exact for the equiangular meshes, while it remains very good for initially rectangular meshes and clearly outperforms the staggered method [5] with tensor AV [9]. The use of dissipation limiting procedure inspired by [9] does not affect the symmetry on equiangular meshes when applied on circle edges only. In the case of initially rectangular mesh, the symmetry of LW+2 limited by the original 5 edges stencil limiter was worse, but it could be slightly recovered by computing limiter on 9 edges. A similar statement can also be expressed for the cylindrical rz geometry. The symmetry remains exact on equiangular meshes, which has been proven for the rz version of LW and tested on several problems. Inspired by [18], the correction term has been designed to make the artificial dissipation symmetry-preserving in rz. Again, although the exact symmetry can not be recovered in cases with initially rectangular meshes, the symmetry keeps very nice. The effect of the limiter remains the same as in Cartesian geometry.

On the other side, on test cases not concerning cylindrical or spherical phenomena, it has been shown that the usage of limiters can improve the solution, especially when the mesh is aligned with the waves, like Saltzman, Triple-point, or Sod.

The simplified *interface fix* resolves the contact discontinuity similarly to the original one, although it is much simpler.

It has been proven that the LW+1 and LW+2 methods satisfy the Geometric Conservation Law in Cartesian geometry. In the case of cylindrical geometry, GCL is satisfied if the Simpson quadrature rule is used in space and time. However, when using this quadrature in all equations, one has to use a different definition of pressure average. The formula is significantly complicated, and hence not very practical from the implementation point of view. Fortunately, the specific volume can be computed directly from volume and mass, so neither Simpson quadrature nor specific pressure average has to be used.

The results for solid dynamics have been illustrated on elastic-plastic piston and Wilkins problem in 1D, and the elastic vibration of the Beryllium plate, the collapse of the Beryllium shell, and Taylor bar impact in 2D. The method has produced reasonably good results, mostly comparable to the method of Maire et al.[32] It has been observed that even some problems involving shock waves can be treated by pure Lax-Wendroff without any dissipation because Wilkins radial return algorithm can remove the oscillations due to LW.

The proposed method is also relatively easy to implement, as it does not require any special nodal solver and can be simply applied also to the different systems of conservation laws. Therefore it is worth further study, for example, in terms of the extension to the 3D geometry, the extension of elastoplasticity to cylindrical rz geometry or other elastoplastic models, application to different systems of conservation laws (e.g., shallow-water equations), using a different equation of state, implementation or extension to ALE (Arbitrary Lagrangian-Eulerian) framework.

The results presented in this thesis were originally published in [64]–[68]. The author of this thesis has participated in developing the numerical method and was responsible mainly for its implementation and validation.

Appendix A

Boundary conditions

A.1 Ghost cells

All presented schemes and formulas assume that for any node or cell edge, the values at adjacent cells are known. That holds true inside the computational domain, but not when the node or cell edge lies on the computational boundary. In such cases, some special treatment has to be done. In other words, we need to compute or prescribe nodal or edge fluxes on the boundaries.

Let us start from the 1D perspective. In such case, the computational domain is 1D interval $\langle a, b \rangle$ divided into N cells indexed $1, 2, \ldots, N$ and nodes $\frac{1}{2}, \ldots, N + \frac{1}{2}$. The boundary conditions tell us how to compute fluxes $\boldsymbol{F}_{\frac{1}{2}}$ and $\boldsymbol{F}_{N+\frac{1}{2}}$. One option is to use a special formula for computing fluxes at the boundary nodes, the second is to add one or more layers of cells (depending on the scheme stencil) outside the computational domain that allows using the same formulas as if the boundary nodes lay inside.

Since in 1D LW+n scheme is computed on three cells stencil, one layer of the extra cell is sufficient. In such case, the left ghost cell will be indexed 0 and the right $N + 1$. The situation is illustrated in Fig. A.1.

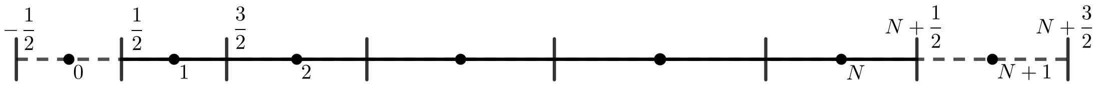
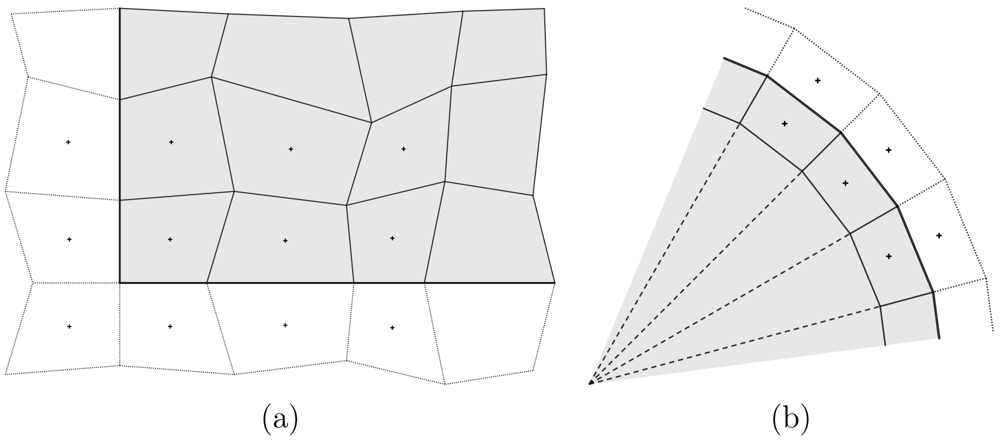


Figure A.1: Illustration of 1D mesh containing ghost cells. Inner cells are plotted by solid lines, ghost cells by dashed lines.

In 2D logically orthogonal grids, the ghost cells will make two extra rows (one on the top and one on the bottom) and two extra columns (one on the left and one on the right).

Since in Lagrangian geometry, the mesh moves with the fluid, ghost cell positions have to be updated, which is done by mirroring the nodal positions from the inside of the computational domain. For straight-line boundaries that are usually treated as the walls or piston, the nodal positions are mirrored by the boundary line, which automatically preserves the same volumes as inside. For curved boundaries, the nodal positions are mirrored by boundary node positions. Both situations are depicted in Figure A.2.

In some cases, it can be advantageous to prescribe fluxes directly, so the ghost nodes are set to have the same positions as the boundary nodes, and therefore the volumes and masses of ghost cells are zero. Therefore, the dual cell laying on boundary consists only from two inner subzones, and predictor integration (4.6) is performed only over them, which was verified to work reasonably well for example, for Kidder tests described in Sections 4.5.1 and 5.5.3, where the exact pressure (which is dependent on time) is prescribed on the boundary.



(a) (b)

Figure A.2: The part of the 2D mesh containing ghost cells. Inner cells are in grey, ghost cells in white color. Mirrored cells on initially rectangular mesh (a); ghost cells formed by mirroring inner points on polar grids (b).

Before we move on to the description of individual types of boundary conditions, let us introduce the following notation. Let us assume a cell that lays inside the computational domain Ω while at least one of its edges constitutes the domain boundary $\partial\Omega$. Such cell will be subscribed i, referring to the *inner* cell. The cell outside the computational domain Ω neighboring with cell i, which shares at least one point with it, will be subscribed by $g(i)$ referring to the *ghost* cell image of cell i. The common edge of cells i and $g(i)$ laying on the boundary subscribed by b and its endpoints will be denoted by $p(b)\pm$.

Let us remark that we always need to know the velocity and the pressure on the boundary to evaluate numerical fluxes over the boundary edge. If we do not want to prescribe it, it is computed using the standard predictor with ghost cell values copied from the inside, as described in more detail in the next paragraph.

A.2 Free boundary condition

A free boundary condition means that no external force is applied, indicating zero pressure. The boundary should move freely, depending on the velocity from the inside of the computational domain. In other words, the velocity and pressure are interpolated from the values of the inner cells. Such boundary condition can be easily implemented using ghost-cells. The vector of conserved variables in the ghost cell $g(i)$ is just a simple copy of the values in cell i:

$$\boldsymbol{w}_{g(i)} = \boldsymbol{w}_i.$$

In this case, the states at the endpoint of the boundary edge $\boldsymbol{w}_{p(b)\pm}$ are computed using the predictor (4.7), or (5.20-5.21). Thermodynamical variables are computed using EOS. Corner ghost cell values are set to the value of the cell, with which they share a single node.

A.3 Prescribed velocity

In the case of prescribed velocity (used for example in Saltzman and Coggeshall tests) we set the velocity in the ghost cell using the following simple extrapolation:

$$\boldsymbol{U}_{g(i)} = 2\boldsymbol{U}_{BC} - \boldsymbol{U}_i, \tag{A.1}$$

and update the specific total energy in ghost cell $g(i)$:

$$E_{g(i)} = \varepsilon_{g(i)} + (\boldsymbol{U}_{BC} \cdot \boldsymbol{U}_{BC})/2. \tag{A.2}$$

Here, we assume that the cell volume and mass of cells i and $g(i)$ are the same, the nodal velocity $\boldsymbol{U}_{p(b)}$ is then computed by the predictor.

A.4 Reflecting boundary

The reflecting boundary condition or no-slip condition, is defined by zero perpendicular component of velocity on the boundary. Besides piston-like problems (e.g. Saltzman), this BC is often used for circularly symmetric tests, such as Noh, Sedov, Kidder, or Coggeshall, where by imposing this condition on x- and y- axes one can compute only one quadrant of the whole problem. Because in this work, this boundary condition is always set on boundaries formed by straight-line, the implementation reduces to a simple combination of free boundary condition and changing sign of perpendicular velocity component, which corresponds to extrapolation (A.1) with $\boldsymbol{U}_{BC} = (u, 0)^t$ for horizontal boundary (x-axis) and $\boldsymbol{U}_{BC} = (0, v)^t$ for vertical boundary (y-axis) respectively.

A.5 Prescribed pressure

Imposing pressure on boundary $\partial\Omega$ is done on the boundary nodes, where one has to do nothing else than updating the pressures on the boundary:

$$P_{p(b)} = P_{BC},$$

while the nodal velocity $\boldsymbol{U}_{p(b)}$ is computed by the predictor (4.7) with ghost cells set to have zero volume and mass, so the integration by predictor is performed only over the two inner subzones.

A.6 Free Surface Boundary

For 2D elastic-perfectly plastic flow, one often needs to prescribe a free surface boundary, expressing the fact that there is no external force acting on the boundary node. The Free Surface boundary condition (also known as Traction Free BC) is given by the following equation [119]:

$$\boldsymbol{\tau} = \mathbb{T} \cdot \boldsymbol{n} = \mathbf{0}, \tag{A.3}$$

where $\boldsymbol{\tau}$ is the traction (force per unit area), $\mathbb{T}$ is Cauchy stress tensor, and $\boldsymbol{n}$ is the unit outward vector normal to the boundary. Let us now study the boundary equation (A.3) in more detail. After breaking it down into components, the previous equation reads:

$$\mathbb{T} \cdot \boldsymbol{n} = \begin{pmatrix} T_{xx} & T_{xy} \\ T_{xy} & T_{yy} \end{pmatrix} \cdot \begin{pmatrix} n_x \\ n_y \end{pmatrix} = \begin{pmatrix} T_{xx} n_x + T_{xy} n_y \\ T_{xy} n_x + T_{yy} n_y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}, \tag{A.4}$$

where we have used the fact that Cauchy tensor is symmetric and hence $T_{xy} = T_{yx}$. This vector equation gives us two constraints on three unique tensor components.

For the sake of simplicity, we will now focus on the special case of West (left) vertical boundary, where $\boldsymbol{n} = (-1, 0)^t$. After substitution into (A.4) we have

$$T_{xx} = 0, \qquad T_{xy} = T_{yx} = 0. \tag{A.5}$$

The remaining component T_{yy} represents shear stress and does not change on the boundary.

For the implementation of this specific case, one has to first compute the Cauchy stress tensor in node $p(b)$ laying on the boundary. The values in ghost cell $g(i)$ are copied from the neighboring cell i, and the predictor (3.27) and (8.16) computes the nodal Cauchy tensor $\mathbb{T}_{p(b)}^{n+\frac{1}{2}}$, which is then updated using the following procedure.

For a special case when the cell edge on the boundary is parallel with the y-axis, the free surface boundary condition is done by applying (A.3) to the Cauchy tensor $\mathbb{T}_{p(b)}^{n+\frac{1}{2}}$:

$$\mathbb{T}_{p(b)}^{n+\frac{1}{2}} = \begin{pmatrix} 0 & 0 \\ 0 & (T_{yy})_{p(b)}^{n+\frac{1}{2}} \end{pmatrix}. \tag{A.6}$$

Any general case can be easily converted to this particular situation using tensor rotation. Let us define rotation tensor

$$\mathbb{R} = \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix}. \tag{A.7}$$

Then Cauchy tensor $\mathbb{T}$ rotated counterclockwise by the angle φ will be.

$$\mathbb{T}^R = \mathbb{R}\mathbb{T}\mathbb{R}^t. \tag{A.8}$$

The general algorithm for Free Surface BC consists of the following steps:

1. Copy the specific vector of conserved variables and Cauchy stress tensor from cell i

to cell $g(i)$ using

$$\boldsymbol{w}_{g(i)} = \boldsymbol{w}_i, \qquad \mathbb{T}_{g(i)} = \mathbb{T}_i,$$

2. compute predicted Cauchy stress tensor $\mathbb{T}_{p(b)}^{n+\frac{1}{2}}$ using the standard predictor (4.7) and (8.16),
3. rotate the tensor using (A.8), so it is aligned with the y-axis,
4. prescribe the components of rotated tensor according to (A.6),
5. rotate the updated tensor back.

Appendix B

Symmetry proofs

B.1 Symmetry preservation for LWrz

For the following symmetry proofs, it will be beneficial to use polar coordinates (R, θ) instead of cylindrical coordinates (r, z). Polar (spherical) radius R is defined as the distance of the point (r, z) from the origin, i.e.

$$R = \sqrt{r^2 + z^2},$$

while the polar angle $\theta = \arctan \frac{r}{z}$ is the angle between the vector (r, z) and the $z-$axis. The polar coordinates are hence connected with the cylindrical ones using transformation

$$(r, z) = R(\sin\theta, \cos\theta). \tag{B.1}$$

Assuming equiangular polar mesh, every node $\boldsymbol{x}_{i+\frac{1}{2},j+\frac{1}{2}}$ can be defined using angle $\theta_{i+\frac{1}{2}}$ and radius $R_{j+\frac{1}{2}}$ so

$$\boldsymbol{x}_{i+\frac{1}{2},j+\frac{1}{2}} = R_{j+\frac{1}{2}}(\sin\theta_{i+\frac{1}{2}}, \cos\theta_{i+\frac{1}{2}}). \tag{B.2}$$

The ray edges on the polar mesh have a constant index i while circles have a constant index j. By the spherical symmetry, we understand the situation when the value of any scalar quantity at point $(r, z) = R(\sin\theta, \cos\theta)$ does not depend on the polar angle θ. Vector quantity is spherically symmetric when its magnitude does not depend on polar angle θ and is spherically radial, which means parallel to the vector (r, z).

Similarly to [67], we will in the following text always assume equiangular meshes, and by symmetry, we will always understand spherical symmetry.

B.2 Equiangular polar mesh

Let us now define equiangular polar mesh and some notations which will be used in the symmetry proofs. We would like to remind here that by A we mean Cartesian area computed by (5.6), while by V we mean true cylindrical volume (5.5). The equiangular polar mesh is illustrated in Figure B.1. A primary cell i, j is a trapezoid given by vertices $\boldsymbol{x}_{i+\frac{1}{2},j-\frac{1}{2}}, \boldsymbol{x}_{i-\frac{1}{2},j-\frac{1}{2}}, \boldsymbol{x}_{i-\frac{1}{2},j+\frac{1}{2}}, \boldsymbol{x}_{i+\frac{1}{2},j+\frac{1}{2}}$. Its area $A_{i,j}$ can be computed as a difference of areas of the triangle with vertices $\boldsymbol{0}, \boldsymbol{x}_{i-\frac{1}{2},j+\frac{1}{2}}, \boldsymbol{x}_{i+\frac{1}{2},j+\frac{1}{2}}$ and the triangle with vertices

$\mathbf{0}, \boldsymbol{x}_{i-\frac{1}{2},j-\frac{1}{2}}, \boldsymbol{x}_{i+\frac{1}{2},j-\frac{1}{2}}$:

$$A_{i,j} = A_{\triangle(\mathbf{0},\boldsymbol{x}_{i-\frac{1}{2},j+\frac{1}{2}},\boldsymbol{x}_{i+\frac{1}{2},j+\frac{1}{2}})} - A_{\triangle(\mathbf{0},\boldsymbol{x}_{i-\frac{1}{2},j-\frac{1}{2}},\boldsymbol{x}_{i+\frac{1}{2},j-\frac{1}{2}})}.$$

A similar formula is also valid for the volume $V_{i,j}$. Let us find the volume of the polar triangle $\mathcal{T} = \triangle(\mathbf{0}, \boldsymbol{x}_{i-\frac{1}{2},j+\frac{1}{2}}, \boldsymbol{x}_{i+\frac{1}{2},j+\frac{1}{2}})$. Its area $A_{\mathcal{T}}$ for given $j+\frac{1}{2}$ is clearly independent on the angle θ_i and its volumes $V_{\mathcal{T}}$ is

$$V_{\mathcal{T}} = \frac{1}{3}(r_{i-\frac{1}{2},j+\frac{1}{2}} + r_{i+\frac{1}{2},j+\frac{1}{2}})A_{\mathcal{T}} = \frac{2}{3}R_{j+\frac{1}{2}} \sin\theta_i \cos\Delta\theta/2.$$

Therefore, assuming symmetric mesh, the primary cell volume depends on the angle as $\sin\theta_i$, and also primary mass cell depends on the angle as $\sin\theta_i$, because the density is symmetric. Dual cell $i-\frac{1}{2}, j+\frac{1}{2}$ is a hexagon with vertices $\boldsymbol{x}_{i-1,j}, \boldsymbol{x}_{i-1,j+1}, \boldsymbol{x}_{i-\frac{1}{2},j+1}, \boldsymbol{x}_{i,j+1}, \boldsymbol{x}_{i,j}, \boldsymbol{x}_{i-\frac{1}{2},j}$ and consists of four quadrilateral subzones with a common vertex $\boldsymbol{x}_{i-\frac{1}{2},j+\frac{1}{2}}$. Its volume is given by the sum of volumes of these subzones. If we look closer on the volume of two subcells with the common ray edge $i-\frac{1}{2}, j+\frac{3}{4}$, using similar arguments as for primary cell, we find that the volumes depend on the angle as $\sin\theta_{i\pm\frac{1}{4}}$ times the same value independent on the angle, thus the volume of their union depends on the angle as $\sin\theta_{i-\frac{1}{2}}$. The same dependency clearly holds for the rest two subzones sharing the common ray edge $i-\frac{1}{2}, j+\frac{1}{4}$ and hence the volume and the mass of the dual cell depend on the angle as $\sin\theta_{i-\frac{1}{2}}$.

B.3 Proof of symmetry of velocity predictor

In this section, we will prove that if quantities in time t^n are spherically symmetric, the nodal velocity at $t^{n+\frac{1}{2}}$ due to (5.20) is also spherically symmetric.

Let us first look into the weighted average given by the first term of the right-hand side of Eq. (5.20). Since the cell velocities are assumed to be symmetric, subcell areas do not depend on the angle θ, and the polar mesh is symmetric, this term is clearly also symmetric. To show the symmetry of the second term of the right-hand side of (5.20), we will first assume two circle separators (dual cell edges) with radial index $j+1$. From the assumption that the underlying pressures are symmetric, the fact that on the symmetric mesh the separators laying on the same circle have the same length, and that their normals are radial, also their sum has to be symmetric. A similar argument also holds for two circle separators with radial index j.

It remains to prove the symmetry of radial separators. Let us consider two rays separators with radial index $j+\frac{3}{4}$. The situation is similar to the circle separators – the pressures on the same circle are the same, and the length of separators is the same, hence the sum of the fluxes through them is symmetric. This is also true for the rest ray separators, and hence the whole term is symmetric.

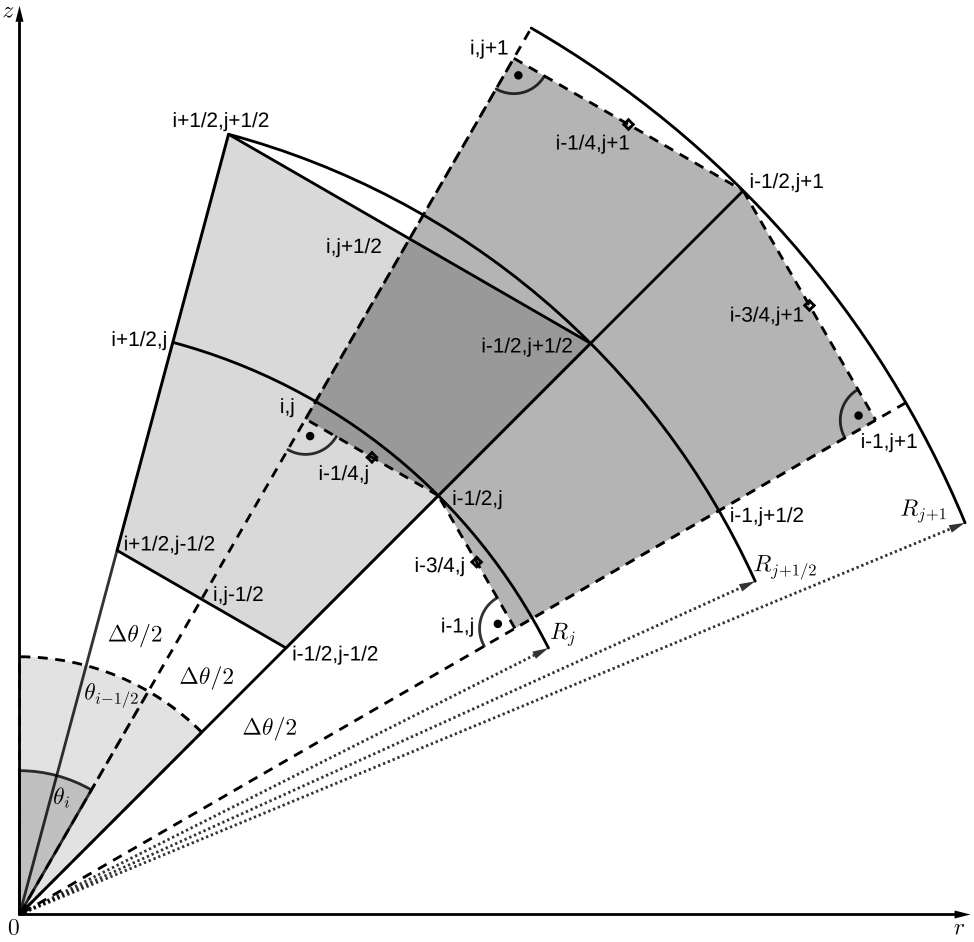


Figure B.1: Illustration of primary cell i, j, and dual cell $i - \frac{1}{2}, j + \frac{1}{2}$ on the equiangular polar grid.

B.4 Proof of symmetry of predictor for density, energy, and pressure

In Section B.2 we have shown that both volume and mass depend on the angle as $\sin\theta_{i-\frac{1}{2}}$. Since the nodal estimate of density is computed as the fraction of mass and velocity, it does not depend on the angle θ and hence is symmetric.

Let us now discuss the predicted total energy given by (5.21). The first term on the right-hand side is symmetric due to the same arguments as in the previous Section B.3.

Now we can advance to the second term of RHS of (5.21), which contains separator (dual cell edge) fluxes $P_c r_{spc} \boldsymbol{U}_c \cdot \boldsymbol{n}_{spc}$. On the ray edges, these fluxes are zero because the radial velocity $\boldsymbol{U}_c$ is perpendicular to $\boldsymbol{n}_{spc}$. The circle fluxes are parallel, and therefore $\boldsymbol{U}_c \cdot \boldsymbol{n}_{spc} = |\boldsymbol{U}_c||\boldsymbol{n}_{spc}| = U_c n_{spc}$, so these fluxes do not depend on the angle θ and can be factored out. The only angular dependence of the flux at the separator $s_{i-\frac{1}{4},j}$ is then

$$r_{i-\frac{1}{4},j+1} + r_{i-\frac{3}{4},j+1} = 2R_{j+\frac{1}{2}} \sin\theta_{i-\frac{1}{2}} \cos(\Delta\theta/4).$$

The entire sum in the second term hence depends on the angle as $\sin\theta_{i-\frac{1}{2}}$, which is the same dependence as for nodal mass m_p, as shown in Section B.2. Since the second term is a fraction of the sum and the nodal mass, the angular dependence cancels, and the whole term is symmetric, therefore the entire energy predictor is symmetric.

As the pressure computed using EOS (2.55) depends only on quantities proved already to be symmetric, it is also symmetric.

B.5 Proof of symmetry of velocity corrector

To prove symmetry for corrector (5.23), we will assume that all quantities at times t^n and $t^{n+\frac{1}{2}}$ are symmetric. The situation is again illustrated in Fig. B.1. Let us start with the ray edges. Due to the symmetry $P_e^{n+\frac{1}{2}} = \bar{P}_c^{n+\frac{1}{2}}$ and hence the ray fluxes are zero. For a circle edge of a primary cell, the pressure difference $P_e^{n+\frac{1}{2}} - \bar{P}_c^{n+\frac{1}{2}}$ does not depend on the angle and the normal $\boldsymbol{n}_e^{n+\frac{1}{2}}$ is radial, and its magnitude is symmetric, therefore the only dependence on the angle is due to the radius r_e, which is for the edge $i, j+\frac{1}{2}$:

$$r_{i,j+\frac{1}{2}} = R_{j+\frac{1}{2}} \sin\theta_i.$$

As shown in Section B.2, the primary cell mass also depends as $\sin\theta_i$. Since the entire last term of (5.23) is the ratio of the sum and the mass, the dependence on the angle θ is canceled out, and hence the whole corrector preserves spherical symmetry.

B.6 Proof of symmetry of corrector for density, energy, and pressure

The density of the primary cell is again given by the ratio of cell mass and cell volume, which, as shown in Sec. B.2, both depend on the angle as $\sin\theta_i$, and hence the density is symmetric.

The corrector for the energy is given by (5.27). At the ray edges of a primary cell, the velocity is perpendicular to the edge normals and hence the flux containing $\boldsymbol{U}_e \cdot \boldsymbol{n}_e$ is zero. At the circle edges, the velocity and edge normals are parallel, so $P_e \boldsymbol{U}_e \cdot \boldsymbol{n}_e = P_e |\boldsymbol{U}_e||\boldsymbol{n}_e|$, and hence the only dependence of the face flux is due to the radius r_e. Similarly to the previous proof in Sec. B.5, the dependence of r_e is the same as the dependence of primary cell mass, so their ratio does not depend on the polar angle, and the corrector (5.27) does not violate the symmetry.

B.7 Symmetry preservation of artificial viscosity term and its correction

The artificial viscous fluxes for the cell c are given by:

$$\sum_{e(c)} r_{e(c)} |\boldsymbol{n}_{e(ca)}| \hat{\sigma}_{e(ca)} (\boldsymbol{U}_a - \boldsymbol{U}_c), \tag{B.3}$$

where the summation $e(c)$ goes over all edges of cell c, and the edge $e(ca)$ is the common edge adjacent to cells c and a. Let us assume the symmetric case depicted in Fig. B.1. We will now investigate the cell $c = i, j$. The edges lying on logical circles, i.e. the edges having the same j-coordinate, are spherically radial, which means that $\boldsymbol{U}_c || \boldsymbol{U}_a$ and therefore $(\boldsymbol{U}_c - \boldsymbol{U}_a) || \boldsymbol{U}_c$. The edge radius $r_{e(ca)} = r_{i,j-\frac{1}{2}}$ and cell mass m_c depend on the angle as $\sin(\theta_i)$, and hence their ratio is independent on the angle θ. All that means that the viscous flux (B.3) at the circle edges is symmetric.

For the ray edges (the edges having the same i-coordinate), the term $|\boldsymbol{n}_{e(ca)} \hat{\sigma}_{e(ca)}|$ is independent on the angle θ. The symmetry of the viscous flux then depends on vector quantity $\boldsymbol{f}_j^{AV}$ given by

$$\boldsymbol{f}_j^{AV} = r_{i-\frac{1}{2},j+1} (\boldsymbol{U}_{i-1,j} - \boldsymbol{U}_{i,j}) + r_{i+\frac{1}{2},j} (\boldsymbol{U}_{i+1,j} - \boldsymbol{U}_{i,j}). \tag{B.4}$$

After some trigonometry manipulations the z-component of AV flux $f_j^{AV,z}$ can be rewritten:

$$f_j^{AV,z} = -8 R_j U_j \sin^2(\Delta\theta/2) \cos(\Delta\theta/2) \sin(\theta_i) \cos(\sigma_i), \tag{B.5}$$

where R_j is the distance from the origin, and U_j is a signed magnitude of velocity.

The factor of $\sin(\sigma_i)$ cancels with the same factor in cell mass $m_{i,j}$ and the factor $\cos(\sigma_i)$ from $f_j^{AV,z}$ is nothing else than the projection of a vector of length

$$8 R_j U_j \sin^2(\Delta\theta/2) \cos(\Delta\theta/2)$$

on the z-axis, so the z-component of (B.4) does not violate the symmetry.

The r-component of (B.4) is

$$f_j^{AV,r} = -4 R_j U_j \sin^2(\Delta\theta/2) \cos(\Delta\theta/2) (2 \sin^2(\theta_i) - 1), \tag{B.6}$$

which violates symmetry. Deriving the correction δ is however straightforward with the following result:

$$\delta = 4R_jU_j\sin^2(\Delta\theta/2)\cos(\Delta\theta/2), \tag{B.7}$$

which can be easily verified:

$$f_j^{AV,r} - \delta = -8R_jU_j\sin^2(\Delta\theta/2)\cos(\Delta\theta/2)\sin^2(\theta_i). \tag{B.8}$$

Here, one factor $\sin(\theta_i)$ cancels with the same one contained in cell mass $m_{i,j}$, and the second one is a projection of vector of a length $8R_jU_j\sin^2(\Delta\theta/2)\cos(\Delta\theta/2)$ on the r-axis.

The distance of two cell centers is:

$$|\boldsymbol{x}_{i+1} - \boldsymbol{x}_i| = 2R_j\sin(\Delta\theta/2)\cos(\Delta\theta/2), \tag{B.9}$$

which can be used for rewriting the symmetry correction (B.7) as:

$$\delta = 4R_jU_j\sin^2(\Delta\theta/2)\cos(\Delta\theta/2) = |\boldsymbol{x}_{i+1} - \boldsymbol{x}_i|^2\frac{u_{i,j}}{r_{i,j}}, \tag{B.10}$$

where $u_{i,j}$ is r-component of velocity $\boldsymbol{U}_{i,j} = (u_{i,j}, v_{i,j})$ and $r_{i,j}$ is the radius of the cell center. The final formula for the correction of the viscous flux (B.3) through the ray edges of cell c writes:

$$\delta = \frac{B_{i-\frac{1}{2},j} + B_{i+\frac{1}{2},j}}{2}\frac{u_{i,j}}{r_{i,j}}, \quad B_{i-\frac{1}{2},j} = \hat{\sigma}_{i-\frac{1}{2},j}|\boldsymbol{n}_{i-\frac{1}{2},j}||\boldsymbol{x}_{i,j} - \boldsymbol{x}_{i-1,j}|^2. \tag{B.11}$$

Going back to the pc-notation, the formula (B.11) can be rewritten as (5.43).

B.8 Proof of symmetry of energy dissipation

Now we will shortly discuss the symmetry of energy dissipation given by (5.42b). Due to the symmetry, $E_a = E_c$ on the ray edge $e(ca)$ and therefore the artificial energy fluxes at the ray edges are zero. At the circle edges, only the radius $r^n_{e(ca)}$ depends on the angle as $\sin(\theta_i)$, which cancels with the cell mass that also depends on the angle as $\sin(\theta_i)$, therefore the energy dissipation fluxes preserve the symmetry.

Bibliography

[1] M. L. Wilkins, *Computer simulation of dynamic phenomena*, ser. Scientific Computation. Springer, 2013, p. 247, ISBN: 3662038854.

[2] J. von Neumann and R. D. Richtmyer, "A method for the numerical calculations of hydrodynamical shocks", *J. Appl. Phys.*, vol. 21, no. 3, pp. 232–238, 1950, ISSN: 0021-8979. DOI: `10.1063/1.1699639`.

[3] M. L. Wilkins, "Calculations of elastic-plastic flow", in *Methods Comput. Phys.*, B. J. Alder, S Fernbach, and M Rotenberg, Eds., vol. III, 1964, pp. 211–263.

[4] D. J. Benson, "Computational methods in Lagrangian and Eulerian hydrocodes", *Comput. Methods Appl. Mech. Eng.*, vol. 99, no. 2-3, pp. 235–394, 1992, ISSN: 00457825. DOI: `10.1016/0045-7825(92)90042-I`.

[5] E. Caramana and M. Shashkov, "Elimination of Artificial Grid Distortion and Hourglass-Type Motions by Means of Lagrangian Subzonal Masses and Pressures", *J. Comput. Phys.*, vol. 142, no. 2, pp. 521–561, 1998, ISSN: 0021-9991. DOI: `10.1006/JCPH.1998.5952`.

[6] D. Burton, "Multidimensional discretization of conservation laws for unstructured polyhedral grids", *SAMGOP-94 2nd Int. Work. Anal. methods Process Optim. fluid gas Mech. Arzamas*, 1994.

[7] J. Campbell and M. Shashkov, "A Compatible Lagrangian Hydrodynamics Algorithm on Unstructured Grids", Los Alamos National Laboratory, Tech. Rep. LA-UR-00-3231, 2000.

[8] E. Caramana, M. Shashkov, and P. Whalen, "Formulations of Artificial Viscosity for Multi-dimensional Shock Wave Computations", *J. Comput. Phys.*, vol. 144, no. 1, pp. 70–97, 1998, ISSN: 0021-9991. DOI: `10.1006/JCPH.1998.5989`.

[9] J. Campbell and M. Shashkov, "A Tensor Artificial Viscosity Using a Mimetic Finite Difference Algorithm", *J. Comput. Phys.*, vol. 172, no. 2, pp. 739–765, 2001, ISSN: 0021-9991. DOI: `10.1006/jcph.2001.6856`.

[10] K. Lipnikov and M. Shashkov, "A Framework for Developing a Mimetic Tensor Artificial Viscosity for Lagrangian Hydrocodes on Arbitrary Polygonal Meshes", *J. Comput. Phys.*, vol. 229, no. 20, pp. 7911–7941, 2010.

[11] A. Burbeau-Augoula, "A Node-centered Artificial Viscosity Method for Two-dimensional Lagrangian Hydrodynamics Calculations on a Staggered Grid", *Commun. Comput. Phys.*, vol. 8, no. 4, pp. 877–900, 2010, ISSN: 1815-2406. DOI: `10.4208/cicp.030709.161209a`.

[12] R. Loubère, P. H. Maire, and P. Váchal, "A second-order compatible staggered Lagrangian hydrodynamics scheme using a cell-centered multidimensional approximate Riemann solver", in *Procedia Comput. Sci.*, vol. 1, Elsevier, 2010, pp. 1931–1939. DOI: `10.1016/j.procs.2010.04.216`.

[13] P. H. Maire, R. Loubère, and P. Váchal, "Staggered Lagrangian discretization based on cell-centered Riemann solver and associated hydrodynamics scheme", *Commun. Comput. Phys.*, vol. 10, no. 4, pp. 940–978, 2011, ISSN: 19917120. DOI: `10.4208/cicp.170310.251110a`.

[14] R. Loubère, P.-H. Maire, and P. Váchal, "3D staggered Lagrangian hydrodynamics scheme with cell-centered Riemann solver-based artificial viscosity", *Int. J. Numer. Methods Fluids*, vol. 72, no. 1, pp. 22–42, 2013, ISSN: 02712091. DOI: `10.1002/fld.3730`.

[15] N. N. R. Morgan, K. N. Lipnikov, D. E. Burton, and M. A. Kenamond, "A Lagrangian staggered grid Godunov-like approach for hydrodynamics", *J. Comput. Phys.*, vol. 259, pp. 568–597, 2014, ISSN: 10902716. DOI: `10.1016/j.jcp.2013.12.013`.

[16] P. Váchal and B. Wendroff, "A Symmetry Preserving Dissipative Artificial Viscosity in r-z Staggered Lagrangian Discretization", *Journal of Computational Physics*, vol. 258, pp. 118–136, 2014. DOI: `10.1016/j.jcp.2013.10.036`.

[17] P. Váchal and B. Wendroff, "A symmetry preserving dissipative artificial viscosity in r-z geometry", *Int. J. Numer. Methods Fluids*, vol. 76, no. 3, pp. 185–198, 2014, ISSN: 10970363. DOI: `10.1002/fld.3926`.

[18] P. Váchal and B. Wendroff, "On preservation of symmetry in r–z staggered Lagrangian schemes", *J. Comput. Phys.*, vol. 307, no. 18, pp. 496–507, 2016, ISSN: 00219991. DOI: `10.1016/J.JCP.2015.11.063`.

[19] S. K. Godunov, "A difference scheme for numerical solution of discontinuous solution of hydrodynamic equations", *Math. Sb.*, vol. 47, pp. 271–306, 1959.

[20] S. K. Godunov, "Reminiscences about difference schemes", *J. Comput. Phys.*, vol. 153, no. 1, pp. 6–25, 1999.

[21] 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*, Unknow, May 1992.

[22] J. K. Dukowicz and B. J. Meltz, "Vorticity errors in multidimensional Lagrangian codes", *J. Comput. Phys.*, vol. 99, no. 1, pp. 115–134, 1992, ISSN: 10902716. DOI: `10.1016/0021-9991(92)90280-C`.

[23] B. Després and C. Mazeran, "Lagrangian Gas Dynamics in Two Dimensions and Lagrangian systems", *Arch. Ration. Mech. Anal.*, vol. 178, no. 3, pp. 327–372, 2005, ISSN: 0003-9527. DOI: `10.1007/s00205-005-0375-4`.

[24] G. Carre, S. Delpino, B. Despres, and E. Labourasse, "A cell-centered Lagrangian hydrodynamics scheme in arbitrary dimension", *J. Comput. Phys.*, vol. 228, no. 14, pp. 5160–5183, 2009.

[25] B. Rebourcet, "Comments on the filtering of numerical instabilities in Lagrangian hydrocodes", in *Conf. Numer. methods multi-material fluid flows*, 2007, pp. 10–14.

[26] P.-H. Maire, R Abgrall, J Breil, and J Ovadia, "A Cell-Centered Lagrangian Scheme for Two-Dimensional Compressible Flow Problems", *SIAM J. Sci. Comput.*, vol. 29, no. 4, pp. 1781–1824, 2007.

[27] P.-H. Maire and J Breil, "A second order cell-centered Lagrangian scheme for two-dimensional compressible flow problems", *Int. J. Numer. Methods Fluids*, vol. 56, pp. 1417–1423, 2008.

[28] M. Ben-Artzi and J. Falcovitz, *Generalized Riemann Problems in Computational Fluids Dynamics*. Cambridge University Press, 2003.

[29] P.-H. Maire, "A high-order cell-centered Lagrangian scheme for compressible fluid flows in two-dimensional cylindrical geometry", *J. Comput. Phys.*, vol. 228, no. 18, pp. 6882–6915, 2009, ISSN: 0021-9991. DOI: `10.1016/J.JCP.2009.06.018`.

[30] P.-H. Maire, "A high-order cell-centered Lagrangian scheme for two-dimensional compressible fluid flows on unstructured meshes", *J. Comput. Phys.*, vol. 228, no. 7, pp. 2391–2425, 2009.

[31] P.-H. Maire and B. Nkonga, "Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics", *J. Comput. Phys.*, vol. 228, no. 3, pp. 799–821, 2009.

[32] P.-H. Maire, R. Abgrall, J. Breil, R. Loubère, and B. Rebourcet, "A nominally second-order cell-centered Lagrangian scheme for simulating elastic–plastic flows on two-dimensional unstructured grids", *J. Comput. Phys.*, vol. 235, pp. 626–665, 2013, ISSN: 0021-9991. DOI: `10.1016/J.JCP.2012.10.017`.

[33] S. K. Sambasivan, M. J. Shashkov, and D. E. Burton, "A cell-centered Lagrangian finite volume approach for computing elasto-plastic response of solids in cylindrical axisymmetric geometries", *J. Comput. Phys.*, vol. 237, pp. 251–288, 2013, ISSN: 0021-9991. DOI: `10.1016/j.jcp.2012.11.044`.

[34] G. Georges, J. Breil, and P. H. Maire, "A 3D finite volume scheme for solving the updated Lagrangian form of hyperelasticity", *Int. J. Numer. Methods Fluids*, vol. 84, no. 1, pp. 41–54, 2017, ISSN: 10970363. DOI: `10.1002/fld.4336`.

[35] P.-H. Maire, "A unified sub-cell force-based discretization for cell-centered Lagrangian hydrodynamics on polygonal grids", *Int. J. Numer. Methods Fluids*, vol. 65, no. 11-12, pp. 1281–1294, 2011, ISSN: 02712091. DOI: `10.1002/fld.2328`.

[36] D. E. Burton, T. C. Carney, N. R. Morgan, S. K. Sambasivan, and M. J. Shashkov, "A cell-centered Lagrangian Godunov-like method for solid dynamics", *Comput. Fluids*, vol. 83, pp. 33–47, 2013, ISSN: 0045-7930. DOI: `10.1016/J.COMPFLUID.2012.09.008`.

[37] Y. Liu, W. Shen, B. Tian, and D. K. Mao, "Maire's new cell-centered Lagrangian method based on arbitrary Riemann solver", in *Proc. - 2014 Int. Conf. Comput. Sci. Comput. Intell. CSCI 2014*, vol. 1, IEEE Computer Society, 2014, pp. 374–379, ISBN: 9781479930098. DOI: `10.1109/CSCI.2014.158`.

[38] A. Barlow and P. Roe, "A cell centered Lagrangian Godunov scheme for shock hydrodynamics", *Comput. Fluids*, vol. 46, pp. 133–136, 2011.

[39] S. K. Sambasivan, M. J. Shashkov, D. E. Burton, and M. A. Christon, "Mimetic Theory for Cell-Centered Lagrangian Finite Volume Formulation on General Unstructured Grids, LA-UR-12-23177", Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), Tech. Rep., 2012. DOI: `10.2172/1047081`.

[40] D. E. Burton, N. R. Morgan, T. C. Carney, and M. A. Kenamond, “Reduction of dissipation in Lagrange cell-centered hydrodynamics (CCH) through corner gradient reconstruction (CGR)”, *J. Comput. Phys.*, vol. 299, pp. 229–280, 2015, ISSN: 0021-9991. DOI: `10.1016/j.jcp.2015.06.041`.

[41] F. Vilar, C. W. Shu, and P. H. Maire, “Positivity-preserving cell-centered Lagrangian schemes for multi-material compressible flows: From first-order to high-orders. Part I: The one-dimensional case”, *J. Comput. Phys.*, vol. 312, pp. 385–415, 2016, ISSN: 10902716. DOI: `10.1016/j.jcp.2016.02.027`.

[42] F. Vilar, C. W. Shu, and P. H. Maire, “Positivity-preserving cell-centered Lagrangian schemes for multi-material compressible flows: From first-order to high-orders. Part II: The two-dimensional case”, *J. Comput. Phys.*, vol. 312, pp. 416–442, 2016, ISSN: 10902716. DOI: `10.1016/j.jcp.2016.01.037`.

[43] W. Boscheri, M. Dumbser, R. Loubère, and P.-H. Maire, “A second-order cell-centered Lagrangian ADER-MOOD finite volume scheme on multidimensional unstructured meshes for hydrodynamics”, *J. Comput. Phys.*, vol. 358, pp. 103–129, 2018, ISSN: 0021-9991. DOI: `10.1016/J.JCP.2017.12.040`.

[44] F. Vilar, P.-H. Maire, and R. Abgrall, “A discontinuous Galerkin discretization for solving the two-dimensional gas dynamics equations written under total Lagrangian formulation on general unstructured grids”, *J. Comput. Phys.*, vol. 276, pp. 188–234, 2014, ISSN: 0021-9991. DOI: `10.1016/j.jcp.2014.07.030`.

[45] W. B. Goad, “WAT: A numerical method for two-dimensional unsteady fluid flow”, Los Alamos National Lab NM, Tech. Rep., 1960.

[46] P Lascaux, “Application de la méthode des éléments finis en hydrodynamique bidimensionnelle utilisant les variables de Lagrange”, *Rapp. Tech. CEA Limeil*, 1972.

[47] P. Lascaux, “Application of the finite element method to 2D Lagrangian hydrodynamics”, *Proc. Int. Symp. Finite Elem. Methods Flow Probl. Swansea, Wales, Jan. 7–11, 1974*, pp. 139–152, 1974.

[48] A. J. Barlow, P.-H. Maire, W. J. Rider, R. N. Rieben, and M. J. Shashkov, “Arbitrary Lagrangian–Eulerian methods for modeling high-speed compressible multimaterial flows”, *J. Comput. Phys.*, vol. 322, pp. 603–665, 2016, ISSN: 0021-9991. DOI: `10.1016/J.JCP.2016.07.001`.

[49] G. Scovazzi, M. A. Christon, T. J. Hughes, and J. N. Shadid, “Stabilized shock hydrodynamics: I. A Lagrangian method”, *Comput. Methods Appl. Mech. Eng.*, vol. 196, no. 4-6, pp. 923–966, 2007, ISSN: 00457825. DOI: `10.1016/j.cma.2006.08.008`.

[50] G. Scovazzi, E. Love, and M. J. Shashkov, “Multi-scale Lagrangian shock hydrodynamics on Q1/P0 finite elements: Theoretical framework and two-dimensional computations”, *Comput. Methods Appl. Mech. Eng.*, vol. 197, no. 9-12, pp. 1056–1079, 2008, ISSN: 00457825. DOI: `10.1016/j.cma.2007.10.002`.

[51] J. Cheng and C. W. Shu, “A third order conservative Lagrangian type scheme on curvilinear meshes for the compressible Euler equations”, *Commun. Comput. Phys.*, vol. 4, no. 5, pp. 1008–1024, 2008, ISSN: 18152406.

[52] F. Vilar, "Cell-centered discontinuous Galerkin discretization for two-dimensional Lagrangian hydrodynamics", *Comput. Fluids*, vol. 64, pp. 64–73, 2012, ISSN: 00457930. DOI: `10.1016/j.compfluid.2012.05.001`.

[53] V. A. Dobrev, T. E. Ellis, T. V. Kolev, and R. N. Rieben, "High-order curvilinear finite elements for axisymmetric Lagrangian hydrodynamics", *Comput. Fluids*, vol. 83, pp. 58–69, 2013, ISSN: 00457930. DOI: `10.1016/j.compfluid.2012.06.004`.

[54] P. D. Lax and B Wendroff, "Systems of conservation laws", *Comm. Pure Appl. Math.*, vol. 13, pp. 217–237, 1960.

[55] P. D. Lax and B Wendroff, "Difference schemes for hyperbolic equations with high order of accuracy", *Comm. Pure Appl. Math*, vol. 17, pp. 381–398, 1964.

[56] R. D. Richtmyer, "A survey of difference methods for non-steady fluid dynamics", National Center for Atmospheric Research, Tech. Rep. NCAR Technical Notes 63-2, 1963, p. 32. DOI: `10.5065/D67P8WCQ`.

[57] J. C. Wilson, "Stability of Richtmyer Type Difference Schemes in any Finite Number of Space Variables and Their Comparison with Multistep Strang Schemes", *IMA J. Appl. Math.*, vol. 10, no. 2, pp. 238–257, 1972, ISSN: 0272-4960. DOI: `10.1093/imamat/10.2.238`.

[58] R. J. LeVeque, *Finite Volume Methods for Hyperbolic Problems*. Cambridge: Cambridge University Press, 2002, p. 558, ISBN: 9780511791253. DOI: `10.1017/CBO9780511791253`.

[59] M. Shashkov and B. Wendroff, "A Composite Scheme for Gas Dynamics in Lagrangian Coordinates", *J. Comput. Phys.*, vol. 150, no. 2, pp. 502–517, 1999, ISSN: 0021-9991. DOI: `10.1006/JCPH.1999.6192`.

[60] P. Bureš and R. Liska, "Hybrid schemes for Euler equations in Lagrangian coordinates", in *Algoritm. 2005*, A Handlovičová, Z Krivá, K Mikula, and D Ševčovič, Eds., Slovak University of Technology, Bratislava, 2005, pp. 83–92.

[61] R. Liska, M. Shashkov, and B. Wendroff, "The early influence of peter lax on computational hydrodynamics and an application of lax-friedrichs and lax-wendroff on triangular grids in Lagrangian coordinates", *Acta Math. Sci.*, vol. 31, no. 6, pp. 2195–2202, 2011, ISSN: 0252-9602. DOI: `10.1016/S0252-9602(11)60393-7`.

[62] T. Lung, "Toward a Simple, Accurate Lagrangian Hydrocode", PhD thesis, University of Michigan, 2015.

[63] A. Harten, P. D. Lax, B van Leer, and B. van Leer, "On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws", *SIAM Rev.*, vol. 25, no. 1, pp. 35–61, 1983, ISSN: 0036-1445. DOI: `10.1137/1025002`.

[64] D. Fridrich, R. Liska, and B. Wendroff, "Some cell-centered Lagrangian Lax–Wendroff HLL hybrid schemes", *J. Comput. Phys.*, vol. 326, pp. 878–892, 2016, ISSN: 00219991. DOI: `10.1016/j.jcp.2016.09.022`.

[65] D. Fridrich, R. Liska, and B. Wendroff, "Cell-centered Lagrangian Lax–Wendroff HLL hybrid method for elasto-plastic flows", *Comput. Fluids*, vol. 157, pp. 164–174, 2017. DOI: `10.1016/j.compfluid.2017.08.030`.

[66] D. Fridrich, R. Liska, and B. Wendroff, “Cell-centred Lagrangian Lax–Wendroff HLL hybrid schemes in cylindrical geometry”, in *Springer Proc. Math. Stat.*, vol. 236, Springer, Cham, 2018, pp. 565–576, ISBN: 9783319915449. DOI: `10.1007/978-3-319-91545-6_43`.

[67] D. Fridrich, R. Liska, and B. Wendroff, “Cell-centered Lagrangian Lax-Wendroff HLL Hybrid Scheme in Cylindrical Geometry”, *J. Comput. Phys.*, vol. 417, p. 109 605, 2020. DOI: `10.1016/j.jcp.2020.109605`.

[68] D. Fridrich, R. Liska, I. Tarant, P. Váchal, and B. Wendroff, “Cell-centered Lagrangian Lax-Wendroff HLL Hybrid Scheme on unstructured meshes”, *Acta Polytech.*, submitted.

[69] J. Serrin, “Mathematical Principles of Classical Fluid Mechanics”, in *Fluid Dynamics I/Strömungsmechanik I*, Springer, 1959, pp. 125–263. DOI: `10.1007/978-3-642-45914-6_2`.

[70] A. J. Chorin and J. E. Marsden, *A mathematical introduction to fluid mechanics.* Springer, 1990, ISBN: 0387904069.

[71] E. N. Dvorkin and M. B. Goldschmit, *Nonlinear Continua.* Springer-Verlag, 2005, p. 266, ISBN: 3-540-24985-0. DOI: `10.1007/3-540-29264-0`.

[72] G. Scovazzi and T. Hughes, “Lecture Notes on Continuum Mechanics on Arbitrary Moving Domains, 2007-6312P”, Sandia National Laboratories, Tech. Rep., 2007.

[73] M. E. Gurtin, E. Fried, and L. Anand, *The Mechanics and Thermodynamics of Continua.* Cambridge: Cambridge University Press, 2010, ISBN: 9780511762956. DOI: `10.1017/CBO9780511762956`.

[74] R. Courant, K. Friedrichs, and H. Lewy, “Über die partiellen Differenzengleichungen der mathematischen Physik”, *Math. Ann.*, vol. 100, no. 1, pp. 32–74, 1928, ISSN: 00255831. DOI: `10.1007/BF01448839`.

[75] F. Vilar, P. H. Maire, and R. Abgrall, “Cell-centered discontinuous Galerkin discretizations for two-dimensional scalar conservation laws on unstructured grids and for one-dimensional Lagrangian hydrodynamics”, *Comput. Fluids*, vol. 46, no. 1, pp. 498–504, 2011, ISSN: 00457930. DOI: `10.1016/j.compfluid.2010.07.018`.

[76] B. Wendroff and A. B. White, “A supraconvergent scheme for nonlinear hyperbolic systems”, *Comput. Math. with Appl.*, 1989, ISSN: 08981221. DOI: `10.1016/0898-1221(89)90232-0`.

[77] J. P. Boris and D. L. Book, “Flux-corrected transport. I. SHASTA, a fluid transport algorithm that works”, *J. Comput. Phys.*, vol. 11, no. 1, pp. 38–69, 1973, ISSN: 10902716. DOI: `10.1016/0021-9991(73)90147-2`.

[78] R. Loubère, “On the effect of the different limiters for the tensor artificial viscosity for the compatible Lagrangian hydrodynamics scheme, LA-UR-05-9301”, Los Alamos National Laboratory, Tech. Rep., 2006.

[79] V. F. Kuropatenko, “Difference methods for hydrodynamics equations”, in *Differ. methods Solut. Probl. Math. physics. Part 1*, Moscow: Nauka, 1966, pp. 107–137.

[80] M. L. Wilkins, “Use of artificial viscosity in multidimensional fluid dynamic calculations”, *J. Comput. Phys.*, vol. 36, no. 3, pp. 281–303, 1980, ISSN: 0021-9991. DOI: `10.1016/0021-9991(80)90161-8`.

[81] E. F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics.* Berlin, Heidelberg: Springer, 2009, ISBN: 978-3-540-25202-3. DOI: `10.1007/b79761`.

[82] S. F. Davis, "Simplified Second-Order Godunov-Type Methods", *SIAM J. Sci. Stat. Comput.*, vol. 9, no. 3, pp. 445–473, 1988, ISSN: 0196-5204. DOI: `10.1137/0909030`.

[83] W. F. Noh, "Errors for calculations of strong shocks using an artificial viscosity and artificial heat flux", *J. Comput. Phys.*, vol. 72, no. 1, pp. 78–120, 1987, ISSN: 10902716. DOI: `10.1016/0021-9991(87)90074-X`.

[84] G. A. Sod, "A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws", *J. Comput. Phys.*, vol. 27, no. 1, pp. 1–31, 1978, ISSN: 00219991. DOI: `10.1016/0021-9991(78)90023-2`.

[85] P. Woodward and P. Colella, "The numerical simulation of two-dimensional fluid flow with strong shocks", *J. Comput. Phys.*, vol. 54, no. 1, pp. 115–173, 1984, ISSN: 0021-9991. DOI: `10.1016/0021-9991(84)90142-6`.

[86] R. Loubère and M. J. Shashkov, "A subcell remapping method on staggered polygonal grids for arbitrary-Lagrangian–Eulerian methods", *J. Comput. Phys.*, vol. 209, no. 1, pp. 105–138, 2005, ISSN: 0021-9991. DOI: `10.1016/J.JCP.2005.03.019`.

[87] L. I. Sedov, *Similarity and dimensional methods in mechanics.* Academic Press, 1959, p. 363, ISBN: 9781483200880.

[88] J. R. Kamm and F. X. Timmes, "On efficient generation of numerically robust Sedov solutions", Los Alamos National Laboratory, Tech. Rep. LA-UR,07-2849, 2007.

[89] J. Saltzman and P. Colella, "Second Order Upwind Transport Methods for Lagrangian Hydrodynamics", Los Alamos National Laboratory, Tech. Rep. LA-UR 85-678, 1985.

[90] M. Kucharik, R. V. Garimella, S. P. Schofield, and M. J. Shashkov, "A comparative study of interface reconstruction methods for multi-material ALE simulations", *J. Comput. Phys.*, vol. 229, no. 7, pp. 2432–2452, 2010, ISSN: 10902716. DOI: `10.1016/j.jcp.2009.07.009`.

[91] V. P. Chiravalle and N. R. Morgan, "A 3D finite element ALE method using an approximate Riemann solution", *Int. J. Numer. Methods Fluids*, vol. 83, no. 8, pp. 642–663, 2017, ISSN: 10970363. DOI: `10.1002/fld.4284`.

[92] W. D. Schulz, "Two-Dimensional Lagrangian Hydrodynamic Difference Equations, ADA385003", California. Univ., Livermore. Lawrence Radiation Lab., Tech. Rep., 1963.

[93] N. V. Mikhailova, V. F. Tishkin, N. N. Tyurina, A. P. Favorskii, and M. Y. Shashkov, "Numerical modelling of two-dimensional gas-dynamic flows on a variable-structure mesh", *USSR Comput. Math. Math. Phys.*, vol. 26, no. 5, pp. 74–84, 1986, ISSN: 00415553. DOI: `10.1016/0041-5553(86)90043-1`.

[94] A. Solov'ev and M. Shashkov, "Difference scheme for the Dirichlet particle method in cylindrical method in cylindrical coordinates, conserving symmetry of gas-dynamical flow", *Differ. Equations*, vol. 24, pp. 817–823, 1988.

[95] P. P. Whalen, "Algebraic Limitations on Two-Dimensional Hydrodynamics Simulations", *J. Comput. Phys.*, vol. 124, no. 1, pp. 46–54, 1996, ISSN: 0021-9991. DOI: `10.1006/JCPH.1996.0043`.

[96] E. Caramana, D. Burton, M. Shashkov, and P. Whalen, "The Construction of Compatible Hydrodynamics Algorithms Utilizing Conservation of Total Energy", *J. Comput. Phys.*, vol. 146, no. 1, pp. 227–262, 1998, ISSN: 0021-9991. DOI: `10.1006/JCPH.1998.6029`.

[97] A. Barlow, D. Burton, and M. Shashkov, "Compatible, energy and symmetry preserving 2D Lagrangian hydrodynamics in rz — cylindrical coordinates", *Procedia Comput. Sci.*, vol. 1, no. 1, pp. 1893–1901, 2010, ISSN: 1877-0509. DOI: `10.1016/J.PROCS.2010.04.212`.

[98] L. G. Margolin, M. Shashkov, and P. K. Smolarkiewicz, "A discrete operator calculus for finite difference approximations", *Comput. Methods Appl. Mech. Eng.*, vol. 187, no. 3-4, pp. 365–383, 2000, ISSN: 0045-7825. DOI: `10.1016/S0045-7825(00)80001-8`.

[99] Z. Shen, G. Yuan, Y. Jingyan, and X. Liu, "A cell-centered Lagrangian scheme in two-dimensional cylindrical geometry", *Sci. China, Ser. A Math.*, vol. 51, no. 8, pp. 1479–1494, 2008, ISSN: 1006-9283. DOI: `10.1007/s11425-008-0121-0`.

[100] J. Cheng and C.-W. Shu, "A cell-centered Lagrangian scheme with the preservation of symmetry and conservation properties for compressible fluid flows in two-dimensional cylindrical geometry", *J. Comput. Phys.*, vol. 229, no. 19, pp. 7191–7206, 2010, ISSN: 10902716. DOI: `10.1016/j.jcp.2010.06.007`.

[101] J. Cheng and C.-W. Shu, "Improvement on Spherical Symmetry in Two-Dimensional Cylindrical Coordinates for a Class of Control Volume Lagrangian Schemes", *Commun. Comput. Phys.*, vol. 11, no. 4, pp. 1144–1168, 2012, ISSN: 1815-2406. DOI: `10.4208/cicp.030710.131210s`.

[102] X. Liu, N. R. Morgan, and D. E. Burton, "Lagrangian discontinuous Galerkin hydrodynamic methods in axisymmetric coordinates", *J. Comput. Phys.*, vol. 373, pp. 253–283, 2018, ISSN: 10902716. DOI: `10.1016/j.jcp.2018.06.073`.

[103] R. Loubère, P. H. Maire, and M. Shashkov, "ReALE: A Reconnection Arbitrary-Lagrangian-Eulerian method in cylindrical geometry", *Comput. Fluids*, vol. 46, no. 1, pp. 59–69, 2011, ISSN: 00457930. DOI: `10.1016/j.compfluid.2010.08.024`.

[104] D. Burton, N. Morgan, and T. Carney, "On the question of area weighting in cell-centered hydrodynamics, LA-UR-13-23155.2", Los Alamos National Lab.(LANL), Los Alamos, NM (United States), Tech. Rep., 2013.

[105] J. D. Huba, *NRL Plasma Formulary.* Naval Research Laboratory Washington, DC, 2019.

[106] S. V. Coggeshall and J. Meyer-ter Vehn, "Group-invariant solutions and optimal systems for multidimensional hydrodynamics", *J. Math. Phys.*, vol. 33, no. 10, pp. 3585–3601, 1992, ISSN: 00222488. DOI: `10.1063/1.529907`.

[107] L. Margolin, M. Shashkov, and M. Taylor, "Symmetry-preserving discretizations for lagrangian gas dynamics", in *Proceedings of the 3rd European Conference, Numerical Mathematics and Advanced Applications*, World Scientific Publisher, 2000, pp. 725–732.

[108] J. A. Trangenstein and P. Colella, "A higher-order Godunov method for modeling finite deformation in elastic-plastic solids", *Commun. Pure Appl. Math.*, vol. 44, no. 1, pp. 41–100, 1991, ISSN: 00103640. DOI: `10.1002/cpa.3160440103`.

[109] B. Howell and G. Ball, "A Free-Lagrange Augmented Godunov Method for the Simulation of Elastic–Plastic Solids", *J. Comput. Phys.*, vol. 175, no. 1, pp. 128–167, 2002, ISSN: 0021-9991. DOI: `10.1006/JCPH.2001.6931`.

[110] V. A. Dobrev, T. V. Kolev, and R. N. Rieben, "High order curvilinear finite elements for elastic–plastic Lagrangian dynamics", *J. Comput. Phys.*, vol. 257, pp. 1062–1080, 2014, ISSN: 0021-9991. DOI: `10.1016/J.JCP.2013.01.015`.

[111] J.-B. B. Cheng, E. F. Toro, S. Jiang, M. Yu, and W. Tang, "A high-order cell-centered Lagrangian scheme for one-dimensional elastic–plastic problems", *Comput. Fluids*, vol. 122, pp. 136–152, 2015, ISSN: 0045-7930. DOI: `10.1016/J.COMPFLUID.2015.08.029`.

[112] B. Panicaud, E. Rouhaud, G. Altmeyer, M. Wang, R. Kerner, A. Roos, and O. Ameline, "Consistent hypo-elastic behavior using the four-dimensional formalism of differential geometry", *Acta Mech.*, vol. 227, no. 3, pp. 651–675, 2016, ISSN: 0001-5970. DOI: `10.1007/s00707-015-1470-8`.

[113] M. Wang, B. Panicaud, E. Rouhaud, R. Kerner, and A. Roos, "Incremental constitutive models for elastoplastic materials undergoing finite deformations by using a four-dimensional formalism", *Int. J. Eng. Sci.*, vol. 106, pp. 199–219, 2016, ISSN: 0020-7225. DOI: `10.1016/J.IJENGSCI.2016.06.006`.

[114] S. Gavrilyuk, N. Favrie, and R. Saurel, "Modelling wave dynamics of compressible elastic materials", *J. Comput. Phys.*, vol. 227, no. 5, pp. 2941–2969, 2008, ISSN: 0021-9991. DOI: `10.1016/j.jcp.2007.11.030`.

[115] G. H. Miller and P Colella, "A High-Order Eulerian Godunov Method for Elastic–Plastic Flow in Solids", *J. Comput. Phys.*, vol. 167, no. 1, pp. 131–176, 2001, ISSN: 0021-9991. DOI: `10.1006/jcph.2000.6665`.

[116] G. Kluth and B. Després, "Discretization of hyperelasticity on unstructured mesh with a cell-centered Lagrangian scheme", *J. Comput. Phys.*, vol. 229, no. 24, pp. 9092–9118, 2010, ISSN: 0021-9991. DOI: `10.1016/j.jcp.2010.08.024`.

[117] M. Aguirre, A. J. Gil, J. Bonet, and C. H. Lee, "An upwind vertex centred Finite Volume solver for Lagrangian solid dynamics", *J. Comput. Phys.*, vol. 300, pp. 387–422, 2015, ISSN: 0021-9991. DOI: `10.1016/j.jcp.2015.07.029`.

[118] W. Boscheri, M. Dumbser, and R. Loubère, "Cell centered direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume schemes for nonlinear hyperelasticity", *Comput. Fluids*, vol. 134, pp. 111–129, 2016, ISSN: 0045-7930. DOI: `10.1016/J.COMPFLUID.2016.05.004`.

[119] A. F. Bower, *Applied mechanics of solids.* CRC press, 2009.

[120] I. Peshkov and E. Romenski, "A hyperbolic model for viscous Newtonian flows", *Contin. Mech. Thermodyn.*, vol. 28, no. 1-2, pp. 85–104, 2016, ISSN: 0935-1175. DOI: `10.1007/s00161-014-0401-6`.

[121] J. Hallquist, "LS-DYNA Theory Manual", *Livermore Softw. Technol. Corp.*, vol. 3, pp. 25–31, 2006.

[122] H. S. Udaykumar, L. Tran, D. M. Belk, and K. J. Vanden, "An Eulerian method for computation of multimaterial impact with ENO shock-capturing and sharp interfaces", *J. Comput. Phys.*, vol. 186, no. 1, pp. 136–177, 2003, ISSN: 00219991. DOI: `10.1016/S0021-9991(03)00027-5`.

[123] W. N. Weseloh, S. P. Clancy, and J. W. Painter, “PAGOSA physics manual, LA-14425-M”, Los Alamos National Lab., Los Alamos, NM (United States), Tech. Rep., pp. 1–249.

[124] R. Kamm, James, J. S. Brock, S. T. Brandon, D. L. Cotrell, B. Johnson, and P. Knupp, “Enhanced Verification Test Suite for Physics Simulation Codes, LLNL-TR-411291”, Lawrence Livermore National Lab. (LLNL), Livermore, CA (United States), Los Alamos, NM, Tech. Rep., 2008. DOI: `10.2172/950084`.

[125] G. I. Taylor, “The use of flat-ended projectiles for determining dynamic yield stress I. Theoretical considerations”, *Proc. R. Soc. London. Ser. A. Math. Phys. Sci.*, vol. 194, no. 1038, pp. 289–299, 1948, ISSN: 2053-9169. DOI: `10.1098/rspa.1948.0081`.