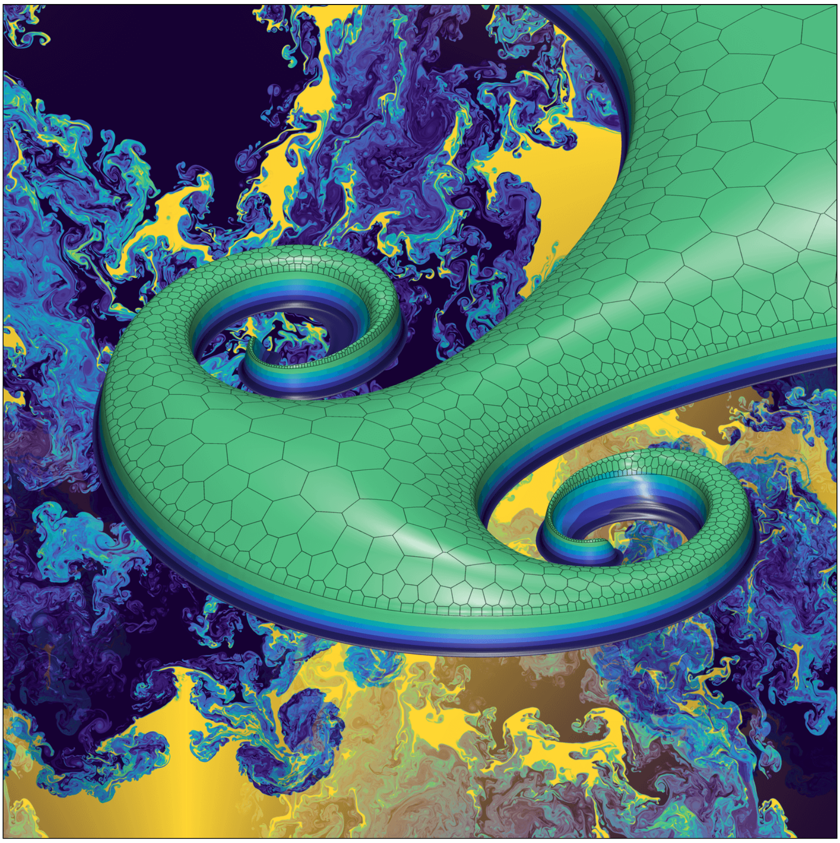
UNIVERSITY OF TRENTO

Doctoral School in Civil, Environmental and
Mechanical Engineering

Curriculum 1. Civil and Environmental Engineering

Simone Chiocchetti

High order numerical methods for a unified theory of fluid and solid mechanics



2022 Doctoral thesis

UNIVERSITY OF TRENTO - Italy
Department of Civil, Environmental
and Mechanical Engineering

Doctoral School in Civil, Environmental and Mechanical Engineering
Topic 1. Civil and Environmental Engineering - XXXIV cycle 2021/2022

Doctoral Thesis - June 2022

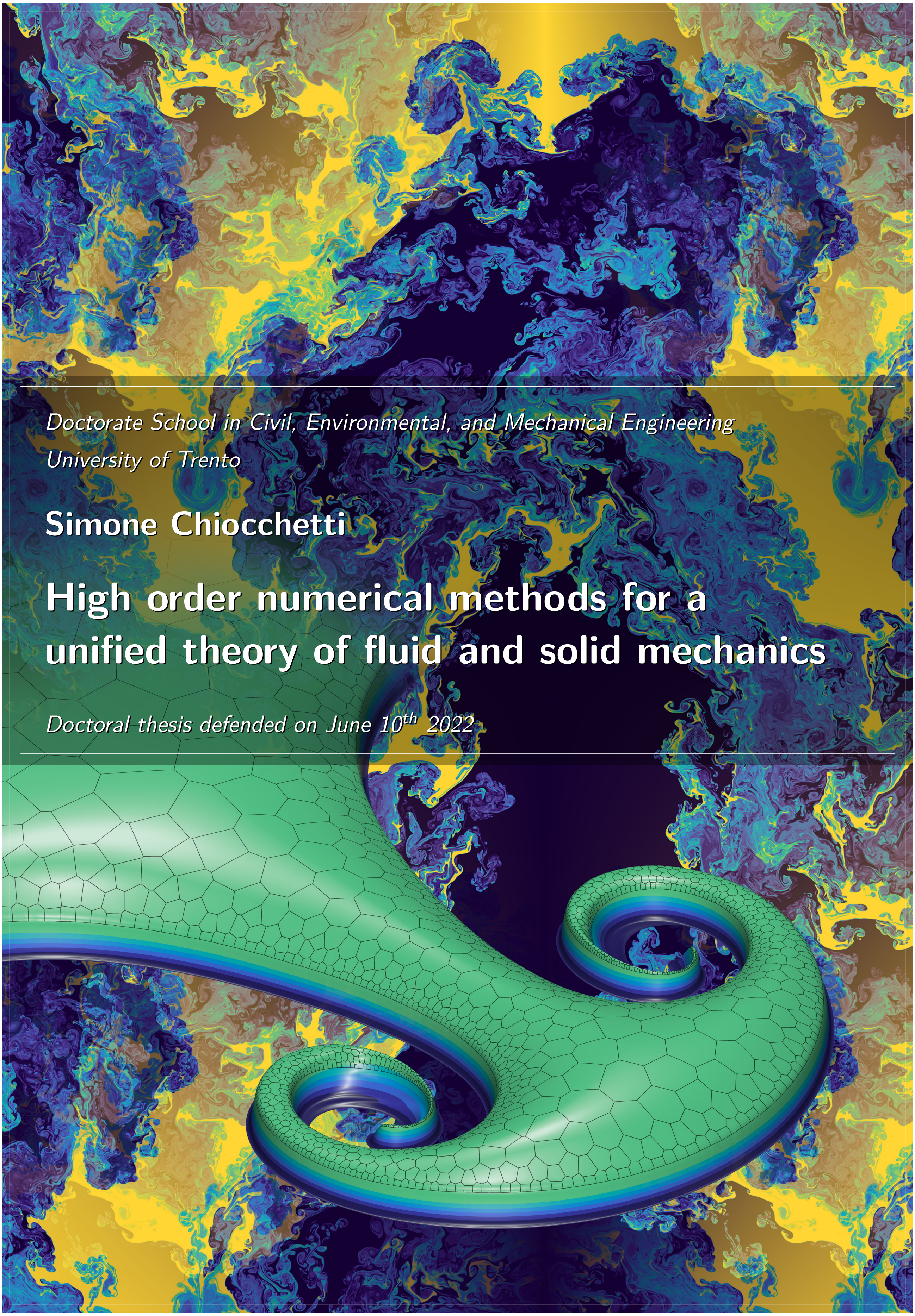
Simone Chiocchetti

High order numerical methods for a unified theory of fluid and solid mechanics

Supervisors
Supervisor: Prof. Dr.-Ing. Michael Dumbser (UNITN)
Co-supervisor: Dr. Ilya Peshkov (UNITN)

Credits of the cover image: Simone Chiocchetti

University of Trento
Doctoral School in Civil, Environmental and Mechanical Engineering
http://web.unitn.it/en/dricam
Via Mesiano 77, I-38123 Trento
Tel. +39 0461 282670 / 2611 - *dicamphd@unitn.it*



Doctorate School in Civil, Environmental, and Mechanical Engineering

University of Trento

Simone Chiocchetti

High order numerical methods for a unified theory of fluid and solid mechanics

Doctoral thesis defended on June 10th 2022

Doctoral thesis in
Civil, Environmental, and Mechanical Engineering,
Cycle XXXIV

Departement of Civil, Environmental, and Mechanical Engineering
University of Trento
Academic year 2021/2022

Advisor **Prof. Dr.-Ing. Michael Dumbser**
Co-advisor **Dr. Ilya Peshkov**

Defended on *June 10th 2022*

University of Trento,
Trento, Italy,
2022

Contents

Acknowledgements

I take this page to say a heartfelt *thank you* to the people that helped me, in Science and in Life, during my PhD studies.

I begin by expressing my gratitude towards my outstanding advisor Michael Dumbser, who has a remarkable aptitude, among other ones, for finding a way for everybody and for steering them in the right direction. I have to say thanks for caring so much and for pushing me when I needed it, while understanding when I wanted some more tinkering to be personally satisfied with my work.

Then I would like to acknowledge Prof. Sergey Gavrilyuk and Prof. Claus-Dieter Munz for refereeing the thesis, as well as for the time I have spent in Marseille and Stuttgart, and for the many teachings I received from them, both those given directly and those passed down to me through Michael's words.

Much appreciation to Prof. Giovanni Russo for hosting me in Catania with the warmest manners, and to Matteo Semplice, for logistics and for kindly answering a thousand variations of the same question. Cheers to Alessandro for printing many stacks of paper over the years. Also, thanks to Raphaël Loubère, for being in my view an all-around inspirational person.

Thanks to Ilya, because he dreams about science and maintains the dream alive for everybody, and to Elena, for saying she liked the compass and for being a good friend by sending the occasional reminder that I should eat and sleep.

To the friends that I have met in Mesiano (fourth and fifth floor): Maurizio, Walter, Matteo, Francesco, Ilya, Saray, Elena, Firas, Laura, Davide, Christian, thank you. At work, eating out (pizza, restaurant, home, fresh air), at conferences abroad, and in Italy, through good times and bad times (in the days of my youth...), thanks to you the PhD years have been that much more enjoyable and full of welcome surprises.

Additionally, I would like to thank Anne, Patrick, Martina, Carlo, Christoph, Jonas, Anna, Jens, Daniel, Pascal, Marcel, i.e. the cool people from Dropit and the IAG, for their help in my Stuttgart adventures.

Many thanks to Elena, Ilya, Firas, and Andrea for last-minute proofreadigng.

Finally, the most special thanks go to my brother Massimino, my mother Ornella, and my father Paolo. To Massimino, because he helps keeping things together, even though sometimes he might be clumsy with physical objects. To my father, because he taught me how to learn new things, which ended up being exactly my job in Life. To my mother, because she is the kindest person on the planet (not a small or trivial feat) and motivates me to try to be better.

Funding

This work has been conducted with the financial support received by the Deutsche Forschungsgemeinschaft (DFG) under the project *Droplet Interaction Technologies (DROPIT)*, grants GRK 2160/1 and GRK 2160/2.

Part of this work was carried out with the financial support from the European Union's Horizon 2020 Research and Innovation Programme under the project *ExaHyPE*, grant agreement number no. 671698 (call FET-HPC-1-2014).

Part of this work has been performed under the Project HPC-EUROPA3 (INFRAIA-2016-1-730897), through the EC Research Innovation Action under the H2020 Programme, in particular, with the support of Prof. Dr. rer. nat. Claus-Dieter Munz at IAG and the supercomputing facilities provided by the HLRS in Stuttgart.

Financial support was received also from the Italian Ministry of Education, University and Research (MIUR) in the frame of the Departments of Excellence Initiative 2018–2022 attributed to DICAM of the University of Trento (grant L. 232/2016) and in the frame of the PRIN 2017 project *Innovative numerical methods for evolutionary partial differential equations and applications*, as well as from the University of Trento via the *Strategic Initiative Modeling and Simulation.*

Abstract

In this thesis, I detail my contribution to the development of a unified mathematical model describing both fluids and elastic solids as special cases of a general continuum, with a simple material parameter choice being the distinction between inviscid or viscous fluid, with Newtonian or non-Newtonian rheology, or elastic solids or visco-elasto-plastic media. My contribution regards in equal measure theoretical aspects of modelling and numerical algorithm design.

The model takes the form of a monolithic system of first order hyperbolic partial differential equations (PDEs), and, besides a unified description of fluids and solids, I show how additional physical effects such as surface tension, rate-dependent material failure and fatigue can be, and have been, included in the same formalism.

The origins of the model can be traced back to the work of Godunov and Romenski in the sixties and seventies, who introduced an hyperelastic formulation of solid mechanics in Eulerian coordinates, counter to the Lagrangian description of the discipline which to this day is prevalent in the solid mechanics community. Such Eulerian equations of solid mechanics feature algebraic relaxation source terms which are sufficient to extend the applicability of the model not only to elasto-plastic solids, but, remarkably, also to fluid flows.

From the standpoint of numerical computation, the development of robust and accurate Finite Volume (FV) and Discontinuous Galerkin (DG) methods for such a general system of partial differential equations poses many challenges that have been overcome through the research presented in this thesis.

On the details of my research work, I begin by addressing the problems of weak hyperbolicity and curl involutions in the hyperbolic equations of capillarity by means of a GLM curl cleaning method, which is an extension to curl involutions of the successful Generalized Lagrangian Multiplier approach for enforcing divergence constraints in the Maxwell equations of electrodynamics and in the equations of Magnetohydrodynamics (MHD).

I then tackle the same problem again from a completely different angle, adopting special involution-preserving discrete differential operators, which, combined with a staggered mesh discretisation of the governing equations, allow

the efficient computation of multiphase low Mach number flows with hyperbolic viscosity and hyperbolic surface tension.

I implement this new numerical algorithm in a distributed memory MPI-parallel Fortran code, which I then benchmark on the HPE–Hawk supercomputer at the HLRS in Stuttgart utilising up to 65 536 CPU cores and handling simulations involving billions of degrees of freedom. A semi-implicit discretisation of the pressure equation, rather than a fully implicit one for the complete PDE system, leads to a small-size, symmetric positive definite discretised linear system that can be solved efficiently with a Jacobi-preconditioned, matrix-free, highly optimized conjugate gradient method. This directly translates to much higher parallel scalability with respect to large scale supercomputer codes based on implicit schemes.

With regard to the solution of the stiff algebraic relaxation source terms, I introduce a new family of efficient and accurate exponential-type and semi-analytical time-integration methods for the stiff source terms which govern friction and pressure relaxation in Baer–Nunziato compressible multiphase flows, as well as for the strain relaxation system of the unified model of continuum mechanics, associated with viscosity and plasticity, and heat conduction effects. These novel semi-analytical methods have been proven very robust, accurate, and efficient towards a variety of applications, having been seamlessly integrated in several radically different computational codes.

Concerning applicability of the model to real world problems, besides those whose solution is enabled by the aforementioned code for viscous two-phase flow, in this thesis I provide results with regards to crack formation in solids, leveraging a diffuse interface approach, again in the framework of the unified model of continuum mechanics. In this context I show that with relatively small formal modifications, the model can describe not only visco-elasto-plastic solids, viscous fluids, and inviscid fluids, but also material damage and in particular brittle and ductile behaviour, as well as material fatigue. The governing equations in this case are solved by means of high order ADER Discontinuous Galerkin and Finite Volume schemes on both fixed Cartesian meshes, which can be employed even for complex geometries, thanks to the diffuse representation of interfaces, but also on moving unstructured polygonal meshes with adaptive connectivity, which are constructed and whose motion is controlled by means of a Fortran library for the generation of high quality Delaunay and Voronoi meshes, written as a stand-alone tool in the early days of my research career.

Most importantly, the development of such numerical algorithms constituted an invaluable opportunity for gaining knowledge on the rich mathematical structure of the equations of the unified model of continuum mechanics.

1 Introduction

1.1 On the importance of a unified understanding of the laws of Nature

Throughout history, the study of physical phenomena features a tendency to progress from observing and cataloguing objects and interactions, towards more fundamental fine scale unifying theories.

A prime example of this propensity is provided by the evolution of our understanding of the composition of matter from the early atomistic view of Democritus (circa 380 BC) who thought of atoms as present in Nature in infinitely many variants, to Empedocle's theory (circa 420 BC) stating that all matter is composed of four fundamental substances, Water, Earth, Wind, and Fire. Although the understanding of both Greek philosophers was somehow far from modern theories of matter, one must commend the extraordinary intuition brought forth by each of the two.

Exactly the same trend can be observed about twenty centuries later: a taxonomic understanding of compounds and their transformations in alchemy and later chemistry constituted the state of the art in research in the seventeenth century and until the end of the eighteenth century, when the observations by Lavoisier, Proust, and Dalton concerning conservation of mass and stoichiometry in chemical processes suggested that the vast variety of different compounds found in Nature might be indeed explained in terms of a smaller set of more fundamental unit particles.

An extensive amount of insight based on similar numerical consideration on the masses of different compounds has been produced during the following century, with a notable example being the periodic tables of elements by Lothar Meyer (1864, 1870) and Mendeleev (1869), both predating not only the experiments of Rutherford, Geiger, and Marsden in the first decade of the twentieth century, which will eventually lead to the discovery of protons and neutrons, but also Thompson, Thownsend and Wilson discovering in 1897 "corpuscles" that would later be named electrons. This latter finding marks a colossal milestone towards the modern unified description of matter in terms of a relatively small number of elementary particles.

Concerning the physical forces of Nature, the search for a single unifying theory is again apparent: Maxwell provided a unified description of light, electricity and magnetism, which is modernly understood as a relativistic effect arising from space dilation observed by moving electric charges, causing an apparent charge imbalance perpendicular to the direction of the motion [140, 234]. Furthermore, in the second half of the twentieth century, Glashow, Salam, and Weinberg, introduced a theory unifying electromagnetism with the weak nuclear interaction, first introduced in 1933 by Enrico Fermi as an explanation of beta decay. Hence, electromagnetism and weak nuclear interactions can be interpreted as two different manifestations, at different energy scales, of a single unified electroweak interaction.

Since then, the construction of a single model for the description of the behaviour of matter is widely regarded as one of the greatest goals of research in Physics.

The aim of this thesis is significantly more modest, but motivated by the same tendency, in that our objective is the development and numerical solution of a set of hyperbolic partial differential equations capable of describing a generic continuum in a unified manner. This means that elastic solids, perfect fluids, viscous fluids and visco-elasto-plastic solids are treated as a general continuum without specifying explicitly of what kind, if not by different material parameters. In the following section we briefly describe the origin and features of such a mathematical model.

1.2 A unified theory of fluids and solids

In light of the existence of much more sophisticated models of Physics, the development of a unified model of continuum mechanics as a theory for uncovering the underlying structure of Nature, might appear to be a futile endeavour. However, the practical need for macroscopic models of the world is undeniable and motivated by lack of computational power, or to be fair, by the enormous amount of computational resources required to accurately reproduce even the simplest of physical systems *ab initio*. This is true not only if one considers *ab initio* to be as starting from the fundamental subatomic forces, but even more practically oriented strategies such as molecular dynamics [188, 189, 308] or Direct Simulation Monte Carlo in gas flows [134, 251, 270, 271], albeit invaluable for specific applications for which they are designed, are computationally too demanding for many practical purposes. Furthermore, even continuum-level models like the Navier–Stokes equations, often must be supplemented with large eddy simulation [160, 233, 246, 262, 303, 331] or coarser scale turbulence models [62, 128, 300], in order to be applicable for the solution of real-world engineering problems. For this reason, a general model of continuum mechanics seems to be a promising platform on which to build such novel macroscopic models.

The modelling of fluid motion and that of solids as continuous media stands at the basis of modern advanced simulations in engineering, from structural finite element analysis, to aerodynamics, fluid-structure interaction, acoustics, sediment transport in river engineering. The descriptions of fluids and solids are strongly interlinked, especially through the common language of conservation and balance laws for mass, momentum, and energy routinely used for stating the time-dependent governing equations of the media, and by a common mathematical framework established in the nineteenth century at the hand of, among others, Cauchy, Navier, d'Alembert, Poisson, Laplace, Stokes, Newton, Euler: this mathematical formalism naturally highlights analogies and shared concepts between the two disciplines, first and foremost the geometric description of deformations and strain, and the presence of compatibility constraints in solid mechanics and incompressible fluid flows.

Despite such a strongly established theoretical link, current computational algorithms are generally built around balance laws with fluids being modelled adopting rheologies that are radically different from those in use for solids. Fluids and solids are generally recognised as different states of matter and studied by two largely distinct communities: in most fluid flow applications, the observer point of view is Eulerian, while solid mechanics is generally studied in Lagrangian coordinates. Some fluid-structure or multiphase flow applications also employ two or more different models, one for each of the involved media, with additional coupling rules regulating the connection between separate phases.

One might then ask whether such a unified theory is really necessary, since very effective numerical schemes for fluids, for solids, and for fluid-solid interaction already exist and are routinely employed in the engineering practice. To counter this point, one well known argument in favour of abandoning the distinction between solids and fluids can be found by considering the Earth's mantle, which is uncontroversially known to be a solid, in which indeed secondary (transverse) seismic waves can propagate, unlike in the outer nucleus of the Planet, which is a fluid. Nevertheless, on geological timescales (millions of years), the mantle has been observed to behave like a viscous fluid, due to the presence of defects and dislocations that, by introducing small local slip surfaces in the medium, give raise to fluid-like motion, when their effects are averaged over sufficiently large control volumes and over long timescales.

The same is observed on the opposite side of the spectrum of continua: if oscillating forces with sufficiently fast frequency are applied to a fluid, e.g. ultrasound waves propagating in water, the fluid will react just in the same way a solid would [54], which is an effect that cannot be explained by the Navier–Stokes equations, but is already naturally foreseen by the *unified model of continuum mechanics* introduced by Godunov and Romenski in [165, 168–170] as an Eulerian formulation of hyperelasticity with strain relaxation adopted for modelling plastic effects.

In [257], Peshkov and Romenski, advanced the key insight that the Godunov–Romenski model may be applied not only to elasto-plastic solids, but to fluid flows as well, up to then unexplored, with the very notable exception of the early paper of Besseling [30], in which an equivalent formalism was introduced.

Each partial differential equation in the system is of first order in space and time and the model as a whole is of hyperbolic nature, that is, all signals propagate through space by means of waves of finite speed. Nonetheless, the model can reproduce the classical mixed hyperbolic/parabolic Navier–Stokes–Fourier theory as a limit case.

The model is a product of the very general formalism of Symmetric Hyperbolic Thermodynamically Compatible systems (SHTC) [163, 166, 256, 279], that can be written in symmetric hyperbolic Godunov form, and can be shown to satisfy both the first principle of thermodynamics (the law of conservation of energy), and the second principle of thermodynamics (the entropy inequality).

In [120] such an approach was proven viable by Dumbser and collaborators, by solving the governing equations not only in the solid regime, but also in the fluid one by means of high order ADER schemes leveraging the approach of [112] for the integration of stiff sources. In [255, 328] the model was solved again, this time with Arbitrary-Lagrangian-Eulerian (ALE) ADER-WENO Finite Volume schemes. The approach is adopted in [24, 53, 137–139, 159, 175, 242, 243] for the study of elasto-plastic solids, and in [194, 195] with explicit reference to the fluid applications proposed by Peshkov and Romenski in [257].

1.2.1 Everything is a solid

The unified model of continuum mechanics object of this work was introduced in [257] and is based on the Eulerian formulation of hyperelasticity by Godunov and Romenski [168–170]. The mathematical model and its many variants have been called differently in different contexts: HPR/GPR (from Hyperbolic–Peshkov–Romenski or Godunov–Peshkov–Romenski) in [120, 121], model for viscoplasticity in [135], unified model of continuum mechanics in [47]. In this thesis, we adopt the generic terminology *Unified model of continuum mechanics*, or occasionally UMCM or GPR, when appropriate for brevity.

While referring to Chapter 2 for a detailed exposition of the mathematical model, we introduce here a minimal formal statement of the governing equations of the unified model of continuum mechanics, which can be written as

$$\partial_t\left(\rho\right) + \nabla\cdot\left(\rho\,\mathbf{u}\right) = 0, \tag{1.1a}$$

$$\partial_t\left(\rho\,\mathbf{u}\right) + \nabla\cdot\left(\rho\,\mathbf{u}\otimes\mathbf{u} - \boldsymbol{\Sigma}\right) = \mathbf{0}, \tag{1.1b}$$

$$\partial_t\left(\rho\,E\right) + \nabla\cdot\left(\rho\,E\,\mathbf{u} - \,\boldsymbol{\Sigma}\,\mathbf{u}\right) = \mathbf{0}, \tag{1.1c}$$

$$\partial_t\left(\mathbf{A}\right) + \left(\nabla\mathbf{A}\right)\,\mathbf{u} + \mathbf{A}\left(\nabla\mathbf{u}\right) = \mathbf{Z}, \tag{1.1d}$$

where the first three equations state the conservation of mass, momentum, and energy respectively, and the fourth equation governs the kinematics of a three by three nonsymmetric matrix **A**, called *distortion field* or *cobasis*. Note in particular the presence of the *strain relaxation* source term **Z** in (1.1d), which is *the key feature* of the *unified model of continuum mechanics*: while the left hand side of (1.1d) is a pure description of kinematics and geometry of the medium, the actual material behaviour, i.e. the rheology, is specified by **Z**.

In this historically accurate view of the model (the equations of Godunov and Romenski for elasto-plastic solids have maintained their form since 1978, when they have been written in Godunov's book [165] in terms of the cobasis **A**, after appearing in Godunov and Romenski's 1972 paper [168] containing the equations for the direct time evolution of the metric tensor $\mathbf{G} = \mathbf{A}^{\mathsf{T}}\,\mathbf{A}$) one could say that fluids are just visco-elasto-plastic solids with fast strain relaxation, where "fast" is defined on the basis of a relative *timescale* which can be, for a human observer, fast in relation to the duration of an experiment, or fast due to an extremely long observation timescale, like in the case of the aforementioned mantle convection problem. In this sense, just like electromagnetism and weak interactions are special cases of a single electroweak phenomenon, separated by an energy scale, fluids can be seen as solids observed on very long timescales, such that the gradual motion of dislocations, or of other microscopic defects in a solid, can appear as fluid flow.

1.2.2 Everything is a fluid

In the formalism of the unified model of continuum mechanics, a perfectly elastic solid (i.e. with vanishing source $\mathbf{Z} = \mathbf{0}$) is a continuum having no *defects*: an analogy between dislocations in crystalline solids and gravity is discussed in [178, 208], where in particular the perfectly elastic solid limit of the equations corresponds to vacuum in spacetime. As soon as mass is present, it introduces *defects* in spacetime, which notoriously contains a nonzero amount of mass. Hence, motivated by the geometric nature and generality of the SHTC equations, it seems reasonable to consider also the interpretation of continua as fluids, in order to allow the presence of defects.

A simple mental model for the interpretation of all continua as fluids could be constructed in analogy with kinetic gas theory and statistical thermodynamics, by which macroscopic mechanisms such as pressure forces and heat conduction can be seen as emergent effects from macroscale observation.

In order to see how fluid-like or solid-like behaviour manifest from averaging of an underlying particle model, one can imagine the continuum as composed of many *rich* particles, which interact not only by exchanging linear momenta and kinetic energies (as hard spheres would) but trough elastic-type forces which compare the cobasis between two interacting particles. This means that each hypothetical particle encodes additional *geometric* information about a local strain, in general specified as a local coordinate triad. Then it is not difficult to imagine how such a particle system, after appropriate space averaging (from macroscopic observation) and time averaging (in order to allow emergent continuum features to develop from statistics of particle interactions) would react differently to external stimuli taking place over different timescales.

If the flow field changes over short times, then reactions due to tangential stresses $\boldsymbol{\sigma}_{\mathrm{s}} = -\rho\, c_{\mathrm{s}}\, \mathbf{G}\ \mathrm{dev}\, \mathbf{G}$ will be of elastic type, being caused only from the local strain characterised by the metric tensor $\mathbf{G} = \mathbf{A}^{\mathsf{T}}\, \mathbf{A}$.

On the other hand, over sufficiently long timescales, it is easy to imagine that one can recover the intended fluid-like viscous effects since the information contained in the Eulerian control volume is no longer pertaining only instantaneous (local) information on the deformation of the medium but rather, if enough time has elapsed, nonlocal information about the gradients of the flow field will be encoded in the local averaged state of the continuum, and specifically in our case effects proportional to strain rate tensor $\mathbf{D} = \left(\nabla\mathbf{u} + \nabla\mathbf{u}^{\mathsf{T}}\right)/2$ can manifest from local information.

On a more simplistic note, the model object of this work can also be seen as an augmentation of fluid flow models such as Euler's equations or Kapila's system [201]. This last practical point of view is, strictly speaking, *incorrect*: the model is derived as a whole from variational principles and geometrical constraints within the formalism of Symmetric Hyperbolic Thermodynamically Compatible systems (SHTC) [163, 166, 256, 279]. Models obtained through the SHTC framework can be cast in symmetric hyperbolic Godunov form and can be shown to satisfy the first and second principle of thermodynamics, meaning that one can show the existence of both an increasing physical entropy function and of a total energy that satisfies a conservation law. Nevertheless, such an interpretation, that is, viewing the governing equations as an augmentation of a previously known model, can be useful in practice in order to recognize and isolate the parts of the system that can be treated with tried and tested techniques, from those that require the development of entirely new schemes and approaches.

1.3 First order hyperbolic equations

This thesis work is dedicated entirely to the study and solution of partial differential equations of first order hyperbolic type [268]. In the following we provide some motivation with regard to this choice.

1.3.1 Causality

An attractive theoretical feature of hyperbolic equations is that they satisfy and mimic two important, widely accepted, and interlinked principles of Nature, namely hyperbolic systems propagate information by means of waves moving with *finite speed* and the *causality principle* is satisfied by construction. This stands in net contrast to equations of parabolic type, such as the Fourier law of heat conduction, for which the effects of *any* change in temperature at *any* location in space are observed *everywhere* and *immediately.*

An apparent exception to the finite speed of propagation of information guaranteed by hyperbolic systems is the presence of stationary differential constraints which must be satisfied everywhere and at all times by solutions to the governing equations, so called *involutions* in certain hyperbolic PDE systems such as the Maxwell equations of electrodynamics, or in many first order hyperbolic reformulations of higher order systems such as the Serre–Green–Naghdi equations [25, 60, 136, 172], the Euler–Korteweg/Van der Waals system, which includes the nonlinear Schrödinger equation written in hydrodynamic form [60, 99] as a special case, surface tension forces in compressible flows [29, 286], or the Z4 formulation of Einstein's Field equations [33, 34, 115]. These involution constraints however, at least at the continuum level, are nothing but a consequence of the structure of the governing equations and it can be shown that they define properties that are automatically satisfied by the solutions of the governing equations, (with compatible initial conditions) and thus require no information to be exchanged in order to hold at all times.

Then, unlike parabolic PDEs, including the Navier–Stokes–Fourier system, first order hyperbolic systems conform to the so called *causality principle*: in the language of Special Relativity [132], causality can be seen as a consequence of finite speed of propagation of light and information in general. Spacetime is partitioned into regions that can be influenced by events happening at a certain spacetime location (the future light cone) or that could possibly have transmitted information to the same spacetime position (the past light cone), or, alternatively, portions of spacetime towards which no information can be sent or from which information cannot be received. In the same way, in a classical non-relativistic world, even if the wave propagation speed is not part of the geometrical structure of spacetime but simply is set by the type of interactions (particle collisions for acoustic waves for example) through which such waves originate, it is entirely reasonable to require that events not influence far regions of space without some physical mechanics to transmit information, and indeed *action at a distance* in general has long been considered a problem [234] to be eliminated from models in theoretical Physics.

1.3.2 Explicit schemes for hyperbolic equations

When constructing explicit methods for hyperbolic partial differential equations, the light cone concept is reflected by the so called Courant–Friedrichs–Lewy (CFL) condition [91], which expresses a maximum timestep size that can be achieved while maintaining stability of time integration, for a given mesh size and a given numerical scheme. In its simplest form, the condition states that information used to update the state vector $\mathbf{Q}$ at a certain mesh point or control volume from a time level t^n to the subsequent time level $t^{n+1} = t^n + \Delta t$, can originate only from within a certain cone of influence. This is achieved by imposing, everywhere in the computational domain, $\Delta t \leq \Delta s / \lambda$, where Δs indicates a local grid size and λ the maximum signal propagation speed.

In the case of explicit numerical integration of parabolic equations, a similar timestep restriction can be derived, but the proportionality between the timestep Δt and the mesh size Δs is quadratic. Despite being less restrictive than a hypothetical law stating that no explicit integration of parabolic equations can be carried out (which would reflect the infinite propagation speed of information), a timestep restriction requiring $\Delta t \sim \Delta s^2$ can become prohibitively expensive on fine meshes and for this reason implicit schemes that solve the governing equations with global coupling are necessary. Such methods require the solution of a system of discrete equations, which can be computationally very demanding and may hinder the scalability of numerical codes on massively parallel supercomputers.

Thus, our interest in hyperbolic equations is motivated not only by aesthetic considerations about the fact that mathematical models should reflect the structure and principles of the underlying reality, but also from practical considerations in regard to the design of highly efficient computational methods.

1.4 Structure of the thesis and main contributions

The remaining part of the dissertation is composed of six parts, alternating theoretical considerations on the modelling aspects, with the development of numerical schemes for the solutions of the same equations:

In **Chapter 2**, the mathematical modelling framework adopted in this thesis is discussed, with reference to several systems in the SHTC family of equations. I illustrate the main features of the unified model of continuum mechanics, the origin of involution constraints, the necessary steps for the inclusion of surface tension forces, and I derive accurate eigenvalue estimates for the model. In the same chapter, I also detail the extension of the hyperbolic generalized Lagrangian Multiplier (GLM) divergence cleaning approach [94, 240] to curl involutions.

In **Chapter 3** are presented the high order ADER schemes [105, 119, 289, 315, 316, 320] of Discontinuous Galerkin and Finite Volume type, on Cartesian meshes , and on moving unstructured polygonal grids with dynamic connectivity. The construction and motion of the unstructured polygonal grids employed by the numerical method described in this chapter are controlled by means of a Fortran library for the generation of high quality Delaunay and Voronoi meshes, written as a stand-alone tool in the early days of my research career, and based on the illuminating work of Lawson [210], Chew [84], Ruppert [281], Shewchuck [292–295], Erten and Üngör [133, 324, 325].

In **Chapter 4**, the development of numerical schemes and the mathematical analysis of the governing equations are tightly woven together. In my personal journey in the study of the equations of Sergey Godunov, Evgeniy Romenski, Ilya Peshkov and Sergey Gavrilyuk [29, 165, 168–170, 257, 286], this chapter represents a fundamental milestone opening new perspectives towards the concrete, practical, immediate applicability of the unified model of continuum mechanics, and for future interpretations of these equations. Specifically, I discuss the mathematical structure of the stiff relaxation source terms present in the equations, which are a major challenge in their solution, and detail the numerical techniques that I developed specifically in order to tackle this interesting problem.

In **Chapter 5**, I introduce a novel staggered, semi-implicit, curl-preserving method for the solution of the hyperbolic formulation of viscous two-phase flow with surface tension, derived starting from [43, 109, 192, 198, 323]. This method is the basis for a highly efficient MPI-parallel Fortran code that constitutes a further step towards the applicability of the unified model of continuum mechanics in real world problems.

In **Chapter 6** an extensive amount of numerical experiments is presented and discussed, with the aim of validating the numerical methods developed in this work, as well as investigating the behaviour of the different variants of the unified model of continuum mechanics here presented. Special attention is given to the role of curl involutions in the surface tension model of Gavrilyuk and collaborators [29, 286], since it clearly shows how important it is to explicitly address the enforcement of such constraints in the discretised equations. The computational results obtained in this thesis are published in [47, 59, 86, 87, 110, 147, 149, 313], and [85] is, at the time of writing of this manuscript, to be submitted for publication.

Finally, in **Chapter 7** are listed the main insights gained while carrying out the work shown in this thesis and discuss future research directions regarding numerical algorithms and modelling perspectives enabled by the unified model of continuum mechanics.

2 A unified mathematical model for continuum mechanics

2.1 Introduction

The unified model of continuum mechanics introduced in [257], based on the Eulerian formulation of hyperelasticity by Godunov and Romenski [168–170], is characterised by the remarkable ability to describe both solids and fluids in a unified way, with the two states of matter differing simply by the choice of material parameters, specifically a timescale τ regulating the rate at which the strain of a given control volume dissipates trough a relaxation process: for $\tau \to 0$ one has inviscid flow (all shear dissipates so fast that no tangential forces are exerted in the medium); for τ small with respect to the timescale of flow convection (the so-called stiff relaxation limit) the model reproduces the Navier–Stokes equations of viscous fluids; for $\tau \to \infty$ relaxation processes are infinitely slow and the continuum behaves exactly like an elastic solid.

The PDE system is of first order in time and space (no spatial or temporal derivatives of order higher than one appear) and of hyperbolic nature, which implies that waves in the continuum propagate with finite speed. This in contrast to the infinite speed of signal propagation encountered for example in models of parabolic nature, such as the Fourier law of heat conduction or in mixed hyperbolic-parabolic models such as the Navier–Stokes equations.

Moreover, in this work, the model is extended with additional physical effects, while maintaining its first order hyperbolic nature. In particular we focus on the inclusion of a hyperbolic formulation of surface tension forces, for which in general the computation of higher order derivatives, used for the geometrical description of material interfaces, would be required, and on a diffuse interface approach to fracture mechanics, that allows the description of material failure, crack propagation, and fatigue by means of a single additional equation for the evolution of a scalar damage variable.

In the context of this thesis, several variants and implementations of the proposed unified hyperbolic model of continuum mechanics have been developed. In the following, different features of the model are presented separately by considering several different sub-systems: first we show a simplified prototype model that can describe elasto-plastic solids and viscous heat conducting fluids with surface tension, then we discuss in detail the inclusion of surface tension forces in a two-phase variant of the first order hyperbolic unified model, and finally we present a simple way of modelling material damage and diffuse solid/vacuum interfaces in the same unified first order hyperbolic framework.

2.1.1 General structure of the model

In order to introduce the main features of the unified model of continuum mechanics developed in this work, we consider a generic simplified prototype system of governing equations written as

$$\partial_t\left(\rho\right) + \nabla\cdot\left(\rho\,\mathbf{u}\right) = 0, \tag{2.1a}$$

$$\partial_t\left(\rho\,\mathbf{u}\right) + \nabla\cdot\left(\rho\,\mathbf{u}\otimes\mathbf{u} - \boldsymbol{\Sigma}\right) = \mathbf{0}, \tag{2.1b}$$

$$\partial_t\left(\rho\,E\right) + \nabla\cdot\left(\rho\,E\,\mathbf{u} - \,\boldsymbol{\Sigma}\,\mathbf{u} + \mathbf{q}\right) = \mathbf{0}, \tag{2.1c}$$

$$\partial_t\left(\mathbf{A}\right) + \left(\nabla\mathbf{A}\right)\,\mathbf{u} + \mathbf{A}\left(\nabla\mathbf{u}\right) = \mathbf{Z}, \tag{2.1d}$$

$$\partial_t\left(\mathbf{J}\right) + \nabla\left(\mathbf{J}\cdot\mathbf{u} + T\right) + \left[\nabla\mathbf{J} - \left(\nabla\mathbf{J}\right)^{\intercal}\right]\mathbf{u} = \mathbf{W}, \tag{2.1e}$$

$$\partial_t\left(\mathbf{b}\right) + \nabla\left(\mathbf{b}\cdot\mathbf{u}\right) + \left[\nabla\mathbf{b} - \left(\nabla\mathbf{b}\right)^{\intercal}\right]\mathbf{u} = \mathbf{0}, \tag{2.1f}$$

where ρ is the mass density of the continuum and (2.1a) is the continuity equation or law of conservation of mass; the momentum density or momentum for short is $\rho\,\mathbf{u} = \rho\,\left(u_1,\ u_2,\ u_3\right)^{\intercal}$ and the momentum conservation equation (Newton's second law) is (2.1b); equation (2.1c) expresses the conservation of total energy $\rho\,E$. Then, Equation (2.1d) governs the geometric evolution of a three by three nonsymmetric matrix $\mathbf{A}$, termed *distortion field*, which describes the local deformation of the medium and is used to evaluate a so-called symmetric positive definite *metric tensor* $\mathbf{G} = \mathbf{A}^{\intercal}\,\mathbf{A}$, which identifies the direction and magnitude of deformations in the continuum. If the source term $\mathbf{Z}$ in (2.1d) is set to $\mathbf{Z} = \mathbf{0}$, then the first four equations of (2.1) constitute the Godunov–Romenski model of hyperelasticity, however the simple addition of an algebraic relaxation source $\mathbf{Z}$ unlocks some major potentialities of such a model, enabling the simulation, with appropriate closure relations, of Newtonian and non-Newtonian fluids, including the Euler and Navier–Stokes equations, as well as visco-elasto-plastic solid continua, by means of the same set of partial differential equations, while retaining the first order hyperbolic form of the governing equations.

Similarly, Equation (2.1e) describes the evolution of a *thermal impulse* vector $\mathbf{J} = \left(J_1,\ J_2,\ J_3\right)^{\intercal}$, which governs heat propagation by means of thermo-acoustic waves of finite speed. The corresponding relaxation source is simply given by $\mathbf{W} = -\mathbf{J}/\tau_{\mathrm{H}}$.

It is easy to verify that, if the right hand side $\mathbf{W} = -\mathbf{J}/\tau_{\mathrm{H}}$ is stiff ($\tau_{\mathrm{H}} \to 0$), then the thermal impulse vector relaxes to $\mathbf{J} = -\tau_{\mathrm{H}}\,\nabla T$ (with T we denote the temperature), so that the heat flux $\mathbf{q} = \rho\, T\, \alpha_T^2\, \mathbf{J}$, will coincide with the Fourier heat flux in the limit $\tau_{\mathrm{h}} \to 0$, with an appropriate definition of $\tau_{\mathrm{H}} = \tau_{\mathrm{h}}\, \rho_0\, T_0\, (\rho\, T)^{-1}$. Note that (2.1e) is *not* a Cattaneo-type [79] evolution equation for the heat flux $\mathbf{q}$ itself, but rather for a new micro-momentum-like quantity $\mathbf{J}$. On a practical level, this difference is made apparent by the fact that relaxation time τ_{h} does not explicitly appear in the eigenstructure of the system. It is also important to remark that the notation $\tau_{\mathrm{h}} \to 0$ should be interpreted as stating that the relaxation timescale τ_{h} is much faster than the observation timescale, as opposed to a stricter constraint. In other words, $\tau_{\mathrm{h}} \to 0$ can also be read as $t \to \infty$, i.e. when an equilibrium state between heat impulse production and relaxation is achieved.

Concerning notation, since in the literature there are at least two different variants of such a heat conduction model [43, 120, 147] using the same symbols with slightly different meanings, a paragraph clearing up the matter is in order. Namely, sometimes the symbol c_{h} (in this work a_{h} for removing any ambiguity), referring to the linear estimate for the speed of heat waves, is adopted instead for the scaling coefficient of thermal impulse energy α_T instead of such characteristic velocity. More commonly, the symbol α (hereafter α_T to distinguish it from the volume fraction α) is taken, as is done in this work, to have SI units $[\mathrm{m^2\,s^{-2}\,K^{-1}}]$, with the thermal impulse $\mathbf{J}$ having units $[\mathrm{s\,K\,m^{-1}}]$. Sometimes instead (for example in [120]), an alternative evolution equation involving for $\rho\,\mathbf{J}$ is used in place of that for $\mathbf{J}$ given in (2.1e), in this case, α_T has SI units $[\mathrm{kg\,m^{-1}\,s^{-2}\,K^{-1}}]$ and $\rho\,\mathbf{J}$ matches the units adopted in this work for $\mathbf{J}$, so that, in [120], $\mathbf{J}$ had been given units of $[\mathrm{s\,K\,m^2\,kg^{-1}}]$. The thermal conductivity coefficient κ given in the following paragraph is always expressed in $[\mathrm{kg\,m\,s^{-3}\,K^{-1}}]$ or equivalently $[\mathrm{W\,K^{-1}\,m^{-1}}]$.

Finally, Equation (2.1f) is again a geometric law governing the evolution of a so-called *interface field* $\mathbf{b} = (b_1,\ b_2,\ b_3)^{\mathsf{T}}$, defined as the gradient $\mathbf{b} = \nabla c$ of a passively advected scalar c satisfying $\partial_t(c) + \mathbf{u}\cdot\nabla c = 0$, and employed for the computation of surface tension forces.

The total specific energy is defined as $E = e{+}e_{\mathrm{k}}{+}e_{\mathrm{s}}{+}e_{\mathrm{h}}{+}e_{\mathrm{t}}$, with the simplest choice for the internal energy e being given by the ideal gas law $\rho\, e = p/(\gamma{-}1)$, e_{k} the specific kinetic energy $e_{\mathrm{k}} = \left\lVert\mathbf{u}\right\rVert^2/2$, the specific strain energy is proportional to the second invariant of the deviator $\operatorname{dev}\mathbf{G} = \mathbf{G} - (\operatorname{tr}\mathbf{G}/3)\ \mathbf{I}$ of the metric tensor $\mathbf{G}$ and reads $e_{\mathrm{s}} = c_{\mathrm{s}}^2\ \operatorname{tr}\left(\operatorname{dev}\mathbf{G}\ \operatorname{dev}\mathbf{G}\right)/4$. The specific energy associated with the thermal impulse vector (or heat vector) $\mathbf{J}$ is $e_{\mathrm{h}} = \alpha_T^2\ \left\lVert\mathbf{J}\right\rVert^2/2$ and the specific energy due to surface tension is $e_{\mathrm{t}} = \sigma\, \|\mathbf{b}\| / \rho$. The total stress tensor $\boldsymbol{\Sigma}$ is computed as

$$\boldsymbol{\Sigma} = -p\,\mathbf{I} + \boldsymbol{\sigma}_{\mathrm{s}} + \boldsymbol{\sigma}_{\mathrm{h}} + \boldsymbol{\sigma}_{\mathrm{t}}, \tag{2.2}$$

and is composed of a spherical part $-p\,\mathbf{I} = -\rho^2\, \partial_\rho E$ (hydrodynamic pressure), a shear stress $\boldsymbol{\sigma}_{\mathrm{s}}$, a thermal stress contribution $\boldsymbol{\sigma}_{\mathrm{h}}$, and a capillarity stress tensor $\boldsymbol{\sigma}_{\mathrm{t}}$. The stress tensors are computed as

$$\boldsymbol{\sigma}_{\mathrm{s}} = -\rho\, \mathbf{A}^{\mathsf{T}}\, \partial_{\mathbf{A}} E = -\rho\, c_{\mathrm{s}}^2\, \mathbf{G}\, \operatorname{dev} \mathbf{G}, \tag{2.3}$$

$$\boldsymbol{\sigma}_{\mathrm{h}} = -\rho\, \mathbf{J} \otimes \partial_{\mathbf{J}} E = -\rho\, \alpha_T^2\, \mathbf{J} \otimes \mathbf{J}, \tag{2.4}$$

$$\boldsymbol{\sigma}_{\mathrm{t}} = -\rho\, \mathbf{b} \otimes \partial_{\mathbf{b}} E = -\sigma\, \left(\|\mathbf{b}\|^{-1}\, \mathbf{b} \otimes \mathbf{b} - \|\mathbf{b}\|\, \mathbf{I} \right), \tag{2.5}$$

with $\partial_{\mathbf{A}} E$ being an object with the same shape as $\mathbf{A}$ whose components are the derivatives of E with respect to the corresponding component of $\mathbf{A}$. The same convention applies to $\partial_{\mathbf{J}} E$, $\partial_{\mathbf{b}} E$, and in general.

Note that, due to the choice of strain energy e_{s} given in this chapter, the shear stress tensor $\boldsymbol{\sigma}_{\mathrm{s}}$ is not trace-free, as it would be in linear elasticity or in the Navier–Stokes equations. However, it is easy to prove that for small deformations, both in the linear elastic regime and in the stiff relaxation limit, one does indeed recover a trace-free shear stress tensor $\boldsymbol{\sigma}_{\mathrm{s}}$. Of course nothing prevents other forms of strain energy from being used, such as that given in [242], which yields a cleaner separation of spherical and deviatoric stresses, as well as a sufficient criterion for ensuring the hyperbolicity of the governing equations. With regards to the thermal stress tensor $\boldsymbol{\sigma}_{\mathrm{h}}$, we remark that $\boldsymbol{\sigma}_{\mathrm{h}}$ is vanishingly small in the stiff relaxation limit $\tau_{\mathrm{h}} \to 0$ in which the Fourier theory of heat conduction is recovered.

2.1.2 Remarks on the distortion matrix

It can be shown that the Equation (2.1d), which describes the evolutive kinematics of the distortion matrix $\mathbf{A}$, can equivalently be rewritten as

$$\partial_t\, (\mathbf{A}) + \nabla\, (\mathbf{A}\, \mathbf{u}) + [\nabla \mathbf{A} - (\nabla \mathbf{A})^{\mathsf{T}}]\, \mathbf{u} = \mathbf{Z}, \tag{2.6}$$

highlighting a conservative part $\nabla\, (\mathbf{A}\, \mathbf{u})$ and a nonconservative curl term $[\nabla \mathbf{A} - (\nabla \mathbf{A})^{\mathsf{T}}]\, \mathbf{u}$ which is identically null for all times if null at a given initial time, if the source term $\mathbf{Z}$ vanishes. This corresponds to the limit $\tau \to \infty$, i.e. when the equations coincide with those of the Godunov–Romenski model of Eulerian hyperelasticity and is an expression of Schwarz's rule of second derivatives, reflecting the origin of $\mathbf{A}$ which in the context of elastic solids is defined as the gradient of a mapping between the Eulerian coordinates $\mathbf{x}$ and the Lagrangian coordinates $\mathbf{X}$. In this particular case, the distortion matrix $\mathbf{A}$ is the *inverse* of the *deformation gradient tensor*, commonly denoted in the technical literature as $\mathbf{F} = \partial \mathbf{x} / \partial \mathbf{X}$ or $F_{ij} = \partial x_i / \partial X_j$. Then, since $\mathbf{A}\,\mathbf{F} = \mathbf{I}$, for elastic solids ($\tau \to \infty$) we have $\mathbf{A} = \partial \mathbf{X} / \partial \mathbf{x}$ or $A_{ij} = \partial X_i / \partial x_j$. The metric tensor $\mathbf{G}$ coincides with the inverse of the left Cauchy–Green deformation tensor $\mathbf{B} = \mathbf{F}\,\mathbf{F}^{\mathsf{T}}$ and is sometimes called Piola tensor or Finger tensor. However, since often the left Cauchy–Green tensor $\mathbf{B}$ is called Finger tensor, and occasionally the inverse $\mathbf{C}^{-1}$ of the right Cauchy–Green deformation tensor $\mathbf{C} = \mathbf{F}^{\mathsf{T}}\,\mathbf{F}$ is called *again* Finger tensor, we adopt the name *metric tensor* for referring to $\mathbf{G} = \mathbf{A}^{\mathsf{T}}\,\mathbf{A}$.

Furthermore, (2.1d) can also be expressed as three separate vector equations

$$\partial_t \left(\mathbf{a}_k\right) + \nabla \left(\mathbf{a}_k \cdot \mathbf{u}\right) + \left[\nabla \mathbf{a}_k - \left(\nabla \mathbf{a}_k\right)^{\mathsf{T}}\right] \mathbf{u} = \mathbf{z}_k, \quad k = 1,\ 2,\ 3, \tag{2.7}$$

one for each row $\mathbf{a}_k$ of the distortion matrix $\mathbf{A}$. Equation (2.7) highlights the fact that $\mathbf{A}$ is not strictly speaking a tensor, but rather a triplet of vectors, sometimes called *cobasis* within the solid mechanics community. The cobasis nomenclature can also be carried over to the fluid regime of the equations, as the distortion matrix $\mathbf{A}$ always describes a *local* triad, however for fluids its rows can no longer be interpreted as the gradient of a coordinate system, and therefore the cobasis is non-holonomic. Note that we define, in an abuse of notation, the gradient $\nabla \mathbf{A}$ to be $\nabla \mathbf{A} = \partial \mathbf{A} / \partial \mathbf{x}$, instead of $(\partial \mathbf{A} / \partial \mathbf{x})^{\mathsf{T}}$. Analogously, throughout this manuscript, the divergence operator for second-rank tensors is applied to each row, and it should be remarked however that in this manuscript such an operator is never applied to non-symmetric second-rank tensors.

Moreover, the explicit expression of the cobasis $\mathbf{A}$ as a triplet of vectors, as given in (2.7), helps to clarify the ambiguity in the definition of the gradient of a rank-2 object, and reflects the fact that even in the computational practice, the discretisation of the left hand-side of (2.6), is implemented as a threefold repetition of (2.7).

An important remark is that, bar the addition of relaxation sources, effectively all three equations (for $\mathbf{A}$ (2.1d), $\mathbf{J}$ (2.1e), $\mathbf{b}$ (2.1f)) express evolution laws for geometrical quantities originating as gradients. For this reason, solutions to these equations must satisfy additional compatibility conditions, so called *involution* constraints, enforcing the symmetry of second derivatives. In particular, we will study in detail the important role of the involutive constraint

$$\nabla \times \mathbf{b} = \mathbf{0}, \tag{2.8}$$

which also holds for each row of $\mathbf{A}$ in the elasticity limit and for the thermal impulse vector $\mathbf{J}$ if the corresponding relaxation source $\mathbf{W}$ vanishes or is linear in $\mathbf{J}$. The congruence condition on $\mathbf{A}$ is well known in the solid mechanics community in the form $\nabla_{\mathbf{X}} \times \mathbf{F} = \mathbf{0}$, stating that the Lagrangian curl $\nabla_{\mathbf{X}} \times \mathbf{F}$, applied separately to each row of the deformation gradient tensor $\mathbf{F}$, must vanish in order for the tensor to be compatible with continuous deformations.

Structure of relaxation sources. In this work, the strain relaxation source takes the form

$$\mathbf{Z} = -\frac{3}{\tau}\,(\det \mathbf{A})^{5/3}\;\mathbf{A}\,\operatorname{dev}\mathbf{G}, \tag{2.9}$$

with τ a fixed parameter which can be interpreted as a strain relaxation timescale. Such a specific form is chosen so that in the fluid limit $\tau \to 0$, one recovers constant the compressible Navier–Stokes equations with constant dynamic viscosity [120]. Note that for elastic solids (i.e., if $\mathbf{Z} = \mathbf{0}$) the three cobasis equations (2.7) evolve independently of each other, that is, each row $\mathbf{a}_k$ of $\mathbf{A}$ satisfies an evolution law that does not depend on the other rows. If relaxation is introduced ($\mathbf{Z} \neq \mathbf{0}$), this is no longer true: in fluids and in viscoplastic solids all components of $\mathbf{A}$ depend on all other components due to the coupling introduced by the relaxation source (2.9). It is immediately apparent that, in absence of differential convection effects ($\nabla\mathbf{u} = \mathbf{0}$), the source term $\mathbf{Z}$ relaxes the metric tensor $\mathbf{G}$ towards a spherical one, that is $\operatorname{dev}\mathbf{G} \to \mathbf{0}$, which corresponds to the zero of the strain energy e_s. In general, the structure of the relaxation sources is taken to be $\mathbf{Z} = -\partial_{\mathbf{A}}E/\vartheta_\mathrm{s}$ and $\mathbf{J}/\tau_\mathrm{H} = -\partial_{\mathbf{J}}E/\vartheta_\mathrm{h}$ so that one may guarantee the existence of an entropy balance law

$$\partial_t\,(\rho\,S) + \nabla\cdot\Big(\rho\,S\,\mathbf{u} + \rho\,\alpha_T^2\,\mathbf{J}\Big) = \frac{\rho}{T}\left\{\frac{\operatorname{tr}\left[(\partial_{\mathbf{A}}E)^\intercal\;\partial_{\mathbf{A}}E\right]}{\vartheta_\mathrm{s}} + \frac{\|\partial_{\mathbf{J}}E\|^2}{\vartheta_\mathrm{h}}\right\} \geq 0 \tag{2.10}$$

in which the source on the right hand side is composed of quadratic forms and therefore guaranteed to be non-negative, thus ensuring that the second principle of thermodynamics be satisfied by the governing equations.

With the reported choices of shear energy e_s and heat energy e_h, which yield $\partial_{\mathbf{J}}E = \alpha_T^2\,\mathbf{J}$ and $\partial_{\mathbf{A}}E = c_\mathrm{s}^2\,\mathbf{A}\,\operatorname{dev}\mathbf{G}$, the positive relaxation scaling functions ϑ_s and ϑ_h are taken to be $\vartheta_\mathrm{s} = (\tau/3)\;c_\mathrm{s}^2\;(\det\mathbf{A})^{-5/3}$ and $\vartheta_\mathrm{h} = \tau_\mathrm{H}\,\alpha_T^2$.

With τ and τ_h we denote the fixed relaxation times for strain and heat effects respectively. Together with ρ_0, T_0, α_T, c_s, these are model parameters that can be chosen in order to reproduce a certain effective dynamic viscosity $\mu = \rho_0\,c_\mathrm{s}^2\,\tau/6$ or thermal conductivity $\kappa = \rho_0\,T_0\,\alpha_T^2\,\tau_\mathrm{h}$. For example, in the practical implementation adopted in this work, the relaxation time τ_h is computed as $\tau_\mathrm{h} = \kappa\,\left(\alpha_T^2\,\rho_0\,T_0\right)^{-1}$ so to impose a given value of the thermal conductivity κ.

In fact, in the formalism of the SHTC systems [163, 166, 256, 279], which univocally defines the structure of the model developed in this work, the energy equation (2.1c) is a consequence of all other equations in the system (2.1) and of the entropy equation (2.10) and can be recovered with an appropriate weighted sum of the equations composing (2.1a, 2.1b, 2.1d, 2.1e, 2.1f) and (2.10) to obtain (2.1c) (see [166, 256, 279] for details on the SHTC formalism). However, from a computational standpoint, solving the energy conservation law (2.1c) not only is much easier than solving the entropy balance (2.10), but it can be mandatory if shockwaves are present in the solution. A notable exception to this statement has been provided in [58, 61], where a novel thermodynamically compatible scheme was introduced, by which one can obtain the energy conservation law as a consequence of the entropy equation at the discrete level.

Algebraic constraints. The mass conservation equation (2.1d) can be expressed as a *consequence* of the cobasis equation (2.1d), following the definition $\rho = \rho_0 \det \mathbf{A}$. However, in the computational practice it is preferable not to eliminate the mass conservation equation from the numerical solver for the obvious reason that (2.1a) is a conservation law with clearly defined Rankine–Hugoniot jump conditions. For this reason, in order to preserve the compatibility between the two definitions, of ρ, which can be in principle be different due to discretisation errors, we manually impose the constraint $\det \mathbf{A} = \rho/\rho_0$ as detailed in Chapter 4.

2.2 Two phase flow with capillarity

The starting point for the inclusion of surface tension forces in the unified hyperbolic model of continuum mechanics is the weakly hyperbolic two-phase, single velocity, single pressure model proposed in the paper of Schmidmayer *et al* [286] :

$$\partial_t \left(\alpha_1\, \rho_1\right) + \nabla \cdot \left(\alpha_1\, \rho_1\, \mathbf{u}\right) = 0, \tag{2.11a}$$

$$\partial_t \left(\alpha_2\, \rho_2\right) + \nabla \cdot \left(\alpha_2\, \rho_2\, \mathbf{u}\right) = 0, \tag{2.11b}$$

$$\partial_t \left(\rho\, \mathbf{u}\right) + \nabla \cdot \left(\rho\, \mathbf{u} \otimes \mathbf{u} + p\, \mathbf{I} + \boldsymbol{\Omega}\right) = \mathbf{0}, \tag{2.11c}$$

$$\partial_t \left(\rho\, E\right) + \nabla \cdot \left[\left(\rho\, E + p\right)\, \mathbf{u} + \boldsymbol{\Omega}\, \mathbf{u}\right] = 0, \tag{2.11d}$$

$$\partial_t \left(\alpha_1\right) + \mathbf{u} \cdot \nabla \alpha_1 - K\, \nabla \cdot \mathbf{u} = 0, \tag{2.11e}$$

$$\partial_t \left(\mathbf{b}\right) + \left(\nabla \mathbf{b}\right)\, \mathbf{u} + \left(\nabla \mathbf{u}\right)^{\mathsf{T}} \mathbf{b} = \mathbf{0}, \tag{2.11f}$$

$$\nabla \times \mathbf{b} = \mathbf{0}. \tag{2.11g}$$

The model consists of two mass conservation equations (2.11a) and (2.11b), one for each of the two phases, a single (vector) equation (2.11c) for the conservation of mixture momentum $\rho\, \mathbf{u}$ and one scalar equation (2.11d) for the conservation of the total energy of the mixture $\rho\, E = \rho\, e + \frac{1}{2}\, \rho\, \left\|\mathbf{u}\right\|^2 + \sigma\, \left\|\mathbf{b}\right\|$, which includes a surface energy contribution $\sigma\, \left\|\mathbf{b}\right\|$ to be added to the usual internal and kinetic energy terms. We denote by $\mathbf{u} = \left(u,\ v,\ w\right)^{\mathsf{T}}$ the velocity field and by $\mathbf{b} = \left(b_1,\ b_2,\ b_3\right)^{\mathsf{T}}$ the *interface field* defined in the following paragraphs.

The nonconservative equation (2.11e) can be seen as the limit of the evolution equation of the volume fraction function in the Baer–Nunziato system [10], with stiff pressure relaxation, but also derived from the pressure equilibrium condition for the two phases $p_1 = p_2 = p$ and imposing isentropic behaviour of each phase. In a sharp interface context, the pressure equilibrium hypothesis would seem to be invalid when surface tension is accounted for: one would find a pressure jump located exactly at the interface. However, in our (diffuse interface) framework, pressure equilibrium is valid also *inside* the diffuse interface and simply defines the fine scale structure of the mixture region of the two fluids.

Because the volume fractions α_1 and α_2 are subject to the saturation constraint $\alpha_1 + \alpha_2 = 1$, one equation is sufficient for the description of both. The mixture density is defined as $\rho = \rho_1\, \alpha_1 + \rho_2\, \alpha_2$, where ρ_1 and ρ_2 are the densities of the first and the second phase, respectively. Neglecting surface tension effects, the first five equations of (2.11) are known as Kapila's [201] five equation model, which is the stiff relaxation limit of the seven-equation Baer–Nunziato model [10]. The system is closed if the internal energy $\rho\, e = \alpha_1\, \rho_1\, e_1 + \alpha_2\, \rho_2\, e_2$ is given as a function of the other variables. In this work, we employ the so-called stiffened gas equation of state, in order to establish a biunivocal relation between the pressure of each phase p_1 or p_2 and the corresponding internal energy density $\rho_1\, e_1$ or $\rho_2\, e_2$ as follows:

$$p_1 = (\gamma_1 - 1)\ \rho_1\, e_1 - \gamma_1\, \Pi_1, \qquad p_2 = (\gamma_2 - 1)\ \rho_2\, e_2 - \gamma_2\, \Pi_2, \tag{2.12}$$

with γ_1, γ_2, Π_1, Π_2 given parameters of the equation of state. Due to the pressure equilibrium assumption $p_1 = p_2 = p$, the mixture equation of state then reads

$$p = \frac{\rho\, e\ (\gamma_1 - 1)\, (\gamma_2 - 1) - \alpha_1\, \gamma_1\, \Pi_1\ (\gamma_2 - 1) - \alpha_2\, \gamma_2\, \Pi_2\ (\gamma_1 - 1)}{\alpha_1\ (\gamma_2 - 1) + \alpha_2\ (\gamma_1 - 1)}. \tag{2.13}$$

Furthermore, for this choice of closure relation, we have

$$K = \frac{\alpha_1\, \alpha_2\ \left(\rho_2\, a_2^2 - \rho_1\, a_1^2\right)}{\alpha_1\, \rho_2\, a_2^2 + \alpha_2\, \rho_1\, a_1^2}, \tag{2.14}$$

$$a_1 = \sqrt{\frac{\gamma_1\ (p + \Pi_1)}{\rho_1}}, \qquad a_2 = \sqrt{\frac{\gamma_2\ (p + \Pi_2)}{\rho_2}}. \tag{2.15}$$

For the purpose of capturing the evolution of the interface geometry, a passively-advected scalar quantity c, commonly termed colour function, is introduced; this quantity, similarly to the volume fraction and mass fraction functions, ranges from zero to one and indicates, in a diffused sense, the position of the interface. The colour function follows a simple transport equation given by $\partial_t\,(c) + \mathbf{u} \cdot \nabla c = 0$, and forces due to surface tension are taken into account by means of a tensor $\boldsymbol{\Omega}$ which can be written in terms of the gradient of the colour function and of a constant surface tension coefficient σ as

$$\boldsymbol{\Omega} = \sigma\ \|\nabla c\| \left(\frac{\nabla c \otimes \nabla c}{\|\nabla c\|^2} - \mathbf{I}\right). \tag{2.16}$$

The associated surface energy density is given by $\sigma\ \|\nabla c\|$, meaning that when the colour function c is a Heaviside-type function, the surface energy is a Dirac-delta-type function. Such a conservative formulation [173] of surface tension is well established and essentially equivalent to the very popular Continuum Surface Force (CSF) approach of Brackbill *et al* [52], but since the tensor $\boldsymbol{\Omega}$ depends non-linearly on the derivatives of the state variables, it is difficult to certify the well-posedness of the resulting initial-value problem. Moreover, in order to compute surface tension forces, one would have to reconstruct the colour function data and compute the diffuse interface normal vectors $\hat{\mathbf{n}} = \nabla c / \left\lVert \nabla c \right\rVert$ from this reconstructed information. In order to obtain a first order hyperbolic formulation, a new *interface field* $\mathbf{b} = \nabla c = (b_1,\ b_2,\ b_3)^{\mathsf{T}}$ was introduced in Schmidmayer *et al* [286] , together with a corresponding Galilean invariant evolution equation (2.11f) in the form, in which all components of the interface field $\mathbf{b}$ should be treated as independent state variables.

This procedure, besides allowing to write the governing equations as a system of first order PDEs, completely avoids the computation of local curvature values and interface normal vectors. The surface tension tensor can now be directly computed from the state variables as a nonlinear algebraic function

$$\boldsymbol{\Omega} = \sigma\ \left\lVert \mathbf{b} \right\rVert\ \left(\frac{\mathbf{b} \otimes \mathbf{b}}{\left\lVert \mathbf{b} \right\rVert^2} - \mathbf{I} \right). \tag{2.17}$$

In turn, a new difficulty is introduced by the necessity of properly representing and transporting the interface field $\mathbf{b}$ itself, which can be extremely challenging for a numerical scheme due to the presence of Dirac-delta-like features (noting that $\mathbf{b} = \nabla c$, hence if c approximates a step function, its gradient is an approximation of the delta distribution), requiring very high spatial resolution and low numerical dissipation. Nonetheless, the resolution requirements can be slightly relaxed by initializing the interface field from a smoothed colour function, without observing strong modifications of the pressure jumps across interfaces, as can be noted in Figure 2.1, in which we show the exact solutions for the pressure profiles inside a droplet having a geometrical representation with varying degree of interface smoothing. To clarify, in this work, contrary to the numerical approach adopted by Schmidmayer *et al* in [286], the colour function c itself is *never* used for computing the capillary stress tensor $\boldsymbol{\Omega}$, or anything else, as equation (2.17) replaces equation (2.16). Even more, the colour function does not have any effect on any of the other fields, that is, the evolution equation of the colour function is only coupled passively with the rest of the system and can be removed altogether from the computation.

Note that the capillarity tensor in (2.17) is well behaved also far away from material interfaces, where the $\mathbf{b}$ field is absent, in the sense that its numerator vanishes more quickly than its denominator. In the interest of clarity we specify that, in numerical computations, we handle division by $\left\lVert \mathbf{b} \right\rVert$ by computing $\boldsymbol{\beta} = (\beta_1,\ \beta_2,\ \beta_3)^{\mathsf{T}} = \mathbf{b} / \max(\left\lVert \mathbf{b} \right\rVert,\ \varepsilon)$, with $\varepsilon = 10^{-14}$. One may alternatively compute $\boldsymbol{\beta} = \mathbf{b} / (\left\lVert \mathbf{b} \right\rVert + \varepsilon)$ which would still have negligible effects in regions where the interface field is present, while employing a cheaper floating point addition instead of the max function and acting as a smooth filter rather than a switch. We have employed the former more expensive option, but we are not arguing that it should be preferred to the second one.

Since the new field $\mathbf{b}$ has been defined as the gradient of a scalar function, it must satisfy the constraint

$$\nabla \times \mathbf{b} = \mathbf{0} \tag{2.18}$$

at all times. In fact, equation or (2.11f) automatically preserves the curl-free property of the interface field $\mathbf{b}$ if it is satisfied by the initial condition. To verify this, one can note that the equation for the curl $\boldsymbol{\omega} = \nabla \times \mathbf{b}$ reads [256]

$$\partial_t(\boldsymbol{\omega}) + \nabla \times (\boldsymbol{\omega} \times \mathbf{u}) + \mathbf{u} \nabla \cdot \boldsymbol{\omega} = \mathbf{0} \tag{2.19}$$

and hence, because $\nabla \cdot \boldsymbol{\omega} = \nabla \cdot (\nabla \times \mathbf{b}) = 0$, the curl $\boldsymbol{\omega} = \nabla \times \mathbf{b}$ is zero for all time instants if it was so initially. Despite being consequence of (2.11f), it is well known that, similarly to the divergence-free constraint in the MHD equations, the involution constraint (2.18) represents one of the major difficulties in the numerical discretisation of the evolution equation of $\mathbf{b}$, which will be thoroughly treated and discussed in this Section.

Moreover, it has appeared that the importance of the curl constraint (2.18) is not only crucial for the numerical treatment of system (2.11) but also for its theoretical analysis. Thus, despite the intention of obtaining a first-order *strongly hyperbolic* formulation for surface tension, the authors in [286] achieved this goal only partially. In particular, it is shown in [286] that equations (2.11e)–(2.11f) are only *weakly hyperbolic*, that is, all the eigenvalues of the system are real, but one cannot find a full set of linearly independent eigenvectors. This, in particular, means that system (2.11) is weakly stable [96], i.e. the system (2.11), if linearized in the vicinity of a stationary solution, admits unbounded solutions (some Fourier-Laplace modes have a polynomial growth in time, e.g. see [96]). In fact, the question of hyperbolicity and thus, of well-posedness of the initial value problem for system (2.11) is not a trivial one. Indeed, formally speaking, the curl constraint $\nabla \times \mathbf{b} = \mathbf{0}$ is not an independent PDE but it is a consequence of the evolution equation (2.11f) and thus, formally, it can be excluded from the hyperbolicity analysis of (2.11). Nonetheless, the results of Dafermos [93] suggest that the definition of hyperbolicity (i.e. the conditions for the well-posedness of the initial value problem) of first-order time-evolution equations constrained by stationary involutions has to be revisited. In particular, in Section 2.2.5, we show that the weakly hyperbolic formulation (2.11) can retain strong hyperbolicity in a narrow sense, meaning if the zero-curl constraint is further used to eliminate the redundant degrees of freedom in the interface field $\mathbf{b}$, which also allows to eliminate the unbounded Fourier–Laplace modes of the linearised system. This is intuitively clear from the fact that the $\mathbf{b}$ vector is the gradient of a scalar colour function, and thus all of its components are uniquely determined once the spatial configuration of such a scalar colour function is given. Nevertheless, for the sake of simplicity, throughout this Section we will refer to the model given in [286] as being *weakly hyperbolic.* In the context of Godunov-type finite volume schemes that do not account for the presence of the curl constraint one must rely on the stability properties of the mathematical model and on the accuracy of the numerical schemes to preserve such a constraint. Thus, we remark that in all the numerical computations presented in this work, we always solve all three evolutionary PDEs associated with the three components of the interface field **b**, with no special *numerical* treatment of the curl involution, but only by modifying the underlying PDE system.

Indeed, preliminary results [111] may indicate that in order to discretise the original weakly hyperbolic model [286] in a stable manner, a special *structure-preserving* numerical scheme may be needed, so that one would be able to preserve the curl-free condition *exactly* at the discrete level for all times, similar to the exactly divergence-free schemes developed in the last decades for the Maxwell and MHD equations, see *e.g.* [11–17, 19, 21, 98, 153, 177, 333] and references therein. Much less is known, instead, on exactly curl-preserving schemes. A rather general framework for the construction of structure-preserving schemes (including curl-preserving methods) was developed by Hyman and Shashkov in [192] and Jeltsch and Torrilhon in [198, 323]. Further work on mimetic and structure-preserving finite difference schemes can be found *e.g.* in [66, 217, 231]. For families of compatible finite element methods, the reader is referred to [8, 65, 186, 237, 244, 245, 275]. Therefore, one of the main goals of this Section is to modify the Schmidmayer *et al* [286] model in order to obtain a *strongly hyperbolic* version, with a full set of linearly independent eigenvectors. In the following Sections we will discuss that at least two different ways of achieving such a goal exist. As a final clarification note, rigorously speaking, obtaining a modified model with a complete basis of eigenvectors would yield a so-called *hyperbolic* system, rather than a *strongly hyperbolic* one, however, in this work we use the two interchangeably in order to stress the difference between a hyperbolic system and a weakly hyperbolic one.

2.2.1 Recovering hyperbolicity of the model with Godunov–Powell-type symmetrising terms

Since the original weakly hyperbolic form of the model is not suitable for the solution with explicit Godunov-type schemes, and motivated by the theory of Symmetric Hyperbolic and Thermodynamically Compatible (SHTC) equations [163, 166, 256, 279], we introduce some formal modifications to system (2.11) in such a way that the new system can be written in the symmetric Godunov form and the eigenvector that was reported missing in the paper of Schmidmayer *et al* [286] is recovered. We note that the resulting symmetrizable system is not symmetric hyperbolic because the energy potential $\rho\, E$ is not convex as required by the SHTC system theory [256]. Nevertheless, in some cases symmetric hyperbolicity can be obtained even for systems with non-convex potentials, e.g. see [143]. Note that in [286], this issue was circumvented by discretising the colour function equation and computing the interface field as its gradient, instead of directly solving the vector equation for the interface field $\mathbf{b}$. It is necessary to emphasise that the applied modifications are valid on smooth solutions, while the validity of the obtained non-conservative hyperbolic model on discontinuities requires further investigation.

The modifications are applied by introducing in the momentum and energy equations two nonconservative terms that, at the continuum level at least, are identically null, thanks to the curl constraint (2.18), which is nothing else but Schwarz's rule of symmetry of second order derivatives

$$\nabla\mathbf{b} = (\nabla\mathbf{b})^{\mathsf{T}}. \tag{2.20}$$

In the paper of Schmidmayer *et al* [286] this property was used in the hyperbolicity study, following Ndanou *et al* [242], to rearrange the evolution equation for the gradient of the colour function $\mathbf{b}$ in two equivalent versions. One is the Galilean-invariant equation given in (2.11f), the other one is the following formally conservative but non-Galilean-invariant

$$\partial_t\left(\mathbf{b}\right) + \nabla\cdot\left[\left(\mathbf{b}\cdot\mathbf{u}\right)\mathbf{I}\right] = \mathbf{0}. \tag{2.21}$$

In order to conduct our mathematical and numerical study of system (2.11), we make use of the same compatibility condition and rewrite the fully nonconservative equation (2.11f) in a semi-conservative form

$$\partial_t(\mathbf{b}) + \nabla\cdot\left[\left(\mathbf{u}\cdot\mathbf{b}\right)\mathbf{I}\right] + \left[\nabla\mathbf{b} - (\nabla\mathbf{b})^{\mathsf{T}}\right]\,\mathbf{u} = \mathbf{0}. \tag{2.22}$$

Furthermore, we add a similar nonconservative contribution as the last term on the left-hand side of (2.22) to the momentum equation, which then becomes

$$\partial_t\left(\rho\,\mathbf{u}\right) + \nabla\cdot\left(\rho\,\mathbf{u}\otimes\mathbf{u} + p\,\mathbf{I} + \mathbf{\Omega}\right) + \left[(\nabla\mathbf{b})^{\mathsf{T}} - \nabla\mathbf{b}\right]\,\sigma\,\frac{\mathbf{b}}{\|\mathbf{b}\|} = \mathbf{0} \tag{2.23}$$

and to the energy equation, formally accounting for the work due to the newly introduced forces

$$\partial_t\left(\rho\,E\right) + \nabla\cdot\left[\left(\rho\,E + p\right)\,\mathbf{u} + \boldsymbol{\Omega}\cdot\mathbf{u}\right] + \left[(\nabla\mathbf{b})^{\mathsf{T}} - \nabla\mathbf{b}\right]\,\sigma\,\frac{\mathbf{b}}{\|\mathbf{b}\|}\cdot\mathbf{u} = 0. \tag{2.24}$$

The modified model with Godunov–Powell-type symmetrising nonconservative products is then written compactly as

$$\partial_t\begin{pmatrix}\alpha_1\,\rho_1\\ \alpha_2\,\rho_2\\ \rho\,\mathbf{u}\\ \rho\,E\\ \alpha_1\\ \mathbf{b}\end{pmatrix} + \nabla\cdot\begin{pmatrix}\alpha_1\,\rho_1\,\mathbf{u}\\ \alpha_2\,\rho_2\,\mathbf{u}\\ \rho\,\mathbf{u}\otimes\mathbf{u} + p\,\mathbf{I} + \boldsymbol{\Omega}\\ \left(\rho\,E + p\right)\,\mathbf{u} + \boldsymbol{\Omega}\,\mathbf{u}\\ 0\\ \left(\mathbf{u}\cdot\mathbf{b}\right)\mathbf{I}\end{pmatrix} + \begin{pmatrix}0\\ 0\\ \left[(\nabla\mathbf{b})^{\mathsf{T}} - \nabla\mathbf{b}\right]\,\sigma\,\mathbf{b}/\|\mathbf{b}\|\\ \left[(\nabla\mathbf{b})^{\mathsf{T}} - \nabla\mathbf{b}\right]\,\sigma\,\mathbf{b}/\|\mathbf{b}\|\cdot\mathbf{u}\\ \mathbf{u}\cdot\nabla\alpha_1 - K\,\nabla\cdot\mathbf{u}\\ \left[\nabla\mathbf{b} - (\nabla\mathbf{b})^{\mathsf{T}}\right]\,\mathbf{u}\end{pmatrix} = \mathbf{0}, \tag{2.25}$$

As a result of the fact that all the new nonconservative terms in (2.22), (2.23), and (2.24) evaluate to zero if the compatibility condition (2.20) is fulfilled, the formulation (2.25) is, at least for smooth solutions on the continuum level, entirely equivalent to the model (2.11). Yet, the important difference is that now a full set of linearly independent eigenvectors (given in the following subsection) can be obtained for this new form of the system, and thus one can prove the strong hyperbolicity of the model. We will then discuss in Section 6.2 the different behaviour that the two formulations exhibit at the discrete level.

It is necessary to emphasise that the transformations described above do not ruin the thermodynamic compatibility of the governing PDEs, that is the over-determined system (2.25), together with an appropriate entropy equation, still forms a compatible system, *e.g.* see [256].

Finally, we explicitly write the SHTC form of the momentum conservation law (2.11c)

$$\partial_t\left(\rho\,\mathbf{u}\right) + \nabla\cdot\Big(\rho\,\mathbf{u}\otimes\mathbf{u} + \rho^2\partial_\rho E\,\mathbf{I} + \rho\,\mathbf{b}\otimes\partial_{\mathbf{b}}E\Big) = \mathbf{0}, \tag{2.26}$$

where $\rho^2\,\partial_\rho E$ is the total thermodynamic pressure which for the given energy potential $E = e + \|\mathbf{u}\|^2/2 + \sigma\,\|\mathbf{b}\|/\rho$ reads

$$\rho^2\,\partial_\rho E = \rho^2\,\partial_\rho E = \rho^2\,\partial_\rho e - \sigma\,\|\mathbf{b}\|\,, \tag{2.27}$$

with $\rho^2\partial_\rho E = p$ being the mixture pressure without the capillarity pressure and given by (2.13), while the non-isotropic stress $\rho\,\mathbf{b}\otimes\partial_{\mathbf{b}}E$ reads

$$\rho\,\mathbf{b}\otimes\partial_{\mathbf{b}}E = \sigma\,\frac{\mathbf{b}\otimes\mathbf{b}}{\|\mathbf{b}\|}. \tag{2.28}$$

In (2.26), one can recognize the important role of the total energy potential E as being a generating potential for all the SHTC systems [256]. In particular, the standard way of symmetrizing the SHTC systems with the structure of (2.11) is to add the term $\left[(\nabla \mathbf{b})^{\mathsf{T}} - \nabla \mathbf{b}\right] \, \rho \, \partial_{\mathbf{b}} E = \mathbf{0}$ to the momentum equation, that is, exactly the last term on the left hand-side in (2.23). After that, the symmetrizable form of the momentum equation (2.26) reads

$$\partial_t \left(\rho \, \mathbf{u}\right) + \nabla \cdot \left(\rho \, \mathbf{u} \otimes \mathbf{u} + \rho^2 \, \partial_\rho E \, \mathbf{I}\right) + \mathbf{b} \, \nabla \cdot \left(\rho \, \partial_{\mathbf{b}} E\right) - \left(\nabla \mathbf{b}\right) \rho \, \partial_{\mathbf{b}} E = \mathbf{0}, \tag{2.29}$$

In the SHTC system theory, the total energy conservation law

$$\partial_t \left(\rho \, E\right) + \nabla \cdot \left[\rho \, E \, \mathbf{u} + \left(\rho^2 \, \partial_\rho E \, \mathbf{I} + \rho \, \mathbf{b} \otimes \partial_{\mathbf{b}} E\right) \mathbf{u}\right] = 0. \tag{2.30}$$

is not an independent equation, but rather it is a consequence of the other equations in the system (see [256] for details).

2.2.2 Eigenstructure of the strongly hyperbolic Godunov–Powell-type model

We continue our description of the new reformulation of the model of Gavrilyuk and Schmidmayer model for two-phase flow with surface tension by providing explicit expressions for the eigenvalues and the complete set of eigenvectors of the system, hence showing hyperbolicity. By defining a vector of conserved variables $\mathbf{Q}$ and one of primitive variables $\mathbf{V}$ as

$$\mathbf{Q} = \left(\alpha_1 \, \rho_1, \ \alpha_2 \, \rho_2, \ \rho \, \mathbf{u}^{\mathsf{T}}, \ \rho \, E, \ \alpha_1, \ \mathbf{b}^{\mathsf{T}}\right)^{\mathsf{T}}, \quad \mathbf{V} = \left(\rho_1, \ \rho_2, \ \mathbf{u}^{\mathsf{T}}, \ p, \ \alpha_1, \ \mathbf{b}^{\mathsf{T}}\right)^{\mathsf{T}} \tag{2.31}$$

the governing PDE system (2.25) can be written in compact matrix-vector notation as

$$\partial_t \mathbf{Q} + \nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q}) \, \nabla \mathbf{Q} = \mathbf{0}, \tag{2.32}$$

where $\mathbf{F}(\mathbf{Q})$ is a nonlinear flux tensor and $\mathbf{B}(\mathbf{Q}) \, \nabla \mathbf{Q} = \mathbf{B}_i \, \partial_{x_i} \mathbf{Q}$, with $i = 1, \ 2, \ 3$, accounts for the non-conservative products. The coefficients of the matrices $\mathbf{B}_i$ are collected in $\mathbf{B} = (\mathbf{B}_1, \ \mathbf{B}_2, \ \mathbf{B}_3)$, and can be easily deduced from the governing PDE system. The quasi-linear form of the PDE in terms of the conservative variables $\mathbf{Q}$ reads

$$\partial_t \mathbf{Q} + \mathbf{A}(\mathbf{Q}) \, \nabla \mathbf{Q} = \mathbf{0}, \tag{2.33}$$

with

$$\mathbf{A}(\mathbf{Q}) = (\mathbf{A}_1, \ \mathbf{A}_2, \ \mathbf{A}_3) = \frac{\partial \mathbf{F}}{\partial \mathbf{Q}} + \mathbf{B}(\mathbf{Q}). \tag{2.34}$$

In terms of the vector of primitive variables $\mathbf{V}$ it can be rewritten as

$$\partial_t \mathbf{V} + \mathbf{C}(\mathbf{V})\,\nabla\mathbf{V} = \mathbf{0}, \tag{2.35}$$

with

$$\mathbf{C}(\mathbf{V}) = (\mathbf{C}_1,\ \mathbf{C}_2,\ \mathbf{C}_3), \qquad \mathbf{C}_i = \frac{\partial \mathbf{V}}{\partial \mathbf{Q}}\,\mathbf{A}_i\,\frac{\partial \mathbf{Q}}{\partial \mathbf{V}}. \tag{2.36}$$

Due to the rotational invariance of system (2.25), in order to compute its eigenstructure, and thus assess its hyperbolicity, it will be sufficient to project the equations along a generic x direction specified by a unit vector $\hat{\mathbf{e}}_x$, so that the matrix of coefficients appearing in (2.35) has a projection $\mathbf{C}_1 = \mathbf{C}\cdot\hat{\mathbf{e}}_x$ which reads

$$\mathbf{C}_1 = \begin{pmatrix} u & 0 & q_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & u & q_2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & u & 0 & 0 & \rho^{-1} & 0 & \sigma_r\,\beta_1\,(1-\beta_1^2) & \sigma_r\,\beta_1\,\beta_2\,\beta_1 & \sigma_r\,\beta_1\,\beta_3\,\beta_1 \\ 0 & 0 & 0 & u & 0 & 0 & 0 & \sigma_r\,\beta_2\,(1-\beta_1^2) & \sigma_r\,\beta_1\,\beta_2\,\beta_2 & \sigma_r\,\beta_1\,\beta_3\,\beta_2 \\ 0 & 0 & 0 & 0 & u & 0 & 0 & \sigma_r\,\beta_3\,(1-\beta_1^2) & \sigma_r\,\beta_1\,\beta_2\,\beta_3 & \sigma_r\,\beta_1\,\beta_3\,\beta_3 \\ 0 & 0 & \rho\,a^2 & 0 & 0 & u & 0 & 0 & 0 & 0 \\ 0 & 0 & -K & 0 & 0 & 0 & u & 0 & 0 & 0 \\ 0 & 0 & b_1 & b_2 & b_3 & 0 & 0 & u & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & u \end{pmatrix}, \tag{2.37}$$

with $q_1 = (\alpha_1 + K)\,\rho_1/\alpha_1$, $q_2 = (\alpha_2 - K)\,\rho_2/\alpha_2$, and $\sigma_r = \sigma/\rho$. For $\mathbf{C}_1$ we computed the following eigenvalues

$$\boldsymbol{\lambda} = \begin{pmatrix} u \\ u \\ u \\ u \\ u \\ u \\ u - \sqrt{k_1 - k_3} \\ u + \sqrt{k_1 - k_3} \\ u - \sqrt{k_1 + k_3} \\ u + \sqrt{k_1 + k_3} \end{pmatrix}, \quad \text{with} \quad \begin{cases} k_1 = \dfrac{a^2 + a_\sigma^2}{2}, \\ k_2 = \dfrac{a^2 - a_\sigma^2}{2}, \\ k_3 = \sqrt{k_2^2 + \beta_1^2\,a^2\,a_\sigma^2}, \\ a_\sigma^2 = \dfrac{\sigma}{\rho}\ \|\mathbf{b}\|\ (1-\beta_1^2), \\ \beta_1 = \dfrac{b_1}{\|\mathbf{b}\|}, \quad \beta_2 = \dfrac{b_2}{\|\mathbf{b}\|}, \quad \beta_3 = \dfrac{b_3}{\|\mathbf{b}\|}, \end{cases} \tag{2.38}$$

and with a being the Wood [329] speed of sound for the mixture defined by

$$a = \sqrt{\frac{\rho_1\, a_1^2\, \rho_2\, a_2^2}{\rho\, \left(\alpha_1\, \rho_2\, a_2^2 + \alpha_2\, \rho_1\, a_1^2\right)}}. \tag{2.39}$$

The model includes six contact waves moving with the fluid velocity u, and four mixed capillarity/pressure waves. The first six linearly independent right eigenvectors of the $\mathbf{C}_1$ matrix are

$$\mathbf{R}_{1-6} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\beta_3 & 0 & 0 \\ 0 & 0 & 0 & \beta_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ \beta_1\,\beta_3 & \beta_1\,\beta_2 & 0 & 0 & 0 & 0 \\ 0 & 1-\beta_1^2 & 0 & 0 & 0 & 0 \\ 1-\beta_1^2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{2.40}$$

and the last four are

$$\mathbf{R}_{7-10} = \begin{pmatrix} -k_{\rho_1}\,\beta_1 & -k_{\rho_1}\,\beta_1 & -k_{\rho_1} & -k_{\rho_1} \\ k_{\rho_2}\,\beta_1 & k_{\rho_2}\,\beta_1 & k_{\rho_2} & k_{\rho_2} \\ k_5\,\beta_1 & -k_5\,\beta_1 & k_4 & -k_4 \\ -k_7\,\beta_2 & k_7\,\beta_2 & -k_6\,\beta_2/\beta_1 & k_6\,\beta_2/\beta_1 \\ -k_7\,\beta_3 & k_7\,\beta_3 & -k_6\,\beta_3/\beta_1 & k_6\,\beta_3/\beta_1 \\ -\rho\, a^2\,\beta_1 & -\rho\, a^2\,\beta_1 & -\rho\, a^2 & -\rho\, a^2 \\ K\,\beta_1 & K\,\beta_1 & K & K \\ k_5\,k_7\,k_{b_1}\,\beta_1 & k_5\,k_7\,k_{b_1}\,\beta_1 & k_4\,k_6\,k_{b_1} & k_4\,k_6\,k_{b_1} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}, \tag{2.41}$$

with

$$\begin{aligned} &k_4 = \sqrt{k_1 + k_3}, && k_5 = \sqrt{k_1 - k_3}, && k_6 = (k_2 - k_3)/k_4, \\ &k_7 = (k_2 + k_3)/k_5, && k_{\rho_1} = (K + \alpha_1)\,\rho_1/\alpha_1, && k_{\rho_2} = (K - \alpha_2)\,\rho_2/\alpha_2, \\ &k_{b_1} = \|\mathbf{b}\|\,/(\beta_1\, a_\sigma^2). \end{aligned} \tag{2.42}$$

We can then conclude that, on smooth solutions, the hyperbolicity of the surface tension model forwarded in [29, 286] can be restored by including the proposed Godunov–Powell symmetrising nonconservative products. Also, in regions where the $\mathbf{b}$ field is absent, the last two columns of (2.41) reduce to the outer eigenvectors of the Kapila system, while the seventh and eighth collapse onto the inner transport waves.

2.2.3 Hyperbolic curl cleaning with the generalized Lagrangian multiplier (GLM) approach

The modified PDE system discussed in the previous sections, which allows to restore strong hyperbolicity compared to the original model of Schmidmayer *et al* [286], very closely follows the ideas of Godunov [164] and Powell *et al* [265, 266] concerning the symmetrisation and the numerical treatment of the divergence-free condition of the magnetic field in the MHD system, respectively.

An alternative and very successful numerical treatment of the divergence-free condition of the magnetic field for the Maxwell and MHD equations is the so-called generalized Lagrangian multiplier (GLM) approach of Munz *et al* [94, 240], which consists in a *hyperbolic* divergence *cleaning* achieved by adding a new auxiliary scalar field to the PDE system, whose role is to transport divergence errors out of the computational domain via acoustic-type waves, so that they cannot accumulate locally. In the following, we adapt the GLM approach to the system (2.11) with the curl involution $\nabla \times \mathbf{b} = \mathbf{0}$. The augmented GLM version of the system reads

$$\partial_t\left(\alpha_1\,\rho_1\right) + \nabla\cdot\left(\alpha_1\,\rho_1\,\mathbf{u}\right) = 0, \tag{2.43a}$$

$$\partial_t\left(\alpha_2\,\rho_2\right) + \nabla\cdot\left(\alpha_2\,\rho_2\,\mathbf{u}\right) = 0, \tag{2.43b}$$

$$\partial_t\left(\rho\,\mathbf{u}\right) + \nabla\cdot\left(\rho\,\mathbf{u}\otimes\mathbf{u} + p\,\mathbf{I} + \boldsymbol{\Omega}\right) = \mathbf{0}, \tag{2.43c}$$

$$\partial_t\left(\rho\,E\right) + \nabla\cdot\left[\left(\rho\,E + p\right)\,\mathbf{u} + \boldsymbol{\Omega}\,\mathbf{u}\right] = 0, \tag{2.43d}$$

$$\partial_t\left(\alpha_1\right) + \mathbf{u}\cdot\nabla\alpha_1 - K\,\nabla\cdot\mathbf{u} = 0, \tag{2.43e}$$

$$\partial_t\left(\mathbf{b}\right) + \nabla\cdot\left[\left(\mathbf{u}\cdot\mathbf{b}\right)\,\mathbf{I}\right] + \left[\nabla\mathbf{b} - \left(\nabla\mathbf{b}\right)^{\mathsf{T}}\right]\,\mathbf{u} + c_h\,\nabla\times\boldsymbol{\psi} = \mathbf{0}, \tag{2.43f}$$

$$\partial_t\left(\boldsymbol{\psi}\right) + \left(\nabla\boldsymbol{\psi}\right)\mathbf{u} - c_h\,\nabla\times\mathbf{b} = -\kappa\,\boldsymbol{\psi}, \tag{2.43g}$$

where c_h is the artificial wave speed associated with the hyperbolic curl cleaning process and κ is a small damping parameter, which in the present work is always set as $\kappa = 0$. For a similar approach applied to a first order hyperbolic reduction of the Einstein field equations, see [113]. Note the curl-curl structure in the equations for $\mathbf{b}$ and the cleaning field $\boldsymbol{\psi}$, which have a Maxwell-type form, i.e. in the augmented GLM curl cleaning system, the constraint violations in the vector field $\mathbf{b}$ are transported away via electromagnetic-type waves. It is easy to see that in the limit $c_h \to \infty$ one obtains $\nabla \times \mathbf{b} \to \mathbf{0}$. Due to the presence of the transport term $(\nabla \boldsymbol{\psi})\,\mathbf{u}$ in the evolution equation (2.43g), which is needed in order to have a Galilean invariant system, the cleaning vector field $\boldsymbol{\psi}$, unlike in [113], does not obey an additional linear divergence-free involution, and thus we chose not to enforce any additional constraints on the cleaning field itself.

In our implementation, we exploited the linear nature of the curl cleaning terms $c_h\,\nabla \times \boldsymbol{\psi}$ and $-c_h\,\nabla \times \mathbf{b}$ to achieve a conservative form, recasting them as $c_h\,\nabla \times \boldsymbol{\psi} = \nabla \cdot (-c_h\,\boldsymbol{\varepsilon} \cdot \boldsymbol{\psi})$ and $-c_h\,\nabla \times \mathbf{b} = \nabla \cdot (c_h\,\boldsymbol{\varepsilon} \cdot \mathbf{b})$, where $\boldsymbol{\varepsilon} = \varepsilon_{ijk}$ is the Levi–Civita tensor and the divergence of a non-symmetric second rank tensor such as $\boldsymbol{\varepsilon} \cdot \boldsymbol{\psi}$ or $\boldsymbol{\varepsilon} \cdot \mathbf{b}$ is taken to be the column vector that has for components the divergence of each column of the tensor. The term $(\nabla \boldsymbol{\psi})\,\mathbf{u}$ is instead discretised as a genuinely nonconservative product.

Note that, similar to [95], in order to account for the effects of curl-cleaning on the total energy balance, one should in principle replace the energy conservation equation (2.43d) with

$$\partial_t\,(\rho\,E) + \nabla \cdot \left[(\rho\,E + p)\;\mathbf{u} + \boldsymbol{\Omega}\,\mathbf{u}\right] + \sigma\,c_h\,\frac{\mathbf{b} \cdot \nabla \times \boldsymbol{\psi}}{\lVert\mathbf{b}\rVert} = 0. \tag{2.44}$$

Nonetheless, except for those of Section 6.2.5, all computations shown in this work are carried out retaining the formally conservative equation (2.43d). In preliminary tests, we found negligible differences between the results from the energy-consistent equation (2.44) and from the formally conservative system which neglects the correction given in Eq. (2.44), and the basic properties of the two systems are the same (namely both systems are hyperbolic, have the same eigenvalues, and a full set of eigenvectors can be found in both cases). Likewise, formulations including the Godunov–Powell nonconservative products, in combination with the GLM curl cleaning equations have been tested and yielded results that are comparable with those obtained with GLM curl cleaning alone. Furthermore, we also tested the equivalence at the discrete level of the interface field equation in its original fully nonconservative form (2.11f) with its conservative (but not Galilean-invariant) discretisation according to Eq. (2.11f). The findings obtained from these tests are presented in Section 6.2.5.

2.2.4 Eigenstructure of the augmented GLM model

Hyperbolicity of the augmented GLM curl cleaning system (2.43) can be shown by repeating the procedure carried out in Section 2.2.2 to compute explicitly a set of thirteen eigenvalues

$$\begin{aligned}\boldsymbol{\lambda} = \Big(u,\ u,\ u,\ u,\ u,\ u-\sqrt{k_1-k_3},\ u+\sqrt{k_1-k_3},\ u-\sqrt{k_1+k_3},\\ u+\sqrt{k_1+k_3},\ u-c_h,\ u-c_h,\ u+c_h,\ u+c_h\Big)\,,\end{aligned} \tag{2.45}$$

with

$$k_1 = \frac{a^2+a_\sigma^2}{2}, \qquad k_2 = \frac{a^2-a_\sigma^2}{2}, \qquad k_3 = \sqrt{k_2^2+\beta_1^2\,a^2\,a_\sigma^2}, \tag{2.46}$$

and

$$\beta_1 = \frac{b_1}{\|\mathbf{b}\|}, \quad \beta_2 = \frac{b_2}{\|\mathbf{b}\|}, \quad \beta_3 = \frac{b_3}{\|\mathbf{b}\|}, \qquad a_\sigma^2 = \frac{\sigma}{\rho}\,\|\mathbf{b}\|\,(1-\beta_1^2). \tag{2.47}$$

As a notational convenience for writing the set of thirteen linearly independent right eigenvectors, we introduce the additional shortand auxiliary parameters

$$\begin{aligned}&k_4 = \sqrt{k_1+k_3}, \quad k_5 = \sqrt{k_1-k_3}, \quad k_6 = (k_2-k_3)/k_4, \quad k_7 = (k_2+k_3)/k_5,\\ &k_8 = \left(k_2+k_3+\beta_1^2\,a_\sigma^2\right)/k_4, \quad k_9 = \left(k_2-k_3+\beta_1^2\,a_\sigma^2\right)/k_5,\\ &k_{10} = \frac{\beta_1^2\,a^2+c_h^2\,\left\{1+(\gamma_1-1)\,(\gamma_2-1)\,[\alpha_2\,(\gamma_1-1)+\alpha_1\,(\gamma_2-1)]^{-1}\right\}}{c_h^2\,(a_\sigma^2-c_h^2)+a^2\,\left[c_h^2-a_\sigma^2\,(1-\beta_1^2)\right]},\end{aligned} \tag{2.48}$$

and

$$\begin{aligned}&k_{\rho_1} = (K+\alpha_1)\,\rho_1/\alpha_1, \quad k_{\rho_2} = (K-\alpha_2)\,\rho_2/\alpha_2, \quad k_{b_1} = \|\mathbf{b}\|\,/a_\sigma^2,\\ &h_1 = \beta_1^2 - a_h^2\,k_{10} + (1-\beta_1^2)\,a_\sigma^2\,k_{10}, \quad h_2 = 1+\beta_2^2\,(a_\sigma^2\,k_{10}-1),\\ &h_3 = 1+\beta_3^2\,(a_\sigma^2\,k_{10}-1), \quad h_4 = 1-a_\sigma^2\,k_{10}, \quad h_5 = 1+a_h^2\,k_{10}-a_\sigma^2\,k_{10},\\ &h_6 = \beta_1\,a_h^2\,k_{10}.\end{aligned} \tag{2.49}$$

The first nine eigenvectors of the augmented GLM curl-cleaning model, associated with the transport and acoustic/capillarity waves, are

$$\mathbf{R}_{1-9} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & k_9\,k_{\rho_1} & k_9\,k_{\rho_1} & k_8\,k_{\rho_1} & k_8\,k_{\rho_1} \\ 0 & 0 & 0 & 1 & 0 & -k_9\,k_{\rho_2} & -k_9\,k_{\rho_2} & -k_8\,k_{\rho_2} & -k_8\,k_{\rho_2} \\ 0 & 0 & 0 & 0 & 0 & -k_9\,k_5 & k_9\,k_5 & -k_8\,k_4 & k_8\,k_4 \\ 0 & 0 & -b_3 & 0 & 0 & -a_\sigma^2\,\beta_1\,\beta_2 & a_\sigma^2\,\beta_1\,\beta_2 & -a_\sigma^2\,\beta_1\,\beta_2 & a_\sigma^2\,\beta_1\,\beta_2 \\ 0 & 0 & b_2 & 0 & 0 & -a_\sigma^2\,\beta_1\,\beta_3 & a_\sigma^2\,\beta_1\,\beta_3 & -a_\sigma^2\,\beta_1\,\beta_3 & a_\sigma^2\,\beta_1\,\beta_3 \\ 0 & 0 & 0 & 0 & 0 & k_9\,\rho\,a^2 & k_9\,\rho\,a^2 & k_8\,\rho\,a^2 & k_8\,\rho\,a^2 \\ 0 & 1 & 0 & 0 & 0 & -k_9\,K & -k_9\,K & -k_8\,K & -k_8\,K \\ 0 & 0 & 0 & 0 & 0 & k_5\,\beta_1\,\|\mathbf{b}\| & k_5\,\beta_1\,\|\mathbf{b}\| & k_4\,\beta_1\,\|\mathbf{b}\| & k_4\,\beta_1\,\|\mathbf{b}\| \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \tag{2.50}$$

while the remaining eigenvectors, corresponding to the cleaning waves, are

$$\mathbf{R}_{10-13} = \begin{pmatrix} h_1\,k_{\rho_1} & h_1\,k_{\rho_1} & h_1\,k_{\rho_1} & h_1\,k_{\rho_1} \\ h_1\,k_{\rho_2} & h_1\,k_{\rho_2} & h_1\,k_{\rho_2} & h_1\,k_{\rho_2} \\ h_1\,c_h & h_1\,c_h & h_1\,c_h & h_1\,c_h \\ -h_2\,c_h\,\beta_1/\beta_2 & -h_4\,c_h\,\beta_1\,\beta_2 & -h_2\,c_h\,\beta_1/\beta_2 & -h_4\,c_h\,\beta_1\,\beta_2 \\ h_4\,c_h\,\beta_1\,\beta_3 & h_3\,c_h\,\beta_1/\beta_3 & h_4\,c_h\,\beta_1\,\beta_3 & h_3\,c_h\,\beta_1/\beta_3 \\ h_5\,\rho\,c_h^2 & -h_5\,\rho\,c_h^2 & -h_5\,\rho\,c_h^2 & h_5\,\rho\,c_h^2 \\ h_1\,K & -h_1\,K & -h_1\,K & h_1\,K \\ h_6\,\|\mathbf{b}\| & -h_6\,\|\mathbf{b}\| & -h_6\,\|\mathbf{b}\| & h_6\,\|\mathbf{b}\| \\ \rho\,c_h^2/(\beta_2\,\sigma) & 0 & -\rho\,c_h^2/(\beta_2\,\sigma) & 0 \\ 0 & -\rho\,c_h^2/(\beta_3\,\sigma) & 0 & \rho\,c_h^2/(\beta_3\,\sigma) \\ 0 & 0 & 0 & 0 \\ 0 & \rho\,c_h^2/(\beta_3\,\sigma) & 0 & \rho\,c_h^2/(\beta_3\,\sigma) \\ \rho\,c_h^2/(\beta_2\,\sigma) & 0 & \rho\,c_h^2/(\beta_2\,\sigma) & 0 \end{pmatrix}. \tag{2.51}$$

The wave structure includes five transport fields (contact waves), four *cleaning waves* with eigenvalues $\lambda_{c_h} = u \pm c_h$, and four waves of mixed capillary/acoustic nature with eigenvalues $\lambda_{p\,\sigma} = u \pm \sqrt{k_1 \pm k_3}$, which are the same obtained from the previous variants of the mathematical model. We can therefore conclude that the augmented GLM system (2.43) is strongly hyperbolic. However, its major advantage over the Godunov–Powell-type system is that the GLM system is conservative not only for mass, but also for momentum and energy, while the Godunov–Powell system is not, at least when standard general-purpose schemes are used that do not satisfy the curl involution constraint exactly at the discrete level.

2.2.5 Hyperbolicity of the constrained system

From the numerical viewpoint, hyperbolicity of system (2.11) is important as it is also equivalent to the strong stability of the linearized system that is, its Fourier–Laplace modes are uniformly bounded [96]. However, (2.11) is not a system of pure time-dependent PDEs as it is required by the standard definition of hyperbolicity: instead, its solution also has to satisfy the stationary constraint (2.18). That is, (2.11) and (2.18) constitute an overdetermined system of stationary and time-dependent PDEs. Thus, as suggested by Dafermos [93] the notion of hyperbolicity for such systems might have to be revisited.

In fact, instead of recovering strong hyperbolicity of the original weakly hyperbolic model (2.11) by adding the symmetrizing terms as in the Godunov–Powell-type system (2.25) or enlarging the original equations with GLM curl cleaning (2.43), hyperbolicity of (2.11) can be restored in a special sense as suggested in [93]. However, the results of [93] can not be entirely transferred to our case because they were obtained for systems of conservation laws while not all equations of system (2.11) are in a conservative form and thus, further research is required. Nevertheless, the ideas from [93] give a hint in what sense the strong hyperbolicity of system (2.11) can be recovered and what type of computational schemes can be used in order to obtain stable numerical solution which also satisfies the involution constraint $\nabla \times \mathbf{b} = \mathbf{0}$ at the discrete level.

Let us write system (2.11) and the constraint (2.18) in quasi-linear form as

$$\partial_t \mathbf{V} + \mathbf{C}\,\nabla\mathbf{V} = \mathbf{0}, \tag{2.52}$$

$$\nabla \times \mathbf{b} = \mathbf{D}\,\nabla\mathbf{V} = \mathbf{0}, \tag{2.53}$$

where $\mathbf{C} = (\mathbf{C}_1,\ \mathbf{C}_2,\ \mathbf{C}_3)$ and $\mathbf{D} = (\mathbf{D}_1,\ \mathbf{D}_2,\ \mathbf{D}_3)$ are the collection of 10×10 matrices $\mathbf{C}_i$ and $\mathbf{D}_i$, and $\mathbf{V}$ is the vector of primitive variables $\mathbf{V} = (\rho_1,\ \rho_2,\ \mathbf{u}^\mathsf{T},\ p,\ \alpha_1,\ \mathbf{b}^\mathsf{T})^\mathsf{T}$. Then, given a generic unit vector $\hat{\mathbf{n}} = (n_1,\ n_2,\ n_3)^\mathsf{T}$, we can write a projection of the constraint matrix $\mathbf{D}$ in the form

$$\mathbf{M}_{\mathbf{D}}(\hat{\mathbf{n}}) = \mathbf{D}\cdot\hat{\mathbf{n}} = \begin{pmatrix} 0 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & \dots & 0 & 0 & -n_3 & n_2 \\ 0 & \dots & 0 & n_3 & 0 & -n_1 \\ 0 & \dots & 0 & -n_2 & n_1 & 0 \end{pmatrix}. \tag{2.54}$$

The standard definition of hyperbolicity of system (2.52) is that for any unit vector $\hat{\mathbf{n}}$, the matrix $\mathbf{M}_{\mathbf{C}}(\hat{\mathbf{n}}) = \mathbf{C}\cdot\hat{\mathbf{n}}$ has only real eigenvalues and is diagonalizable. In particular, this can be used to show that small perturbations to the solution do not grow locally in time (see e.g. [96]). On the other hand, for a quasi-linear system with involution constraints, like (2.52)–(2.53), Dafermos demonstrated in [93] that not all perturbation directions in the space of state variables $\mathbf{V}$ matter. He suggested that perturbations in certain directions are suppressed and wave amplitudes are confined on a proper subcone of the space of state variables, the so-called *cone of amplitudes* $\mathcal{K}$. In order to find the cone $\mathcal{K}$, one needs to compute the product of the constraint matrix $\mathbf{M}_{\mathbf{D}}$ by the primitive system Jacobian $\mathbf{M}_{\mathbf{C}}$:

$$\mathbf{M}_{\mathbf{D}}(\hat{\mathbf{n}})\,\mathbf{M}_{\mathbf{C}}(\hat{\mathbf{n}}) = \mathbf{u}\cdot\hat{\mathbf{n}} \begin{pmatrix} 0 & \dots & 0 & 0 & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & 0 & 0 & 0 & 0 \\ 0 & \dots & 0 & 0 & -n_3 & n_2 \\ 0 & \dots & 0 & n_3 & 0 & -n_1 \\ 0 & \dots & 0 & -n_2 & n_1 & 0 \end{pmatrix}, \tag{2.55}$$

from which one can deduce that every non-zero row of the constraint matrix $\mathbf{M}_{\mathbf{D}}(\hat{\mathbf{n}})$ is a left eigenvector of $\mathbf{M}_{\mathbf{C}}(\hat{\mathbf{n}})$, with $\mathbf{u}\cdot\hat{\mathbf{n}}$ as eigenvalue. Because of this, the cone $\mathcal{K}$, defined as the null space $\mathcal{K}(\hat{\mathbf{n}}) = \ker[\mathbf{M}_{\mathbf{D}}(\hat{\mathbf{n}})]$, contains every characteristic direction of $\mathbf{M}_{\mathbf{C}}$ except two directions which span the space $\mathbb{R}^{10} \setminus \ker(\mathbf{M}_{\mathbf{D}})$. Thus, according to [93], the involutionary constrained system (2.52)–(2.53) is hyperbolic in the direction $\hat{\mathbf{n}}$ if the matrix $\mathbf{M}_{\mathbf{C}}(\hat{\mathbf{n}})$ is diagonalizable in the cone $\mathcal{K}(\hat{\mathbf{n}})$.

Let us demonstrate this for system (2.11) in a simpler way, following the approach adopted by Roe and Balsara [276] for the MHD equations. In one space dimension (with x as a single coordinate), the involution constraint simplifies to

$$\partial_x b_2 = 0, \qquad \partial_x b_3 = 0, \tag{2.56}$$

and therefore, b_2 and b_3 may be considered as given parameters, rather than evolved variables, and the vector of primitive variables can be reduced to $\mathbf{V} = (\rho_1,\ \rho_2,\ \mathbf{u}^\mathsf{T},\ p,\ \alpha_1,\ b_1)^\mathsf{T}$. For the corresponding reduced quasi-linear system, we can find a complete set of eight linearly independent eigenvectors

$$\mathbf{R} = \begin{pmatrix} 0 & 0 & 0 & 1 & k_9\, k_{\rho_1} & k_9\, k_{\rho_1} & k_8\, k_{\rho_1} & k_8\, k_{\rho_1} \\ 0 & 0 & 1 & 0 & -k_9\, k_{\rho_2} & -k_9\, k_{\rho_2} & -k_8\, k_{\rho_2} & -k_8\, k_{\rho_2} \\ 0 & 0 & 0 & 0 & -k_5\, k_9 & k_5\, k_9 & -k_4\, k_8 & k_4\, k_8 \\ 0 & -b_3 & 0 & 0 & -a_\sigma^2\, \beta_1\, \beta_2 & a_\sigma^2\, \beta_1\, \beta_2 & -a_\sigma^2\, \beta_1\, \beta_2 & a_\sigma^2\, \beta_1\, \beta_2 \\ 0 & b_2 & 0 & 0 & -a_\sigma^2\, \beta_1\, \beta_3 & a_\sigma^2\, \beta_1\, \beta_3 & -a_\sigma^2\, \beta_1\, \beta_3 & a_\sigma^2\, \beta_1\, \beta_3 \\ 0 & 0 & 0 & 0 & k_9\, \rho\, a^2 & k_9\, \rho\, a^2 & k_8\, \rho\, a^2 & k_8\, \rho\, a^2 \\ 1 & 0 & 0 & 0 & -k_9\, K & -k_9\, K & -k_8\, K & -k_8\, K \\ 0 & 0 & 0 & 0 & -k_5\, \beta_1\, \|\mathbf{b}\| & k_5\, \beta_1\, \|\mathbf{b}\| & k_4\, \beta_1\, \|\mathbf{b}\| & -k_4\, \beta_1\, \|\mathbf{b}\| \end{pmatrix}. \tag{2.57}$$

Note that the eigenvectors (2.57) can be found by removing the first column and the last five rows of the eigenvector matrix (2.50), relative to the GLM curl cleaning system. In the framework of [93], this corresponds to saying that, for $\hat{\mathbf{n}} = (1,\ 0,\ 0)^\mathsf{T}$, the weakly hyperbolic system (2.11), is diagonalizable in the cone $\mathcal{K}(\hat{\mathbf{n}})$. Yet, it is not clear if matrix $\mathbf{M_C}(\hat{\mathbf{n}})$ is diagonalizable (in the cone $\mathcal{K}(\hat{\mathbf{n}})$) for arbitrary direction $\hat{\mathbf{n}}$.

One may assume that this information has to be taken into account while developing a Riemann solver for a Godunov-type method not only for a weakly hyperbolic involutionary constrained system like system (2.11) but also for strongly hyperbolic systems (2.25) and (2.43). We hope to investigate this in future research.

2.2.6 Exact equilibrium solution for a symmetric droplet with diffuse interface

A steady state solution can be easily obtained for a two-dimensional water column or a three-dimensional droplet (hereafter we will take the liberty to call *droplets* the two-dimensional objects as well) by first assigning a radial profile $c(r) = c(\|\mathbf{x}\|)$ for the interface between the two phases, then computing the corresponding interface field $\mathbf{b} = \nabla c$ and balancing the surface tension forces, which are known once a specific geometrical configuration is chosen, with the pressure field.

For convenience, we define the dimensionless radial coordinate $r_* = r/R = \|\mathbf{x}\|/R$, with R being the radius of the water column or droplet. Here, with the notation $\mathbf{x}$ we indicate the Cartesian position vector, independently from the number of space dimensions d. We then set the radial profile of the colour function to be a smoothed Heaviside step function of the form

$$c(r_*) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{r_* - 1}{k_\varepsilon}\right), \tag{2.58}$$

with the dimensionless interface thickness parameter k_ε controlling the intensity of the smoothing. The Cartesian gradient of the colour function is immediately computed as

$$\mathbf{b}(\mathbf{x}) = -\frac{\mathbf{x}}{\sqrt{\pi}\,k_\varepsilon\,R\,\|\mathbf{x}\|}\exp\left[-\left(\frac{r_* - 1}{k_\varepsilon}\right)^2\right]. \tag{2.59}$$

It can be verified by easy calculations that the dimensionless interface scaling parameter k_ε corresponds to four times the standard deviation of the Gaussian curve describing the profile of the interface energy $\sigma\,\|\mathbf{b}\|$ along the radial direction, rendered dimensionless with respect to the nominal radius of the droplet. To give a clear physical meaning to the quantity, one can say that about 95.5% of the surface energy is stored in the region of space bounded by $1 - k_\varepsilon/2 \leq r_* \leq 1 + k_\varepsilon/2$. In a uniform flow, all the governing equations are satisfied for any choice of the density and volume fraction fields, and one can compute the radial pressure profile from the momentum equation by requiring that the pressure gradient be balanced with the divergence of the surface tension tensor $\boldsymbol{\Omega}$. Clearly from a physical/geometrical standpoint, the colour function and volume fraction variables are closely related and cannot be set independently. One can then derive from the momentum equation

$$\nabla p + \sigma\,\mathbf{b}\,\nabla\cdot\frac{\mathbf{b}}{\|\mathbf{b}\|} = \mathbf{0} \tag{2.60}$$

a simple ordinary differential equation

$$\frac{\mathrm{d}p}{\mathrm{d}r}(r) = \partial_r p(r) = -\frac{\mathbf{x}}{r}\cdot\left(\nabla\cdot\boldsymbol{\Omega}\right), \tag{2.61}$$

that, by evaluating the divergence of the capillarity tensor $\boldsymbol{\Omega}$ from (2.17) and substituting the ansatz for the interface field (2.59) yields

$$\frac{\mathrm{d}p}{\mathrm{d}r_*}(r_*) = -(d-1)\,\frac{\sigma}{R}\,\frac{1}{\sqrt{\pi}\,k_\varepsilon\,r_*}\,\exp\left[-\left(\frac{r_* - 1}{k_\varepsilon}\right)^2\right], \tag{2.62}$$

Note that the ODE (2.62) and thus the pressure profile depend parametrically on the group $(d-1)\,\sigma/R$ and are otherwise solely functions of the geometry expressed through Eq. (2.59). One can then directly integrate (2.62) with atmospheric pressure p_{atm} as a far field boundary condition in order to obtain the equilibrium pressure field

$$p\,(r_*) = p_{\text{atm}} + (d-1)\,\frac{\sigma}{R}\int_{r_*}^{\infty}\frac{1}{\sqrt{\pi}\,k_\varepsilon\,r'_*}\,\exp\left[-\left(\frac{r'_*-1}{k_\varepsilon}\right)^2\right]\mathrm{d}r'_*, \tag{2.63}$$

where r'_* is an auxiliary integration variable. The integral can be computed to machine precision with the aid of a Gauss–Legendre quadrature rule with the precaution of defining a sufficiently refined integration mesh in the interface region.

2.2.7 Consistency with the Young–Laplace law.

In the limit of vanishing interface thickness ($k_\varepsilon \to 0$), one can verify that Eq. (2.63) yields a pressure jump between atmospheric condition and the centre of the droplet of the form

$$\lim_{k_\varepsilon\to 0}\Delta p(k_\varepsilon) = \lim_{k_\varepsilon\to 0}\Big[p\,(r_*=0)|_{k_\varepsilon} - p_{\text{atm}}\Big] = (d-1)\,\frac{\sigma}{R}, \tag{2.64}$$

so that the well-known Young–Laplace formula is recovered in the limit of a sharp interface. Also, we can point out that, even for heavily smoothed droplets, the Young–Laplace formula provides similar values to the ones obtained from

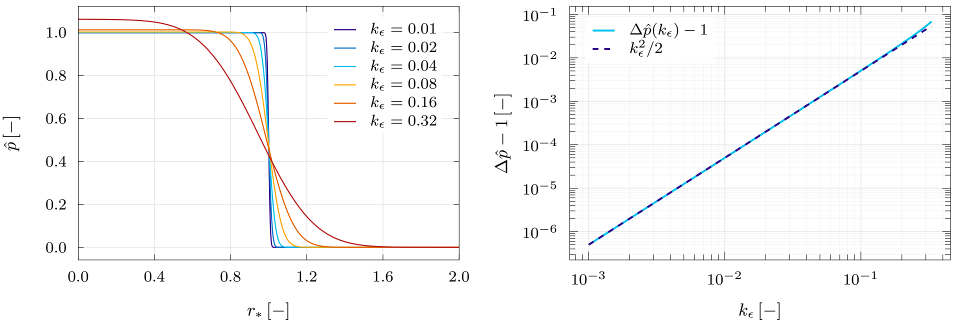


Figure 2.1. Exact pressure profiles for two-dimensional droplets with diffuse interface. On the left, the radial profile of the dimensionless pressure $\hat{p} = (p - p_{\text{atm}})\,R/[(d-1)\,\sigma]$ is shown for a range of values of the interface thickness parameter k_ε. On the right, we plot the error of the dimensionless pressure jump $\Delta\hat{p}$ of a *smooth droplet* with respect to the dimensionless Young–Laplace pressure jump (unity) as a function of the interface thickness parameter k_ε. The dashed line represents a power law approximation of the curve.

Eq. (2.63), as can be seen in the left panel of Figure 2.1, and that both estimates for the pressure jump converge to the same value quite quickly: from the right panel of Figure 2.1 it is apparent that the following approximation holds

$$\Delta p(k_\varepsilon) \simeq (d-1)\,\frac{\sigma}{R}\,\left(1+\frac{1}{2}\,k_\varepsilon^2\right), \tag{2.65}$$

which means that the pressure jump is overall affected only by relatively small deviations from the Young–Laplace law, even for droplets with rather large diffuse interface region and converges to the sharp interface reference solution quadratically as the interface thickness k_ε vanishes.

2.3 Elasto-plastic solids with material damage

In this section we present a simple model for the description of visco-elasto-plastic solids with material failure/damage, crack formation and propagation, and fatigue. The aim towards freeing fracture modelling from the necessity of prior knowledge regarding pre-existing cracks, and even an imposed crack propagation direction is expressed in modern literature by means of variational methods [51, 145] and phase field methods [287, 330].

In the framework of the unified model of continuum mechanics, such a goal is achieved by adding a single scalar equation for a so-called *damage variable* ξ to the unified model discussed in Section 2.1.1, which proved sufficient for simulating brittle and ductile materials, crack propagation in brittle solids, and even material fatigue effects. The model also features a volume fraction field α that enables the computation of real world problems with complex geometries on simple *uniform fixed Cartesian meshes*, in our case using high order ADER-DG methods with *a-posteriori* subcell limiting. The governing equations read

$$\partial_t\left(\rho\right) + \nabla\cdot\left(\rho\,\mathbf{u}\right) = 0, \tag{2.66a}$$

$$\partial_t\left(\alpha\,\rho\,\mathbf{u}\right) + \nabla\cdot\left(\alpha\,\rho\,\mathbf{u}\otimes\mathbf{u} - \alpha\,\boldsymbol{\Sigma}\right) = \mathbf{0}, \tag{2.66b}$$

$$\partial_t\left(\rho\,E\right) + \nabla\cdot\left(\rho\,E\,\mathbf{u} - \,\boldsymbol{\Sigma}\,\mathbf{u} + \mathbf{q}\right) = \mathbf{0}, \tag{2.66c}$$

$$\partial_t\left(\mathbf{A}\right) + \left(\nabla\mathbf{A}\right)\,\mathbf{u} + \mathbf{A}\left(\nabla\mathbf{u}\right) = \mathbf{Z}, \tag{2.66d}$$

$$\partial_t\left(\mathbf{J}\right) + \nabla\cdot\left[\left(\mathbf{J}\cdot\mathbf{u} + T\right)\,\mathbf{I}\right] + \left[\nabla\mathbf{J} - \left(\nabla\mathbf{J}\right)^{\mathsf{T}}\right]\mathbf{u} = -\mathbf{J}/\tau_{\mathrm{H}}, \tag{2.66e}$$

$$\partial_t\left(\xi\right) + \mathbf{u}\cdot\nabla\xi = \psi_\xi, \tag{2.66f}$$

$$\partial_t\left(\alpha\right) + \mathbf{u}\cdot\nabla\alpha = 0, \tag{2.66g}$$

$$\partial_t\left(c\right) + \mathbf{u}\cdot\nabla c = 0. \tag{2.66h}$$

Here, (2.66g) is the evolution equation for the colour function α that is needed in the diffuse interface approach (DIM) as introduced in [59, 311] for the description of solids of arbitrary geometry, where $\alpha = 1$ the solid is present, while $\alpha = 0$ represents vacuum. In proximity to the interface, the volume fraction α assumes intermediate values and transitions between the two extrema over several computational cells or sub-cells, hence the name Diffuse interface method [7, 32, 197, 203, 236, 334]. The mass conservation law is (2.66a) and ρ is the material density; (2.66b) is the momentum conservation law and $\mathbf{u}$ is the velocity field; (2.66d) is the evolution equation for the distortion field $\mathbf{A}$

Equation (2.66e) governs the evolution for the specific thermal impulse $\mathbf{J}$ which in turn regulates the heat conduction in the medium; (2.66f) is the evolution equation for the material damage variable $\xi \in [0,\ 1]$, where $\xi = 0$ indicates fully intact material and $\xi = 1$ fully damaged material. Finally, (2.66c) is the energy conservation law. The total stress tensor is $\mathbf{\Sigma} = -p\,\mathbf{I} + \boldsymbol{\sigma}$ is and $p = \rho^2\,\partial_\rho E$ is the contribution to the stress tensor due to volume deformations (pressure), while the contribution to the stress tensor due to shear and thermal stress is $\boldsymbol{\sigma} = -\rho\,\mathbf{A}\,\partial_\mathbf{A} E + \rho\,\mathbf{J}\,\partial_\mathbf{J} E$. The temperature is denoted by T and the heat flux is $\mathbf{q} = \rho\,T\,\partial_\mathbf{J} E = \rho\,T\,\alpha_T^2\,\mathbf{J}$. Note that in system (2.66) we only use a simplified diffuse interface approach, which completely neglects the dynamics of the air surrounding the solid medium. In our model the solid volume fraction α is only used to locate and track the shape of the moving solid, see [311]. In comparison, in the work of Favrie and Gavrilyuk and collaborators [138, 139, 243], real multi-phase flows of compressible solids embedded in compressible fluids were considered.

The dissipation in the medium includes two relaxation processes: the strain relaxation (or shear stress relaxation) characterized by the source term $\mathbf{Z} = -(3/\tau_\mathrm{m})\,(\det\mathbf{A})^{5/3}\,\mathbf{A}\ \mathrm{dev}\,\mathbf{G}$ and the heat flux relaxation source $\mathbf{J}/\tau_\mathrm{H}$, with τ_H being a state-dependent timescale computed as $\tau_\mathrm{H} = \tau_\mathrm{h}\,T_0\,\rho_0\,(T\,\rho)^{-1}$. The characteristic time τ_h is taken to be constant, with a practical choice explained below. A third stiff source $\psi_\xi = -\vartheta\,\partial_\xi E$, with ϑ and $\partial_\xi E$ to be given explicit expressions at this section, governs the dynamics of the damage variable ξ. Unlike the sources for $\mathbf{A}$ and $\mathbf{J}$, which are of relaxation type, the dynamics described by ψ_ξ are more akin to chemical reaction kinetics, having very mild effects until a certain activation threshold has been reached, past which an abrupt increase of the damage variable ξ is observed, with an almost discontinuous transition encountered for brittle materials and more complex nonlinear rate-dependent behaviour observed in materials with ductile failure. Such extremely fast timescales are the reason for which we adopted an adaptive sub-timestepping approach for the integration of the stiff source for the coupled strain/damage system.

Further to the above evolution equations we need to add the transport PDE for the local Lamé parameters λ and μ, material yield stress Y_0 and other material constants. For small deformations it is sufficient to assume them as constant in time, but for large deformations those parameters have to move with the local velocity field. The *colour variable c* is included in the model specifically for the purpose of transporting potentially non-homogeneous material parameters such as the Lamé constants λ_I, λ_D, μ_I, μ_D and the reference denisty ρ_0, according to the motion of the medium. The implied PDEs for the material parameters therefore read

$$\partial_t(\lambda) + \mathbf{u}\cdot\nabla\lambda = 0, \qquad \partial_t(\mu) + \mathbf{u}\cdot\nabla\mu = 0, \qquad \partial_t(\rho_0) + \mathbf{u}\cdot\nabla\rho_0 = 0. \quad (2.67)$$

If $\alpha = 1$, then it can be immediately seen that the model (2.66) is thermodynamically compatible as soon as the total energy is defined as a function of parameters of state [120, 170, 257, 277], since it satisfies the first and second principle of thermodynamics.

Furthermore, in order to close the system one must specify the total specific energy potential E as a function of the other state variables. This potential then generates all terms in the fluxes and source terms by means of its partial derivatives with respect to the state variables. Therefore, as discussed in [120], the energy specification is the key step in the model formulation.

Here, we make the choice $E = e_1 + e_2 + e_\mathrm{k}$, decomposing the energy into a contribution from the microscale e_1, the mesoscale e_2 and the macroscale e_k. The individual contributions read as follows:

$$e_1 = \frac{K}{2\rho_0}\left(1 - \frac{\rho}{\rho_0}\right)^2 + c_v\, T_0\left(\frac{\rho}{\rho_0}\right)\Big[\exp\left(S/c_v\right) - 1\Big]\,, \tag{2.68}$$

which is the equation of state (EOS) of the medium, where ρ_0 is the reference mass density and $K = \lambda + 2\mu/3$ is the bulk modulus expressed in terms of the two Lamé parameters λ and μ, which depend on the damage variable ξ, c_v is the heat capacity at constant volume, ρ_0 a reference density and T_0 is a reference temperature. The mesoscale energy is taken in the form

$$e_2 = \frac{1}{4}\, c_\mathrm{s}^2\ \mathrm{tr}\left(\mathrm{dev}\,\mathbf{G}\ \mathrm{dev}\,\mathbf{G}\right) + \frac{1}{2}\,\alpha_T^2\ \left\|\mathbf{J}\right\|^2\,, \tag{2.69}$$

where $c_\mathrm{s} = \sqrt{\mu/\rho_0}$ is the sound speed for small amplitude shear waves and α_T is related to the speed of heat waves in the medium (also called the second sound [260]): in the linear regime one can find $a_\mathrm{h} = \sqrt{\alpha_T^2\, T/c_v}$ to be a characteristic speed for the heat waves. The heat relaxation time τ_h is specified as $\tau_\mathrm{h} = \kappa\,\left(\rho_0\, T_0\, \alpha_T^2\right)^{-1}$, so to impose a known value for the thermal conductivity κ. For an alternative choice of the mesoscale energy e_2 with better mathematical properties, see [242]. Here $\mathrm{dev}\,\mathbf{G} = \mathbf{G} - (\mathrm{tr}\,\mathbf{G}/3)\,\mathbf{I}$ is the deviator of the metric tensor $\mathbf{G} = \mathbf{A}^\mathsf{T}\mathbf{A}$ that describes the deformation of the medium. Furthermore, the macro-scale energy is the classical kinetic energy $e_\mathrm{k} = \left\|\mathbf{u}\right\|^2/2$. Note that we define the energies $e_1,\ e_2$ by (2.68), (2.69) in order to be able to reduce $e_1 + e_2$ in case of small deformations to the quadratic energy of small strain corresponding to Hooke's law and indeed a more complex nonlinear dependence of $e_1,\ e_2$ on density and strain is allowed.

2.3.1 Constitutive relations for the damaged medium

In order to make the model (2.66) applicable to the description of the damage processes, it is necessary to define accordingly its material parameters and constitutive relations. Our approach is based on the mixture model proposed in [272] for the damage of an elastoplastic continuum in case of small deformations and generalized later for the study of the structure of failure waves in the case of finite deformations in [278]. The idea of the mentioned approach is to consider a damaged material as a mixture of the intact and "fully damaged" materials. These two materials have their own material parameters and closing relations, such as functions characterizing the rate of shear stress relaxation. The key strength of the approach of Resnyansky and Romenski [272, 278] is that transition from the intact material to the fully damaged material is governed by *a single additional* damage variable $\xi \in [0,\ 1]$ satisfying the advection–reaction equation (2.66f) with source term depending on the state parameters of the medium (pressure, stress and temperature). Then, if to assume that in the case of small deformations the mixture parameters of state satisfy the simple mixture rules such as an additivity of small strain and continuity of stress field, one can derive the governing equations for the damaged medium considered as an elastoplastic continuum with material parameters and closing relations depending on the damage variable ξ. This mixture model of damaged medium allows one to describe a degradation of elastic moduli during the damage process and to fit experimental stress-strain diagrams depending on the strain rate.

Let us assume that the elastic moduli (Lamé constants) of the intact material λ_I, μ_I and of the fully damaged material λ_D, μ_D are known. Further, assume that the material parameters of both phases, corresponding to the heat transfer processes, such as heat capacity, thermal expansion coefficient and thermal conductivity coefficients are close to each other. The latter assumptions allow us to avoid an excessive complexity of the model and can be relaxed if additional modelling detail is required.

Thus, following [272], one can define the elastic moduli in the equation of state of the damaged material as

$$\mu(\xi) = \frac{\mu_I\,\mu_D}{\tilde{\mu}}, \qquad \lambda(\xi) = \frac{K_I\,K_D}{\tilde{K}} - \frac{2}{3}\,\mu(\xi), \tag{2.70}$$

where $K_I = \lambda_I + 2\,\mu_I/3$, $K_D = \lambda_D + 2\,\mu_D/3$, $\tilde{K} = \xi\,K_I + (1-\xi)\,K_D$, $\tilde{\mu} = \xi\,\mu_I + (1-\xi)\,\mu_D$. It is easy to see that for the intact material ($\xi = 0$) we have the mixture moduli equal to the ones of the intact material: $\lambda = \lambda_I$, $\mu = \mu_I$, while for the fully damaged material ($\xi = 1$) we obtain $\lambda = \lambda_D$, $\mu = \mu_D$.

The mixture-model averaging method of [272] applied to the definition of the rate of shear stress relaxation gives us the dependence of the shear stress relaxation time τ_1 on the damage variable ξ as follows:

$$\tau_1 = \left(\frac{1-\xi}{\tau_I} + \frac{\xi}{\tau_D} \right)^{-1}, \tag{2.71}$$

where τ_I and τ_D are the shear stress relaxation times for the intact and fully damaged materials respectively, which are usually highly nonlinear functions of the parameters of state. The particular choice of τ_I and τ_D that is used in this work reads

$$\tau_I = \tau_{I0} \exp\left[\alpha_I - \beta_I \left(1-\xi\right) Y\right], \quad \tau_D = \tau_{D0} \exp\left[\alpha_D - \beta_D \, \xi \, Y\right], \tag{2.72}$$

where Y is the equivalent stress (e.g. the Von Mises stress), while τ_{I0}, α_I, β_I, τ_{D0}, α_D, β_D are constants.

The parameter ϑ governing the rate of damage is also a nonlinear function of the parameters of state and in our numerical examples we take it in the following form:

$$\vartheta = \vartheta_0 \left(1-\xi\right) \left(\xi + \xi_\varepsilon\right) \left[\left(1-\xi\right) \left(\frac{Y}{Y_0}\right)^a + \xi \left(\frac{Y}{Y_1}\right) \right], \tag{2.73}$$

where ξ_ε, Y_0, Y_1, and a are constant parameter to be calibrated for a specific material. ξ_ε is usually taken as 10^{-16} in order to provide the growth of ξ with the initial data $\xi = 0$.

The derivative $\partial_\xi E$ can be computed with the use of (2.68), (2.69), (2.70) and reads as

$$\frac{\partial E}{\partial \xi} = -\frac{\left(K_I - K_D\right) K_I \, K_D}{2 \, \rho_0 \, \tilde{K}^2} \left(1 - \frac{\rho}{\rho_0}\right)^2 - \frac{\left(\mu_I - \mu_D\right) \mu_I \, \mu_D}{4 \, \rho_0 \, \tilde{\mu}^2} \operatorname{tr}(\operatorname{dev} \mathbf{G} \, \operatorname{dev} \mathbf{G}). \tag{2.74}$$

For the sake of simplicity we assume that the material parameters characterizing thermal properties of intact and damaged material are the same, that means that c_v, α_T^2 are constant and do not depend on ξ.

2.3.2 Properties and challenges of the model

The model (2.66) generalizes the unified model of continuum mechanics presented in [120, 257], taking into account also material damage processes and the possibility to simulate moving free surface problems via the use of a diffuse interface method (DIM) that simply employs a scalar colour function α in order to define where the solid is present ($\alpha = 1$) and where it is not ($\alpha = 0$), see [311]. The additional evolution equations for the material parameters (2.67) are added in order to capture correctly the motion of a continuum with heterogeneous material parameters undergoing large deformations. Note, that the governing equations for the heat transfer in the present model differ from that considered in [120] and are taken in the form originally proposed in [277]. The latter formulation of the heat transfer processes seems to be more natural, because it can be derived as a geometric-like constraint in the minimization of a Lagrangian and is in agreement with the Hamiltonian GENERIC formulation [256], as all equations from the general SHTC class.

If $\alpha = 1$ then the PDE system (2.66) is a hyperbolic thermodynamically compatible system and advanced high-order methods can be applied to solve these equations. Nevertheless, there are very stiff algebraic source terms in the equations for distortion, thermal impulse and damage variable which create a significant difficulty for numerical computations. The exponential ODE integrator presented in Chapter 4 helps to avoid problems related to the stiffness of the algebraic source terms in the governing PDE system.

Another difficulty is an application of the model of damaged medium to the solution of real problems. It relates to the definition of the constitutive relations and, in particular, in the definition of the function ϑ characterizing the rate of change of the damage variable, as well as the shear and heat flux relaxation parameters τ_{m}, and τ_{h}. On the one hand, the material constants in the equation of state of the intact elastic medium can in principle be found from experimental measurements, which means that the Lamé constants λ_I, μ_I and the heat capacity c_v can be considered as known. On the other hand, there is no way to get the sample of the fully damaged medium which appeared as a result of the deformation of the medium, and hence there seems to be no possibility to obtain the material constants of the damaged material by direct measurements.

The description of the damaged material requires not only material constants in the equation of state, but also a constitutive relations for the shear stress relaxation time τ_{m} and for the function ϑ governing the rate of the damage. All these characteristics of the medium can in principle be found with the use of experimental stress-strain diagrams, which are usually available from standard traction, torsion and compression experiments in solid mechanics. The method consists in doing a series of numerical computations and obtaining a set of stress-strain diagrams numerically, and then by variation of the material constants try to fit as much as possible the available experimental diagrams. Such a procedure was successfully used in the past for the closure of the nonlinear elastoplastic Godunov-Romenski model and the idea how to do this can be found in [170]. Recently this method has also been used for the closure of complex elastoplastic media with hardening, see [23].

The dynamic behavior of damaged materials is very complex and depends on the type of the medium. The damage followed by fracture can be brittle, or brittle-ductile, which means that the stress-strain diagrams can be completely different for different materials. In Section 6.3.3 two typical examples of brittle and ductile material behavior are presented. In our model, these diagrams can also depend on the *strain rate*, which is also shown in the same section. In this work we do not calibrate the model parameters explicitly for such a dependence, since there is no experimental data available. Nevertheless, the constitutive relations chosen in Section 2.3.1 and the set of material constants presented in the Section on numerical tests gives good results in all considered test problems.

Last but not least, our thermodynamically compatible approach to material failure naturally includes also the phenomenon of *material fatigue*, i.e. the reduced resistance of the material to stress applied over a very large number of load cycles, see Section 6.3.3.

2.4 Eigenvalue estimates for the unified model of continuum mechanics

Due to the large size of the hyperbolic PDE system coupling convective, acoustic, thermal, shear, and capillarity effects, it is at the present time *impossible* to explicitly compute the eigenstructure of the full system, that is, the eigenvalues and eigenvectors of the matrix $\mathbf{C}_1 = \mathbf{C} \cdot \hat{\mathbf{n}}$, called in an abuse of terminology, Jacobian of the system in the direction specified by the unit vector $\hat{\mathbf{n}}$. Formally, $\mathbf{C}$ is defined as

$$\mathbf{C} = \left(\frac{\partial \mathbf{Q}}{\partial \mathbf{V}}\right)^{-1} \frac{\partial \mathbf{F}}{\partial \mathbf{V}} + \mathbf{B}\,\frac{\partial \mathbf{Q}}{\partial \mathbf{V}}, \tag{2.75}$$

the quasi-linear form of the governing equations

$$\partial_t \mathbf{V} + \mathbf{C}\,\nabla \mathbf{V} = \mathbf{S}, \tag{2.76}$$

recalling that the general first order balance law reads

$$\partial_t \mathbf{Q} + \nabla \cdot \mathbf{F} + \mathbf{B}\,\nabla \mathbf{Q} = \mathbf{S}. \tag{2.77}$$

As a matter of fact, even the eigenvalues of $\mathbf{C}$ can be obtained in closed form only with several simplificative assumptions, such as setting the distortion matrix $\mathbf{A}$ identity or the thermal impulse vector $\mathbf{J}$ to null. Moreover, even the numerical computation of the eigenvalues of the system can become prohibitively expensive as the Jacobian matrix of the fully coupled system for two-phase flow with shear, surface tension, and heat conduction is large enough that application of standard numerical eigenvalue methods whenever a Riemann solver of Rusanov of HLL type is evaluated is rather wasteful from a computational standpoint.

Since the Rusanov flux only requires an estimate of the maximum absolute eigenvalue λ_{max} of the system Jacobian, one might be tempted to employ a power iteration method (Von Mises iteration or Rayleigh quotient iteration) for the computation of the spectral radius of the matrix. However, even the *evaluation* of the Jacobian matrix for a given state vector is arithmetically very intensive and thus it is preferable to avoid the procedure entirely, favouring instead simpler estimates that can be computed directly from the state vector.

A practical and effective choice, which we found to be rather safe for the estimation of the spectral radius of the Jacobian matrix of the full system, while only leading to occasional mild overestimates, is setting

$$\lambda_{\text{max}} = \max\left(|\mathbf{u}\cdot\hat{\mathbf{n}} + \lambda|,\ |\mathbf{u}\cdot\hat{\mathbf{n}} - \lambda|\right) \tag{2.78}$$

with

$$\lambda = \sqrt{\lambda_{\text{ps}}^2 + \lambda_{\text{h}}^2 + \lambda_{\text{t}}^2}, \tag{2.79}$$

where λ_{ps} accounts for mixed pressure/shear waves, λ_{h} is an estimate of the contribution to the maximum eigenvalue due to heat waves only, and λ_{t} is an estimate of the contribution due to capillarity waves only. In principle, the eigenvalues of the full model including two-phase flow, shear, surface tension and heat conduction, appear to couple all effects in mixed thermal-shear-pressure-capillarity waves, and the same is true for the surface tension sub-system yielding acoustic-capillary waves, the shear sub-system yielding pressure-shear waves, and the heat conduction sub-system yielding thermo-acoustic waves, thus it is impossible to rigorously assign a wavespeed to only one of the effects. Nonetheless, by means of numerical experimentations, we found that surprisingly robust and accurate estimates of the maximum absolute eigenvalue of the system can be achieved with appropriate choices of the three estimates λ_{ps}, λ_{h}, and λ_{t}, which are given in the following paragraphs.

2.4.1 Wavespeed estimate for capillarity waves

We begin discussing the eigenvalue estimates for the capillary system, Since the full eigenstructure of the capillarity sub-system

$$\begin{cases} \partial_t\left(\alpha_1\,\rho_1\right) + \nabla\cdot\left(\alpha_1\,\rho_1\,\mathbf{u}\right) = 0, \\ \partial_t\left(\alpha_2\,\rho_2\right) + \nabla\cdot\left(\alpha_2\,\rho_2\,\mathbf{u}\right) = 0, \\ \partial_t\left(\rho\,\mathbf{u}\right) + \nabla\cdot\left(\rho\,\mathbf{u}\otimes\mathbf{u} + p\,\mathbf{I} + \boldsymbol{\Omega}\right) = \mathbf{0}, \\ \partial_t\left(\rho\,E\right) + \nabla\cdot\left[\left(\rho\,E + p\right)\,\mathbf{u} + \boldsymbol{\Omega}\,\mathbf{u}\right] = 0, \\ \partial_t\left(\alpha_1\right) + \mathbf{u}\cdot\nabla\alpha_1 - K\,\nabla\cdot\mathbf{u} = 0, \\ \partial_t\left(\mathbf{b}\right) + \left(\nabla\mathbf{b}\right)\,\mathbf{u} + \left(\nabla\mathbf{u}\right)^{\mathsf{T}}\,\mathbf{b} = \mathbf{0}, \qquad \nabla\times\mathbf{b} = \mathbf{0}, \end{cases} \tag{2.80}$$

can be computed analytically as shown in Section 2.2. A simple estimate for the speed associated with capillarity waves λ_{t}, can be computed analytically by recognising different velocities (i.e. the adiabatic sound speed) in the expression of the eigenvalues for the system of two-phase flow with surface tension, as already formally detailed in Chapter 2. For clarity, the capillarity tensor in the form

$$\boldsymbol{\Omega} = \sigma\,\left(\left\|\mathbf{b}\right\|^{-1}\,\mathbf{b}\otimes\mathbf{b} - \left\|\mathbf{b}\right\|\,\mathbf{I}\right), \tag{2.81}$$

and K is computed as $K = \left(\rho_2\,a_2^2 - \rho_1\,a_1^2\right)\,\alpha_1\,\alpha_2/(\alpha_1\,\rho_2\,a_2^2 + \alpha_2\,\rho_1\,a_1^2)$. We generically denote with a the mixture speed of sound (adiabatic pressure waves), which, for Kapila-type models such as those considered in this section, is the Wood [329] speed of sound which reads

$$a = \sqrt{\frac{\rho_1\,a_1^2\,\rho_2\,a_2^2}{\rho\,\left(\alpha_1\,\rho_2\,a_2^2 + \alpha_2\,\rho_1\,a_1^2\right)}}, \tag{2.82}$$

with the frozen soundspeeds a_1 and a_2 computed as

$$a_1 = \sqrt{\rho_1^{-1}\,\gamma_1\,\left(p + \Pi_1\right)}, \qquad a_2 = \sqrt{\rho_2^{-1}\,\gamma_2\,\left(p + \Pi_2\right)}, \tag{2.83}$$

due to the stiffened gas equation of state being adopted for both phases,

The simplest estimate for the speed associated with capillarity waves λ_{t}, is then given by

$$\lambda_{\mathrm{t}} = a_\sigma = \sqrt{\rho^{-1}\,\sigma\,\left\|\mathbf{b}\right\|\,\left[1 - \left\|\mathbf{b}\right\|^{-2}\left(\mathbf{b}\cdot\hat{\mathbf{n}}\right)^2\right]}. \tag{2.84}$$

We refer to Section 2.2 for a detailed derivation of the eigenstructure of the capillarity sub-system.

2.4.2 Wavespeed estimate for heat waves

In analogy to the approach adopted for capillarity waves, we consider a simplified system for two-phase flow with heat conduction

$$\begin{cases}\partial_t\,(\alpha_1\,\rho_1) + \nabla\cdot(\alpha_1\,\rho_1\,\mathbf{u}) = 0,\\ \partial_t\,(\alpha_2\,\rho_2) + \nabla\cdot(\alpha_2\,\rho_2\,\mathbf{u}) = 0,\\ \partial_t\,(\rho\,\mathbf{u}) + \nabla\cdot\Big(\rho\,\mathbf{u}\otimes\mathbf{u} + p\,\mathbf{I} + \rho\,\alpha_T^2\,\mathbf{J}\otimes\mathbf{J}\Big) = \mathbf{0},\\ \partial_t\,(\rho\,E) + \nabla\cdot\Big[(\rho\,E + p)\;\mathbf{u} + \rho\,\alpha_T^2\,T\,\mathbf{J}\Big] = 0,\\ \partial_t\,(\alpha_1) + \mathbf{u}\cdot\nabla\alpha_1 - K\,\nabla\cdot\mathbf{u} = 0,\\ \partial_t\,(\mathbf{J}) + \nabla\cdot\left[(\mathbf{J}\cdot\mathbf{u} + T)\;\mathbf{I}\right] + \left[\nabla\mathbf{J} - (\nabla\mathbf{J})^{\mathsf{T}}\right]\mathbf{u} = -\mathbf{J}/\tau_{\mathrm{H}},\end{cases} \tag{2.85}$$

with

$$T = \frac{\rho\,e}{\rho\,c_v},\quad \tau_{\mathrm{H}} = \rho\,T/(\rho_0\,T_0\,\tau_{\mathrm{h}}). \tag{2.86}$$

A simple wavespeed expression can be obtained in the limit $\|\mathbf{J}\|$, which is often verified in practice due to $\mathbf{J}$ relaxing to $-\tau_{\mathrm{H}}\,\nabla T$ if the source term $-\mathbf{J}/\tau_{\mathrm{H}}$ is stiff, that is, when $\tau_{\mathrm{H}} \to 0$. In this case, the characteristic velocity for heat waves is computed as

$$a_{\mathrm{h}}^2 = \frac{\rho\,\alpha_T^2\,T}{\rho\,c_v}. \tag{2.87}$$

Two additional more robust estimates, accounting for nonzero $\|\mathbf{J}\|$ and, in particular, accurate for both small values of $\|\mathbf{J}\|$ and for very large values of $\|\mathbf{J}\|$, are given by

$$\lambda_{\mathrm{h},0}^2 = a_{\mathrm{h}}^2 + \alpha_T^2\;\|\mathbf{J}\|^2 + 2\,\alpha_T^2\,J_{\mathrm{n}}^2, \tag{2.88}$$

$$\lambda_{\mathrm{h},1}^2 = \lambda_{\mathrm{h},0}^2 + 4\;(1-k_2)\;k_3\,|j_{\mathrm{n}}|\,a_{\mathrm{h}}^2\left(2\,\lambda_{\mathrm{h},0}^2/3\right)^{-1/2}, \tag{2.89}$$

with

$$k_1 = \rho\,\frac{\partial T}{\partial p} = \frac{1}{c_v}\left(\frac{\alpha_1}{\gamma_1 - 1} + \frac{\alpha_2}{\gamma_2 - 1}\right),\quad k_2 = c_v\,k_1,\quad k_3 = \alpha_T^2\,k_1^2, \tag{2.90}$$

and

$$\mathbf{j} = \frac{\mathbf{J}}{k_1},\quad j_{\mathrm{n}} = \frac{\mathbf{J}\cdot\hat{\mathbf{n}}}{k_1},\quad J_{\mathrm{n}} = \mathbf{J}\cdot\hat{\mathbf{n}}. \tag{2.91}$$

Furthermore, if even higher accuracy of the estimates is required, one can define

$$\lambda_{\mathrm{h},2}^2 = \lambda^2 - a^2, \tag{2.92}$$

which is obtained by subtracting the acoustic component a^2 from the thermo-acoustic wavespeed λ, and can be computed by solving the quartic polynomial

$$\lambda^4 + d_3\,\lambda^3 + d_2\,\lambda^2 + d_1\,\lambda + d_0 = 0, \tag{2.93}$$

having coefficients given by

$$\begin{aligned}
d_0 &= a_{\mathrm{h}}^4\,k_2/k_3 + k_3\,\left(a^2 - k_3\,j_{\mathrm{n}}^2\right)\,\left(\|\mathbf{j}\|^2 - j_{\mathrm{n}}^2\right) - k_3\,j_{\mathrm{n}}^2\,a_{\mathrm{h}}^2,\\
d_1 &= -2\,\left[k_3\,a^2 + (1-k_2)\,a_{\mathrm{h}}^2\right]\,j_{\mathrm{n}},\\
d_2 &= -\left[a^2 + a_{\mathrm{h}}^2 + k_3\,\left(\|\mathbf{j}\|^2 + 2\,j_{\mathrm{n}}^2\right)\right],\\
d_3 &= 0.
\end{aligned} \tag{2.94}$$

Then the speed of heat waves can be taken to be

$$\lambda_{\mathrm{h}} = \sqrt{\max(\lambda_{\mathrm{h},0}^2,\ \lambda_{\mathrm{h},1}^2,\ \lambda_{\mathrm{h},2}^2)}. \tag{2.95}$$

A final remark is that, in practice, the solution of (2.93), can be avoided since the two previous estimates $\lambda_{\mathrm{h},0}^2$ and $\lambda_{\mathrm{h},1}^2$ already capture the linear limit $\|\mathbf{J}\| \to 0$, yielding $\lambda_{\mathrm{h},0}^2 = \lambda_{\mathrm{h},1}^2 = a_{\mathrm{h}}^2$, which in most cases is an adequate choice on its own (it is the one adopted in [120]), as well as providing the correct asymptotic behavior when $\|\mathbf{J}\|$ is very large.

2.4.3 Wavespeed of large amplitude pressure-shear waves

The sub-system for two-phase flow with viscosity, which of course can also model hyperelastic solids by simply settings the relaxation time $\tau \to \infty$, reads

$$\begin{cases}
\partial_t\,(\alpha_1\,\rho_1) + \nabla\cdot(\alpha_1\,\rho_1\,\mathbf{u}) = 0,\\
\partial_t\,(\alpha_2\,\rho_2) + \nabla\cdot(\alpha_2\,\rho_2\,\mathbf{u}) = 0,\\
\partial_t\,(\rho\,\mathbf{u}) + \nabla\cdot\left(\rho\,\mathbf{u}\otimes\mathbf{u} + p\,\mathbf{I} + \rho\,c_{\mathrm{s}}^2\,\mathbf{G}\ \mathrm{dev}\,\mathbf{G}\right) = \mathbf{0},\\
\partial_t\,(\rho\,E) + \nabla\cdot\left[(\rho\,E + p)\ \mathbf{u} + \rho\,c_{\mathrm{s}}^2\,\mathbf{G}\ \mathrm{dev}\,\mathbf{G}\,\mathbf{u}\right] = 0,\\
\partial_t\,(\alpha_1) + \mathbf{u}\cdot\nabla\alpha_1 - K\,\nabla\cdot\mathbf{u} = 0,\\
\partial_t\,(\mathbf{A}) + (\nabla\mathbf{A})\ \mathbf{u} + \mathbf{A}\,(\nabla\mathbf{u}) = -3\,\tau^{-1}\ (\det\mathbf{A})^{5/3}\ \mathbf{A}\ \mathrm{dev}\,\mathbf{G}.
\end{cases} \tag{2.96}$$

By algebraic manipulation of the generalised Jacobian matrix appearing in the quasilinear form of (2.96), one can find that the eigenvalues do not depend directly on the nine components of $\mathbf{A}$, but only on the metric tensor $\mathbf{G} = \mathbf{A}^\mathsf{T}\,\mathbf{A}$. In three space dimensions, i.e. for a general metric tensor of the form

$$\mathbf{G} = \begin{pmatrix} G_{11} & G_{12} & G_{13} \\ G_{12} & G_{22} & G_{23} \\ G_{13} & G_{23} & G_{33} \end{pmatrix}, \tag{2.97}$$

one has that the three eigenvalues λ_1, λ_2, and λ_3 associated with mixed pressure/shear waves are the square roots of the eigenvalues of the symmetric matrix

$$\mathbf{M} = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{12} & m_{22} & m_{23} \\ m_{13} & m_{23} & m_{33} \end{pmatrix}, \tag{2.98}$$

with

$$\begin{aligned}
m_{11} &= c_\mathrm{s}^2\,\left[10\,G_{11}^2 + 9\,\left(G_{12}^2 + G_{13}^2\right) - 3\,G_{11}\,(G_{22}+G_{33})\right]/3 + a^2,\\
m_{22} &= c_\mathrm{s}^2\,\left\{4\,G_{12}^2 + 3\,G_{23}^2 + G_{22}\,[2\,(G_{11}+G_{22}) - G_{33}]\right\}/3,\\
m_{33} &= c_\mathrm{s}^2\,\left\{4\,G_{13}^2 + 3\,G_{23}^2 + G_{33}\,[2\,(G_{11}+G_{33}) - G_{22}]\right\}/3,\\
m_{12} &= 2\,c_\mathrm{s}^2\,[3\,G_{13}\,G_{23} + G_{12}\,(4\,G_{11} + 2\,G_{22} - G_{33})]\,/3,\\
m_{13} &= 2\,c_\mathrm{s}^2\,[3\,G_{12}\,G_{23} + G_{13}\,(4\,G_{11} + 2\,G_{33} - G_{22})]\,/3,\\
m_{23} &= 2\,c_\mathrm{s}^2\,[2\,G_{12}\,G_{13} + G_{23}\,(G_{11} + G_{22} + G_{33})]\,/3,
\end{aligned} \tag{2.99}$$

Note that the above expression for the components of $\mathbf{M}$ is associated with the Jacobian matrix of the system projected along the x-axis direction, i.e. for $\hat{\mathbf{n}} = (1,\ 0,\ 0)^\mathsf{T}$. However, due to the rotational invariance of the governing equations, it is always possible to define a local reference frame in which the new x direction is that along which the directional eigenvalue estimate is sought, that would be along any of the outward normal vectors of the space-time faces when computing approximate Riemann fluxes. The characteristic polynomial associated with $\mathbf{M}$ then reads

$$(\lambda^2)^3 - (\lambda^2)^2\ \mathrm{tr}\,\mathbf{M} - \lambda^2\ \left[\mathrm{tr}\,(\mathbf{M}\,\mathbf{M}) - (\mathrm{tr}\,\mathbf{M})^2\right]/2 - \det\mathbf{M} = 0 \tag{2.100}$$

and can be solved analytically for λ^2 by means of the Del Ferro–Tartaglia–Cardano procedure. However, from a computational standpoint, it is much more efficient and accurate to apply the Jacobi method to the symmetric matrix $\mathbf{M}$ directly. In any case we formally set the wavespeed estimate due to mixed pressure/shear waves to be $\lambda_\mathrm{ps} = \max(\lambda_1,\ \lambda_2,\ \lambda_3)$, with λ_1^2, λ_2^2, λ_3^2 obtained by solving (2.100).

In two space dimensions, the G_{13} and G_{23} components of the metric tensor $\mathbf{G}$ vanish, thus the eigenvalues of the system can be found as the square roots of the eigenvalues of a simplified matrix

$$\mathbf{M} = \begin{pmatrix} m_{11} & m_{12} & 0 \\ m_{12} & m_{22} & 0 \\ 0 & 0 & m_{33} \end{pmatrix}, \tag{2.101}$$

for which closed form expression are easy to compute and write. In particular the components of (2.98) simplify to

$$\begin{aligned} m_{11} &= c_\mathrm{s}^2 \left\{ 4\, G_{11}^2 + 9 \left[G_{12}^2 + G_{11}\, \left(G_{11} - \operatorname{tr}\mathbf{G}/3\right)\right]\right\}/3 + a^2, \\ m_{22} &= c_\mathrm{s}^2 \, \left[G_{22}\, \left(2 \operatorname{tr}\mathbf{G} - 3\, G_{33}\right) + 4\, G_{12}\, G_{12}\right]/3, \\ m_{33} &= c_\mathrm{s}^2\, G_{33}\, \left(2 \operatorname{tr}\mathbf{G} - 3\, G_{22}\right)/3, \\ m_{12} &= 2\, c_\mathrm{s}^2\, G_{12}\, \left(4\, G_{11} + 2\, G_{22} - G_{33}\right)/3, \end{aligned} \tag{2.102}$$

and the eigenvalues of the sub-model for two-phase flow with shear, associated with pressure/shear waves are

$$\lambda_1 = \sqrt{m_4 + m_5}, \quad \lambda_2 = \sqrt{m_4 - m_5}, \quad \lambda_3 = \sqrt{m_{33}}, \tag{2.103}$$

with

$$m_4 = \left(m_{11} + m_{22}\right)/2, \quad m_5 = \sqrt{m_4^2 + m_{12}^2 - m_{11}\, m_{22}}. \tag{2.104}$$

For small deformations, i.e. when $\mathbf{G} \to \mathbf{I}$, it is easy to verify that the components of (2.102) further simplify

$$m_{11} = a^2 + 4\, c_\mathrm{s}^2/3, \quad m_{22} = c_\mathrm{s}^2, \quad m_{33} = c_\mathrm{s}^2, \quad m_{12} = 0, \tag{2.105}$$

and thus the linearised estimates for the eigenvalues are recovered

$$\lambda_1 = \sqrt{a^2 + 4\, c_\mathrm{s}^2/3}, \quad \lambda_2 = c_\mathrm{s}, \quad \lambda_3 = c_\mathrm{s}, \tag{2.106}$$

as given and employed in [120]. For simplicity, when adopting semi-implicit schemes, such as the one presented in Chapter 5, the eigenvalues estimates used in the Rusanov dissipation and to define the timestep size, are taken to be the same used for explicit methods, but setting the adiabatic sound speed to $a = 0$, reflecting the fact that the implicit solution of the pressure subsystem eliminates the timestep restrictions due to acoustic waves.

3 High order ADER schemes

High order methods of the ADER family have proven to be an effective testing platform for novel hyperbolic models in a variety of fields: dispersive shallow water flows [25, 60, 136, 172], the Euler–Korteweg/Van der Waals system and the nonlinear Schrödinger equation [60, 99], shallow water flows with turbulence [61, 158], or the Z4 formulation of Einstein's Field equations [33, 34, 113, 115].

Especially ADER Discontinuous Galerkin schemes with *a posteriori* Finite Volume subcell limiting, constitute an excellent framework for investigating the behaviour of complex systems of hyperbolic partial differential equations, thanks to the superior resolution they naturally offer due to high order piecewise polynomial representation of data, and without compromise with regard to robustness in presence of strong gradients or shockwaves, thanks to the shock capturing properties at the aid of an automatic procedure (MOOD [88, 100, 221]) that allows to locally switch between Finite Volume and Discontinuous Galerkin solution methods, whenever the solution obtained with the DG scheme is found to be oscillatory or not physically admissible.

Moreover, ADER schemes can be preferable to the more commonly used semi-discrete methods with Runge–Kutta timestepping [171, 297, 298] because: (*i*) They are not affected by the well known Butcher barriers [64] as opposed to Runge–Kutta schemes and therefore Finite Volume schemes of the ADER variety require only one reconstruction step for each timestep [122], rather than one for each Runge–Kutta substage. This translates into a reduced computational effort for schemes of very high order of accuracy and mitigates the numerical diffusion issues due to multiple reconstruction operations per time step. (*ii*) They can be implemented in a predictor–corrector fashion, with the predictor step fully local to each mesh element, which makes them particularly well suited for parallelization and high performance computing. This is especially true for the Discontinuous Galerkin variant of ADER schemes.

A further positive aspect of the ADER formalism is that, thanks to the fully-discrete, one-step nature of the methods, computations are operated between two successive timesteps on a set of clearly defined *space-time* control volumes, and this in principle allows to *prescribe* the shape of such control volumes not only in space, through a structured or unstructured mesh, but also *in time*, by specifying motion and deformations of the mesh elements between two successive timesteps. This ultimately yields a natural way of constructing *direct* Arbitrary-Lagrangian-Eulerian (ALE) numerical methods.

The origin of ADER Finite Volume schemes can be traced back back to the pioneering work of Toro and Titarev [289, 315, 316, 318, 320] on approximate solvers for the generalized Riemann problem (GRP). ADER schemes have been originally developed in the Eulerian framework on fixed grids [112, 289, 315, 316, 318, 320] and have subsequently also been extended to moving meshes in the ALE context [39, 41, 42, 46]. More in detail, the family of ADER methods was expanded to treat nonlinear hyperbolic conservation laws with sources on Cartesian grids in [315, 316, 320, 321], scalar conservation laws on unstructured meshes in [202], linear and nonlinear hyperbolic systems on unstructured meshed in two and three space dimensions in [117, 118], and magnetohydrodynamics in [309].

In their original formulation [288, 290, 319], a high order time approximation is obtained via a Taylor expansion of the solution, computing all time derivatives as a function of space derivatives only, by means of the Cauchy–Kovalevskaya procedure. This approach was at the time already in use for the construction of second order schemes, most notably the Lax–Wendroff scheme [211] and the MUSCL–Hancock method due to van Leer [326, 327].

In their modern variant, ADER method hinge on an important building block, called *local space-time Discontinuous Galerkin predictor* [105, 119], which constructs a space-time polynomial preliminary solution of the governing equation starting exclusively from a cell local spatial high order polynomial representation of the data. Unlike the Cauchy–Kovalevskaya procedure, the local space-time Discontinuous Galerkin predictor is entirely PDE-independent and greatly expands the applicability of ADER schemes to novel mathematical models.

A recent review of the history of the development of ADER schemes can be found in [59]. For details on some of the first fully discrete one-step Lax-Wendroff-type time discretisations proposed for DG methods, see [119, 154, 220, 267].

In the following Sections we provide the most important details about the ADER methodology, first for fixed Cartesian grids in 3.1, and then for Arbitrary-Lagrangian-Eulerian ADER schemes on moving polygonal grids with dynamic connectivity.

3.1 ADER Finite Volume and Discontinuous Galerkin schemes on Fixed Cartesian meshes

In this section, we summarise the key elements of the numerical methods employed in this work, which are the high order ADER Discontinuous Galerkin ($\mathbb{P}_N\mathbb{P}_N$) and ADER-WENO Finite Volume ($\mathbb{P}_0\mathbb{P}_M$) schemes with *a posteriori* subcell limiting. These methods can be applied to general nonconservative hyperbolic systems of balance laws. ADER Discontinuous Galerkin schemes for nonconservative hyperbolic systems have been introduced in [108, 116, 273], following the path-conservative approach of Castro and Parés, originally developed for the Finite Volume framework [68, 249] and based on the theory of Dal Maso, Le Floch and Murat [232].

Modern explicit ADER schemes follow a fully discrete predictor-corrector procedure, which can be regarded as a high order extension of the simple and successful MUSCL–Hancock approach [317, 326, 327], rather than using a semi-discrete formulation in conjunction with multi-stage timestepping, as in Runge–Kutta DG methods [89].

At each timestep, a cell-local space-time predictor solution $\mathbf{q}_h(\mathbf{x},\ t)$ is computed either from a piecewise polynomial reconstruction $\mathbf{w}_h(\mathbf{x})$ of cell average data (for FV methods), or is directly available from the evolved piecewise polynomial data $\mathbf{w}_h(\mathbf{x}) = \mathbf{u}_h(\mathbf{x})$ for DG methods.

The predictor step evolves the polynomial data locally within each cell to obtain an approximate space-time polynomial solution pertaining to that cell, without taking into account any coupling with neighbouring cells.

Then, in a second (corrector) step, volume integrals arising from a weak formulation of the differential problem

$$\partial_t \mathbf{Q} + \nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q})\ \nabla \mathbf{Q} = \mathbf{S}(\mathbf{Q}), \tag{3.1}$$

can be easily evaluated with the aid of appropriate quadrature formulas, and quadrature at space-time faces are used to compute averaged numerical fluxes corresponding to the Riemann problems arising from extrapolation of the predictor solution from adjacent cells. Since the two families of schemes (FV and DG) use the same discrete data representation (nodal degrees of freedom of a Gauss–Legendre–Lagrange polynomial), the space-time Discontinuous Galerkin predictor can be formulated in full generality for both, or even for the more general family of $\mathbb{P}_N\mathbb{P}_M$ schemes [102, 105, 151].

To clarify, the ADER $\mathbb{P}_N\mathbb{P}_M$ family includes, as a special case, ADER $\mathbb{P}_N\mathbb{P}_N$ Discontinuous Galerkin schemes (ADER-DG for short), which are a one-step fully discrete formulation of the widely used Discontinuous Galerkin methods.

These methods evolve in time data that is piecewise polynomial (of uniform arbitrary degree N in this case).

On the other hand of the spectrum of possible $\mathbb{P}_N\mathbb{P}_M$ methods, one can find high order Finite Volume schemes $\mathbb{P}_0\mathbb{P}_M$ or ADER-FV, where data is assumed to be piecewise constant ($N = 0$) over a set of control volumes, and a data reconstruction is operated in order to obtain a piecewise polynomial approximant for each one of the control volume. As stated above, the well known MUSCL–Hancock scheme can be seen as a particular case of second-order ADER $\mathbb{P}_0\mathbb{P}_1$ Finite Volume method, with a space-time continuous Galerkin predictor generating the space-time face midpoint values that are eventually used in flux computations.

In the general case, a $\mathbb{P}_N\mathbb{P}_M$ scheme will represent data in each control volume as polynomials of order N, and apply a reconstruction operator to obtain a new set of polynomials of higher degree $M > N$. In all cases, at any discrete time level t^n, a polynomial representation of the data is available, then the $\mathbb{P}_N\mathbb{P}_M$ formalism prescribes that a space-time polynomial approximant to the solution be computed in each control volume/mesh element, independently of the origin of the spatially high order data (Discontinuous Galerkin degrees of freedom or Finite Volume reconstruction in particular). This means that the space-time predictor step is carried out exactly in the same manner regardless of whether the polynomial data originated from a reconstruction, or not.

From the space-time predictor solution one can immediately compute all the volume integrals appearing in the fully discrete, one-step update formulas (3.26); in particular, this operation can be carried out quite conveniently thanks to the choice of employing a nodal basis where the nodal values are located at the Gauss–Legendre quadrature nodes and the basis functions $\varphi_m(\boldsymbol{\xi}) = \psi_{n_x}(\xi)\,\psi_{n_y}(\eta)\,\psi_{n_z}(\zeta)$ are two-dimensional or three-dimensional tensor products of the Lagrange polynomials interpolating the Gauss–Legendre quadrature nodes.

In this work, spurious oscillations that typically occur when employing higher than first order linear schemes, see [162], are minimised as follows: in the case of Finite Volume methods, we employ a nonlinear WENO reconstruction procedure, while for Discontinuous Galerkin schemes we adopt the *a posteriori* subcell Finite Volume limiting strategy [125], that is, at each timestep a candidate solution is computed without any limiter, and then afterwards, if this candidate solution violates one or more physical and numerical admissibility criteria (floating point exceptions, violation of positivity, violation of a discrete maximum principle), then it is discarded and a new discrete solution is recomputed, starting again from valid data at the previous time step. This data is obtained by projecting the DG polynomial on a fine subcell grid, or directly from the subcell average representation of the data if it was already available at the previous timestep. Afterwards, the discrete solution is reconstructed back from subcell averages to a DG polynomial.

3.1.1 Data representation and notation

The computational domain is partitioned in conforming Cartesian elements

$$\Omega_{ijk} = \left[x_i - \frac{\Delta x_i}{2},\, x_i + \frac{\Delta x_i}{2}\right] \times \left[y_j - \frac{\Delta y_j}{2},\, y_j + \frac{\Delta y_j}{2}\right] \times \left[z_k - \frac{\Delta z_k}{2},\, z_k + \frac{\Delta z_k}{2}\right], \tag{3.2}$$

and for each element a reference frame of coordinates is defined by

$$\xi = \frac{x - x_i}{\Delta x_i} + \frac{1}{2}, \qquad \eta = \frac{y - y_j}{\Delta y_j} + \frac{1}{2}, \qquad \zeta = \frac{x - z_k}{\Delta z_k} + \frac{1}{2}. \tag{3.3}$$

Discrete data is given as the degrees of freedom of a d-dimensional polynomial (in this exposition we will use $d = 3$ without loss of generality) represented by means of a set of nodal basis functions in the form of three-dimensional tensor products of the Lagrange polynomials $\psi_{n_x}(\xi)$, $\psi_{n_y}(\eta)$, and $\psi_{n_z}(\zeta)$ satisfying, at Gauss–Legendre quadrature node locations $\xi^{\mathrm{GL}}_{p_x}$, $\eta^{\mathrm{GL}}_{p_y}$, and $\zeta^{\mathrm{GL}}_{p_z}$, the interpolation conditions $\psi_{n_x}(\xi^{\mathrm{GL}}_{p_x}) = \delta_{m_x\,p_x}$, $\psi_{n_y}(\eta^{\mathrm{GL}}_{p_y}) = \delta_{m_y\,p_y}$, and $\psi_{n_z}(\zeta^{\mathrm{GL}}_{p_z}) = \delta_{m_z\,p_z}$, with δ_{ij} being the usual Kronecker symbol.

Throughout this Section we will use a compact multi-index notation so that the three-dimensional position of the generic Gauss–Legendre quadrature node of index p is written $\boldsymbol{\xi}^{\mathrm{GL}}_p = (\xi^{\mathrm{GL}}_{p_x},\ \eta^{\mathrm{GL}}_{p_y},\ \zeta^{\mathrm{GL}}_{p_z})$ and the three dimensional basis function of index m can be expressed as $\varphi_m(\boldsymbol{\xi}) = \psi_{n_x}(\xi)\,\psi_{n_y}(\eta)\,\psi_{n_z}(\zeta)$. Note that the interpolation property can be written in an entirely analogous fashion with respect to the one-dimensional case, that is, with the notation $\varphi_m(\boldsymbol{\xi}^{\mathrm{GL}}_p) = \delta_{mp}$.

Within each cell Ω_{ijk}, at a given time $t = t^n$, the discrete solution is then written (dropping for convenience of notation the cell indices i, j, k) as a polynomial of order M in each direction

$$\begin{aligned} \mathbf{u}_h(\mathbf{x},\ t^n) = \mathbf{u}_h(\boldsymbol{\xi}(\mathbf{x}),\ t^n) &= \sum_{m_x=1}^{M+1} \sum_{m_y=1}^{M+1} \sum_{m_z=1}^{M+1} \psi_{n_x}(\xi)\,\psi_{n_y}(\eta)\,\psi_{n_z}(\zeta)\,\hat{\mathbf{u}}^n_{m_x\,m_y\,m_z} \\ &= \sum_{m=1}^{(M+1)^d} \varphi_m(\boldsymbol{\xi})\,\hat{\mathbf{u}}^n_m. \end{aligned} \tag{3.4}$$

Furthermore, we will use the Einstein summation convention over repeated indices so that the discrete solution can be expressed as $\mathbf{u}_h(\mathbf{x},\ t^n) = \varphi_m(\boldsymbol{\xi})\,\hat{\mathbf{u}}_m^n$. The Finite Volume data representation (cell average values $\mathbf{Q}_{ijk}^n$) can be regarded as a special case of (3.4) in which $M = 0$ and the single basis function is the constant function $\varphi_m^{\mathrm{FV}}(\boldsymbol{\xi}) = 1$ within each element.

3.1.2 Polynomial WENO reconstruction

In order to obtain a high order data reconstruction from Finite Volume cell averages, we employ a *full polynomial* WENO reconstruction, introduced in [117] for unstructured meshes and employed, for example, in [124] on Cartesian grids. The most prominent difference between this approach and the original formulation by Jiang and Shu [199] is that instead of computing *pointwise* values of the conserved variables at the aid of optimal linear weights, here we seek to obtain the degrees of freedom of the *entire reconstruction polynomial*, to be used in the computation of fluxes, non-conservative products and source terms via high order quadrature formulae. At this point we would also like to stress that entire WENO reconstruction polynomials with a reconstruction stencil of *optimal compactness* can be achieved via the elegant CWENO approach forwarded by Puppo, Russo, Semplice *et al* in [92, 107, 213, 215, 291]. The first step is to define, for a generic element $\Omega_{ijk} = [x_{i-1/2},\ x_{i+1/2}]\times[y_{j-1/2},\ y_{j+1/2}]\times[z_{k-1/2},\ z_{k+1/2}]$, the three sets of reconstruction stencils, one for each space dimension. Each stencil will be identified by the triplet of subscripts $(i,\ j,\ k)$ of the cell in which the reconstruction is sought, together with the superscript describing the spatial direction (x, y or z) of the reconstruction and an integer superscript s that identifies the specific stencil in the set. The three generic elements of such reconstruction stencil sets will be then written as

$$S_{ijk}^{x,\,s} = \bigcup_{m=i-L}^{i+R} \Omega_{mjk}, \qquad S_{ijk}^{y,\,s} = \bigcup_{m=j-L}^{j+R} \Omega_{imk}, \qquad S_{ijk}^{z,\,s} = \bigcup_{m=k-L}^{k+R} \Omega_{ijm}, \tag{3.5}$$

where $L = L(M,\ s)$ and $R = R(M,\ s)$ are positive integers representing the number of elements in the stencil respectively to the left and to the right of the principal cell. In each space direction, for even values of the the reconstruction degree M, one always has $N_{\mathrm{s}} = 3$ stencils, one central and two off-centre, with left and right extensions given by

$$L(M,\ s) = \begin{cases} M/2 & \text{if } s = 1, \\ M & \text{if } s = 2, \\ 0 & \text{if } s = 3, \end{cases} \quad R(M,\ s) = \begin{cases} M/2 & \text{if } s = 1, \\ 0 & \text{if } s = 2, \\ M & \text{if } s = 3, \end{cases} \tag{3.6}$$

while for odd values of M we define $N_{\mathrm{s}} = 4$ types of stencils, two central, two off-centre, having extensions

$$L(M,\ s) = \begin{cases} (M+1)/2 & \text{if } s=1, \\ (M-1)/2 & \text{if } s=2, \\ M & \text{if } s=3, \\ 0 & \text{if } s=4, \end{cases} \qquad R(M,\ s) = \begin{cases} (M-1)/2 & \text{if } s=1, \\ (M+1)/2 & \text{if } s=2, \\ 0 & \text{if } s=3, \\ M & \text{if } s=4. \end{cases} \tag{3.7}$$

This choice ensures that each stencil be composed by a number of elements equal to the nominal order of the scheme, which is $M+1$.

The dimension-by-dimension reconstruction is carried out by repeated application (over each space dimension) of a one-dimensional-sweep procedure which constructs, in each cell, the $M+1$ degrees of freedom of a polynomial of degree M, first solving a set of linear reconstruction equations (imposing conservation of cell averages on each element of a given stencil), and then combining the solutions of the reconstruction equations in a data-dependent, nonlinear fashion in order to ensure the non-oscillatory character of the reconstructed polynomials.

In the generic element Ω_{ijk}, the reconstruction polynomials obtained in each of the three subsequent passes are expressed in terms of their degrees of freedom $\hat{\mathbf{w}}^{1\mathrm{D}}_{ijk,p}$, $\hat{\mathbf{w}}^{2\mathrm{D}}_{ijk,pq}$, and $\hat{\mathbf{w}}^{3\mathrm{D}}_{ijk,pqr}$, relative to a tensor-product-type Gauss–Legendre–Lagrange basis function. The degrees of freedom and thus the polynomials are obtained as nonlinear convex combinations, written as

$$\begin{aligned} &\mathbf{w}_h^{1\mathrm{D}}(x) = \psi_p(\xi)\,\hat{\mathbf{w}}^{1\mathrm{D}}_{ijk,\,p}, && \text{with } \hat{\mathbf{w}}^{1\mathrm{D}}_{ijk,\,p} = \omega_s^{1\mathrm{D}}\,\hat{\mathbf{w}}^{1\mathrm{D},\,s}_{ijk,\,p}, \\ &\mathbf{w}_h^{2\mathrm{D}}(x,y) = \psi_p(\xi)\,\psi_q(\eta)\,\hat{\mathbf{w}}^{2\mathrm{D}}_{ijk,\,pq}, && \text{with } \hat{\mathbf{w}}^{2\mathrm{D}}_{ijk,\,pq} = \omega_s^{2\mathrm{D}}\,\hat{\mathbf{w}}^{2\mathrm{D},\,s}_{ijk,\,pq}, \\ &\mathbf{w}_h^{3\mathrm{D}}(x,y,z) = \psi_p(\xi)\,\psi_q(\eta)\,\psi_r(\zeta)\,\hat{\mathbf{w}}^{3\mathrm{D}}_{ijk,\,pqr}, && \text{with } \hat{\mathbf{w}}^{3\mathrm{D}}_{ijk,\,pqr} = \omega_s^{3\mathrm{D}}\,\hat{\mathbf{w}}^{3\mathrm{D},\,s}_{ijk,\,pqr}, \end{aligned} \tag{3.8}$$

having defined, for each stencil $S^{x,\,s}_{ijk}$, $S^{y,\,s}_{ijk}$, and, $S^{z,\,s}_{ijk}$ the linear (in the sense that they are not affected by limiting) reconstruction polynomials computed from the solution of the reconstruction equations

$$\mathbf{w}_h^{1\mathrm{D},\,s}(x) = \psi_p(\xi)\,\hat{\mathbf{w}}^{1\mathrm{D},\,s}_{ijk,\,p}, \tag{3.9}$$

$$\mathbf{w}_h^{2\mathrm{D},\,s}(x,y) = \psi_p(\xi)\,\psi_q(\eta)\,\hat{\mathbf{w}}^{2\mathrm{D},\,s}_{ijk,pq}, \tag{3.10}$$

$$\mathbf{w}_h^{3\mathrm{D},\,s}(x,y,z) = \psi_p(\xi)\,\psi_q(\eta)\,\psi_r(\zeta)\,\hat{\mathbf{w}}^{3\mathrm{D},\,s}_{ijk,\,pqr}, \tag{3.11}$$

where the indices p, q, and r run from 0 to M (covering the number of degrees of freedom to be reconstructed in the in each space dimension) and where we adopted Einstein's summation convention over repeated indices.

Each pass is analogous to the first one, in that a one-dimensional stencil is used and only a one-dimensional oscillation indicator has to be computed, but it must be remarked that, as a result of performing the nonlinear stencil selection procedure in a given dimension before operating the linear reconstructions in the next one, each one of the final three-dimensional degrees of freedom is subject to composite limiting in the three space dimensions, which includes information not only from the direct face neighbours, but from the node neighbours as well, and the reconstruction is thus genuinely multi-dimensional. For alternative and more efficient multi-dimensional finite volume WENO reconstructions on Cartesian meshes, see [56, 57].

More in detail, first, one constructs for each cell a one-dimensional polynomial $\mathbf{w}_h^{1\mathrm{D}}(x)$, while maintaining the data in the remaining directions (y and z) in piecewise constant cell-averaged form.

The linear reconstruction equations, enforcing integral conservation on all elements in the stencil $S_{ijk}^{x,\,s}$, constitute a linear system whose solutions are the $M+1$ unknown degrees of freedom $\hat{\mathbf{w}}_{ijk,\,p}^{1\mathrm{D},s}$, and in the first space dimension read

$$\frac{1}{\Delta x_m}\int_{x_{m-\frac{1}{2}}}^{x_{m+\frac{1}{2}}} \psi_p(\xi(x))\,\hat{\mathbf{w}}_{ijk,\,p}^{1\,\mathrm{D},\,s}\,\mathrm{d}x = \mathbf{Q}_{mjk}^n, \quad \forall\,\Omega_{mjk}\in S_{ijk}^{x,\,s}. \tag{3.12}$$

Then the nonlinear coefficients for the combination of the polynomials obtained from each stencil are computed at each line sweep as

$$\omega_s^{n\mathrm{D}} = \tilde{\omega}_s^{n\mathrm{D}}\left(\sum_{s=1}^{N_s}\tilde{\omega}_s^{n\mathrm{D}}\right)^{-1}, \quad \text{with} \quad \tilde{\omega}_s^{n\mathrm{D}} = \lambda_s\,(\sigma_s^{n\mathrm{D}}+\varepsilon)^{-r}, \tag{3.13}$$

where the oscillation indicator $\sigma_s^{n\mathrm{D}}$ is defined as

$$\sigma_s^{n\mathrm{D}} = \sum_{l=0}^{M}\sum_{m=0}^{M}\Sigma_{l\,m}\,\frac{\hat{\mathbf{w}}_{ijk,\,l}^{n\mathrm{D},\,s}}{\mathbf{w}_0}\,\frac{\hat{\mathbf{w}}_{ijk,\,m}^{n\mathrm{D},\,s}}{\mathbf{w}_0}, \text{ with } \Sigma_{l\,m} = \sum_{\alpha=1}^{M}\int_0^1 \frac{\partial^\alpha\psi_l(\xi)}{\partial\xi^\alpha}\,\frac{\partial^\alpha\psi_m(\xi)}{\partial\xi^\alpha}\,\mathrm{d}\xi. \tag{3.14}$$

The numerical parameters used for the computation of the nonlinear weights are $\lambda_s = 1$ for off-centre stencils and $\lambda_s = 10^5$ for central stencils, and we set $\varepsilon = 10^{-14}$ and $r = 8$. An important consideration is that since the oscillation indicators are highly nonlinear, particular care should be taken in dividing the input values by an appropriate scaling factor $\mathbf{w}_0$. As a practical example, it is often the case that when using the stiffened gas equation of state, very large values for the mixture energy variable appear even in standard pressure conditions, which could lead to catastrophic loss of precision in the computation of the weights. Such a scaling factor can be computed for example as

$$\mathbf{w}_0 = \varepsilon_0 + \sum_{s} \sum_{m=0}^{M} \left| \hat{\mathbf{w}}^{n\mathrm{D},\,s}_{ijk,\,m} \right|, \tag{3.15}$$

that is, by evaluating, variable by variable, the sum of the absolute values of all the degrees of freedom of the input data over all stencils and adding a new constant $\varepsilon_0 = 10^{-14}$ to avoid division by zero.

In the second pass, one obtains data in the two-dimensional polynomial form $\mathbf{w}_h^{2\mathrm{D},\,s}(x,\ y)$, by first solving

$$\frac{1}{\Delta y_m} \int_{y_{m-\frac{1}{2}}}^{y_{m+\frac{1}{2}}} \psi_q(\eta(y))\, \hat{\mathbf{w}}^{2\mathrm{D},\,s}_{ijk,\,pq}\ \mathrm{d}y = \hat{\mathbf{w}}^{1\mathrm{D}}_{imk,\,p}, \quad \forall\, \Omega_{imk} \in S^{y,\,s}_{ijk}, \tag{3.16}$$

for each degree of freedom $\hat{\mathbf{w}}^{1\mathrm{D}}_{imk,\,p}$ and then carrying out the nonlinear selection as in the first pass. Analogously, in the third space dimension, conservation over each element of the stencil gives

$$\frac{1}{\Delta z_m} \int_{z_{m-\frac{1}{2}}}^{z_{m+\frac{1}{2}}} \psi_r(\zeta(z))\, \hat{\mathbf{w}}^{3\mathrm{D},\,s}_{ijk,\,pqr}\ \mathrm{d}z = \hat{\mathbf{w}}^{2\mathrm{D}}_{ijm,\,pq}, \quad \forall\, \Omega_{ijm} \in S^{z,\,s}_{ijk}, \tag{3.17}$$

to be solved once for each degree of freedom $\hat{\mathbf{w}}^{2\mathrm{D}}_{ijm,\,pq}$, and finally one obtains the sought three-dimensional weighted essentially-non-oscillatory polynomial after the nonlinear combination of the individual stencil polynomials has been applied.

3.1.3 Reconstruction in primitive variables

In this work, we employ a primitive variable reconstruction in order to better treat some of the peculiar issues that are typically encountered in the numerical solution of multiphase flow models, namely the presence of a complex, volume fraction-dependent equation of state and/or other issues due to different material interfaces evolving separately, already reported in [1] and [209]: this separation between interfaces might give rise to nonphysical discontinuities in the velocity and density fields as well as positivity violations in the mass fraction. The use of a primitive variable reconstruction for the TVD second order MUSCL–Hancock scheme, or the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ scheme used in the troubled elements as subcell limiter schemes was found to significantly mitigate these problems.

The primitive variable reconstruction procedure used for the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ limiter was introduced in [335], along with a predictor step formulated in terms of the primitive form of the governing equations. The reconstruction is performed as follows: first, a conservative polynomial WENO reconstruction is computed and the polynomials obtained from this step are *evaluated* at the cell centres so to obtain high order accurate point values for the conserved variables. Then one can convert the point values of conserved variables to primitive variables and perform a second WENO reconstruction to achieve a high order polynomial reconstruction of the primitive data. This second reconstruction step repeats the same steps described in Section 3.1.2, with the difference that the reconstruction equations are not based on directly enforcing conservation on a stencil, but rather they are obtained by requiring that the primitive variable reconstruction polynomials interpolate the cell centre value where the conversion from conserved to primitive variables has taken place. For alternative high order WENO reconstructions in primitive variables, see [20, 261].

3.1.4 Space-time Discontinuous Galerkin predictor on fixed Cartesian grids

We now describe the procedure to obtain the space-time predictor polynomials, which are defined for tensor product Cartesian basis functions as

$$\begin{aligned}\mathbf{q}_h(\xi,\ \eta,\ \zeta,\ \tau) &= \psi_{n_x}(\xi)\,\psi_{n_y}(\eta)\,\psi_{n_z}(\zeta)\,\psi_{n_t}(\tau)\,\hat{\mathbf{q}}_{m_x\,m_y\,m_z\,m_t} = \\ &= \vartheta_m(\xi,\ \eta,\ \zeta,\ \tau)\,\hat{\mathbf{q}}_m,\end{aligned} \tag{3.18}$$

again formally allowing referencing to the components of ϑ with mono-indexing or multi-indexing.

The first step for the local time evolution, starting from the polynomial data $\mathbf{w}_h(\mathbf{x},\ t^n)$, is to write the governing PDE (2.25) in a weak integral form in space and time as

$$\begin{aligned}&\int_{t_n}^{t_{n+1}}\int_{\Omega_i}\vartheta_p\,\partial_t\mathbf{q}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t + \int_{t_n}^{t_{n+1}}\int_{\Omega_i}\vartheta_p\,\nabla\cdot\mathbf{F}(\mathbf{q}_h)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t\ + \\ &+\int_{t_n}^{t_{n+1}}\int_{\Omega_i}\vartheta_p\,\mathbf{B}(\mathbf{q}_h)\,\nabla\mathbf{q}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \int_{t_n}^{t_{n+1}}\int_{\Omega_i}\vartheta_p\,\mathbf{S}(\mathbf{q}_h)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t,\end{aligned} \tag{3.19}$$

and then integrate by parts in time the first term. Then, by upwinding in time the value of $\mathbf{q}_h(\mathbf{x},\ t^n)$ from the reconstruction polynomial $\mathbf{w}_h(\mathbf{x},\ t^n)$, we can write

$$\begin{aligned}
&\int_{\Omega_i} \vartheta_p(\mathbf{x},\ t^{n+1})\,\mathbf{q}_h(\mathbf{x},\ t^{n+1})\ \mathrm{d}\mathbf{x} - \int_{\Omega_i} \vartheta_p(\mathbf{x},\ t^{n})\,\mathbf{w}_h(\mathbf{x},\ t^{n})\ \mathrm{d}\mathbf{x} + \\
&\quad - \int_{t_n}^{t_{n+1}} \int_{\Omega_i} \partial_t\vartheta_p\,\mathbf{q}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t + \int_{t_n}^{t_{n+1}} \int_{\Omega_i} \vartheta_p\,\nabla\cdot\mathbf{F}(\mathbf{q}_h)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t + \\
&\quad + \int_{t_n}^{t_{n+1}} \int_{\Omega_i} \vartheta_p\,\mathbf{B}(\mathbf{q}_h)\,\nabla\mathbf{q}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \int_{t_n}^{t_{n+1}} \int_{\Omega_i} \vartheta_p\,\mathbf{S}(\mathbf{q}_h)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t.
\end{aligned} \tag{3.20}$$

By then substituting the ansatz (3.18) in (3.20) one obtains a system of $(M+1)^{d+1}$ nonlinear algebraic equations which one can solve by means of a discrete Picard iteration with appropriate initial guess, as discussed in [59, 112, 114].

3.1.5 Predictor step in primitive variables

In conjunction with the primitive variable WENO reconstruction described in Section 3.1.2, as well as for pure ADER Discontinuous Galerkin schemes, for which primitive variable polynomials can be obtained by simply evaluating the primitive state vector in correspondence of each quadrature node location (nodal degree of freedom), we also carry out the local time evolution procedure with a primitive variable formulation, as per the methodology introduced in [335]. This variant of the local space-time predictor step is based on a primitive variable version of the governing equations, which directly evolves the primitive state vector $\mathbf{V}$, uses only gradients of the primitive variables $\nabla\mathbf{V}$ and is recovered by applying the chain rule to the governing equations in the form (3.24) to obtain

$$\frac{\partial \mathbf{V}}{\partial t} + \mathbf{C}(\mathbf{V})\,\nabla\mathbf{V} = \left(\frac{\partial \mathbf{Q}}{\partial \mathbf{V}}\right)^{-1}\mathbf{S}, \tag{3.21}$$

We now assign the notation $\mathbf{w}_h^*$ to represent the discrete reconstruction data *in primitive variables*, obtained either by the primitive variable WENO reconstruction, or by a straightforward conversion of the nodal degrees of freedom for ADER-DG schemes. Furthermore, we define $\mathbf{v}_h$ to be the discrete space-time predictor solution *in primitive variables*, and we can write a weak form of the governing equations as

$$\begin{aligned}
&\int_{t_n}^{t_{n+1}} \int_{\Omega_i} \vartheta_l\,\partial_t\mathbf{v}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t + \int_{t_n}^{t_{n+1}} \int_{\Omega_i} \vartheta_l\,\mathbf{C}(\mathbf{V})\,\nabla\mathbf{v}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \\
&\qquad = \int_{t_n}^{t_{n+1}} \int_{\Omega_i} \vartheta_l \left(\frac{\partial \mathbf{Q}}{\partial \mathbf{V}}\right)^{-1} \mathbf{S}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t,
\end{aligned} \tag{3.22}$$

and again integrating by parts in time one obtains a nonlinear algebraic system of $(M+1)^{d+1}$ equations

$$\begin{aligned}&\int_{\Omega_i} \vartheta_l(\mathbf{x},\ t^{n+1})\,\mathbf{v}_h(\mathbf{x},\ t^{n+1})\ \mathrm{d}\mathbf{x}\ + \\ &-\int_{\Omega_i} \vartheta_l(\mathbf{x},\ t^{n})\,\mathbf{w}_h^*(\mathbf{x},t^{n})\ \mathrm{d}\mathbf{x} - \int_{t_n}^{t_{n+1}}\int_{\Omega_i} \partial_t\vartheta_l\,\mathbf{v}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t\ + \\ &+\int_{t_n}^{t_{n+1}}\int_{\Omega_i} \vartheta_l\,\mathbf{C}(\mathbf{V})\,\nabla\mathbf{v}_h\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \int_{t_n}^{t_{n+1}}\int_{\Omega_i} \vartheta_l\left(\frac{\partial\mathbf{Q}}{\partial\mathbf{V}}\right)^{-1}\mathbf{S}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t,\end{aligned} \tag{3.23}$$

again to be solved via a discrete Picard iteration [112] and then extrapolated to the cell boundaries to compute the numerical fluxes and fluctuations, as well as the volume integrals of the explicit update formulas (3.26). Following this step, converting back from the polynomial space in primitive variables $\mathbf{v}_h$, to the polynomial space in conserved variables $\mathbf{q}_h$ is a very simple procedure thanks to the use of a nodal basis function, which allows projection between the two spaces by simple evaluation at the space–time coordinates where the degrees of freedom are collocated.

3.1.6 One-step, fully discrete, explicit update formulas on fixed Cartesian grids

We consider a general nonconservative hyperbolic system written as

$$\partial_t\mathbf{Q} + \nabla\cdot\mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q})\,\nabla\mathbf{Q} = \mathbf{S}(\mathbf{Q}) \tag{3.24}$$

in a space time control volume $\Omega_i \times [t^n,\ t^{n+1}]$; then we then define the differential volume element $\mathrm{d}\mathbf{x} = \mathrm{d}x\ \mathrm{d}y\ \mathrm{d}z$ for compactly writing integrals over the control volume Ω_i and the surface element $\mathrm{d}\mathbf{s}$ for compactly writing integrals over its boundary $\partial\Omega_i$. By multiplying each term of the PDE (3.24) with a test function φ_p, formally integrating over the space-time element and applying Gauss's theorem for integrating the divergence of fluxes in space, we have a weak formulation

$$\begin{aligned}&\int_{t_n}^{t_{n+1}}\int_{\Omega_i} \varphi_p\,\partial_t\mathbf{Q}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t + \int_{t_n}^{t_{n+1}}\int_{\partial\Omega_i} \varphi_p\,\mathbf{F}(\mathbf{Q})\cdot\hat{\mathbf{n}}\ \mathrm{d}\mathbf{s}\ \mathrm{d}t\ + \\ &+\int_{t_n}^{t_{n+1}}\int_{\Omega_i} \Big\{\varphi_p\,\Big[\mathbf{B}(\mathbf{Q})\,\nabla\mathbf{Q} - \mathbf{S}(\mathbf{Q})\Big] - \nabla\varphi_p\cdot\mathbf{F}(\mathbf{Q})\Big\}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \mathbf{0},\end{aligned} \tag{3.25}$$

with $\hat{\mathbf{n}}$ defined as the outward unit normal vector on the element boundary. Then, by substituting the sought polynomial solution $\mathbf{u}_h(\mathbf{x},\ t^n) = \varphi_m(\boldsymbol{\xi})\,\hat{\mathbf{u}}_m^n$, as well as the polynomials $\mathbf{q}_h(\mathbf{x},\ t) = \vartheta_m(\boldsymbol{\xi},\ \tau)\,\hat{\mathbf{q}}_m$ obtained from the local space-time predictor detailed in the previous section, we have the fully-discrete one-step update formula

$$\left(\int_{\Omega_i} \varphi_p\,\varphi_q\,\mathrm{d}\mathbf{x}\right)\left(\hat{\mathbf{u}}_q^{n+1} - \hat{\mathbf{u}}_q^{n}\right) + \int_{t_n}^{t_{n+1}}\int_{\partial\Omega_i} \varphi_p\,\mathbf{F}_{\mathrm{RP}}\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+\right)\cdot\hat{\mathbf{n}}\ \mathrm{d}\mathbf{s}\ \mathrm{d}t\ +$$
$$\int_{t_n}^{t_{n+1}}\int_{\partial\Omega_i} \varphi_p\,\mathbf{D}_\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+\right)\cdot\hat{\mathbf{n}}\ \mathrm{d}\mathbf{s}\ \mathrm{d}t - \int_{t_n}^{t_{n+1}}\int_{\Omega_i^\circ} \nabla\varphi_p\cdot\mathbf{F}(\mathbf{q}_h)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t\ +$$
$$+\int_{t_n}^{t_{n+1}}\int_{\Omega_i^\circ} \varphi_p\ \left[\mathbf{B}(\mathbf{q}_h)\,\nabla\mathbf{q}_h - \mathbf{S}(\mathbf{q}_h)\right]\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \mathbf{0}, \tag{3.26}$$

with $\Omega_i^\circ = \Omega_i \setminus \partial\Omega_i$ and where we denoted with $\mathbf{F}_{\mathrm{RP}}$ the generic numerical flux function, that would be, for this work, the Rusanov flux (3.34) or the HLL flux (3.31), but also other approximate Riemann solvers could be used, such as the generalized Osher and HLLEM methods forwarded in [103, 123]. Analogously, we define the path-conservative fluctuation term as

$$\mathbf{D}_\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+\right)\cdot\hat{\mathbf{n}} = \omega\int_0^1 \mathbf{B}\left[\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+,\ s\right)\right]\cdot\hat{\mathbf{n}}\ \mathrm{d}s\ \left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right), \tag{3.27}$$

in which $\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+,\ s\right) = \mathbf{q}_h^- + s\,(\mathbf{q}_h^+ - \mathbf{q}_h^-)$ is a simple segment path function connecting the left and right states, and the path integral can be computed with a three-point Gauss–Legendre quadrature regardless of the order of the scheme. The coefficients ω must be chosen so to enforce the consistency condition [68, 249]

$$\mathbf{D}_\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+\right)\cdot\hat{\mathbf{n}} - \mathbf{D}_\Psi\left(\mathbf{q}_h^+,\ \mathbf{q}_h^-\right)\cdot\hat{\mathbf{n}} = \int_0^1 \mathbf{B}\left[\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+,\ s\right)\right]\cdot\hat{\mathbf{n}}\,\frac{\partial\Psi}{\partial s}\ \mathrm{d}s \tag{3.28}$$

and simple expressions are provided in Sections 3.1.7 and 3.1.7 for the HLL and Rusanov fluxes. The inversion of the mass matrix integrating the products $\varphi_p\,\varphi_q$ is trivial, as the choice of basis yields an orthogonal basis and thus a diagonal mass matrix. The volume integrals appearing in (3.26) may be directly evaluated by Gauss–Legendre quadrature using the nodes on which the degrees of freedom of the space-time predictor solution $\mathbf{q}_h$ are defined, while for face integrals one has to extrapolate $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$ from two adjacent cells onto the Gauss–Legendre quadrature nodes at a face, then evaluate the two-state numerical fluxes at each one of the quadrature nodes, and finally operate the weighted sum of all the numerical fluxes.

Since numerical flux functions can be in principle computationally quite expensive, an attractive alternative choice for the integration of fluxes at space-time cell boundaries, with respect to the tensor-product quadrature rule, is the following: during the space-time predictor step, automatically a polynomial approximation of the physical fluxes $\mathbf{f}_h = \mathbf{f}(\mathbf{q}_h)$ is computed within each cell. When performing the extrapolation of $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$ to the space-time boundaries, one may also directly extrapolate the approximation of the physical fluxes to the boundaries, obtaining thus at each space-time cell boundary $\mathbf{f}_h^{\text{L}}$ and $\mathbf{f}_h^{\text{R}}$.

Here we denoted with $\mathbf{f}$ the projection of the physical flux tensor $\mathbf{F}$ on one of the three canonical basis vectors indicating the orientation of the face-normal onto which the flux is to be extrapolated, that is $\mathbf{f} = \mathbf{F} \cdot \hat{\mathbf{e}}_x$, $\mathbf{g} = \mathbf{F} \cdot \hat{\mathbf{e}}_y$, or $\mathbf{h} = \mathbf{F} \cdot \hat{\mathbf{e}}_z$ in the first, second, or third direction, respectively.

Then one can treat $\mathbf{q}_h^-$, $\mathbf{q}_h^+$, $\mathbf{f}_h^{\text{L}}$, and $\mathbf{f}_h^{\text{R}}$ as four independent variables and recognise that the numerical fluxes employed in the present work can be seen, if wavespeed estimates are considered fixed, as split into a centred part (solely function of $\mathbf{f}_h^{\text{L}}$ and $\mathbf{f}_h^{\text{R}}$) and a diffusive part (function of $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$). Moreover, such a four-variable numerical flux with fixed wavespeed estimates is linear in its arguments and in order to exploit this property, the coefficients may be evaluated only once at a space-time-face-averaged state and employed for all space-time face integration points. Thanks to the simple choice of a linear segment path, one can apply the same approach to the computation of the path integral of nonconservative products, and compute the average nonconservative product coefficient matrix

$$\mathbf{B}_{\boldsymbol{\Psi}} = \int_0^1 \mathbf{B}\left[\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+,\ s\right)\right] \cdot \hat{\mathbf{n}}\ \mathrm{d}s \tag{3.29}$$

only once, integrating between the averaged states at the two faces, then multiplying (3.29) by the two weights ω_{LR} and ω_{RL} and by the space-time-face average jump between $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$, yielding $\mathbf{D}_\Psi\left(\mathbf{q}_h^-,\ \mathbf{q}_h^+\right)\cdot\hat{\mathbf{n}}$ and $\mathbf{D}_\Psi\left(\mathbf{q}_h^+,\ \mathbf{q}_h^-\right)\cdot\hat{\mathbf{n}}$, respectively.

This means that only one nonlinear computation of the wavespeed estimates and other nonlinearities in the Riemann solver has to be performed (with the face-averaged state of $\mathbf{q}_h$), while the central part of the flux can be integrated directly, as well as the jump term in conserved variables. An added benefit of this approach is that the scheme need not to retain information regarding the space-time degrees of freedom of the predictor solution, making it possible and easy to implement low-storage schemes that are of uniform arbitrary high order in space and time.

Finally, in order to guarantee stability of the explicit timestepping, in this work we restrict the timestep size by

$$\Delta t = k_{\text{CFL}}\, \frac{k_N \min\left(\Delta x,\ \Delta y,\ \Delta z\right)}{d\, \lambda_{\max}}, \tag{3.30}$$

with N being the degree of the piecewise polynomial data representation, d the number of space dimensions, and $\lambda_{\max}$ the maximum absolute value of all eigenvalues found in the domain (more specifically, searching over all the quadrature nodes, i.e. where the degrees of freedom of the nodal basis are collocated). With $k_{\text{CFL}} \leq 1$ we denote a Courant-type number [91] that is typically chosen as $k_{\text{CFL}} = 0.9$ for all the simulations presented in this work. The function k_N was defined by numerical Von Neumann stability analysis in [105] for polynomials of degree up to four, while for higher values of N, we refer to an experimental determination based on numerical tests with linear advection.

The first five values of k_N are 1.0, 0.33, 0.17, 0.10, and 0.069 starting from Finite Volume ($N = 0$) up to $N = 4$ (fifth order $\mathbb{P}_4\mathbb{P}_4$ ADER-DG scheme), while for $5 \leq N \leq 9$ one can find, with numerical experiments on the linear advection equation, the coefficients $k_5 = 0.045$, $k_6 = 0.038$, $k_7 = 0.03$, $k_8 = 0.02$, $k_9 = 0.01$. We conclude by pointing out that condition (3.30) follows the same behaviour of the common $\Delta t_{\max} \sim 1/(2\,N+1)$ hyperbola found for RKDG methods, but is slightly more restrictive. For general polygons, a similar timestep restriction is imposed, as detailed in Section 3.2.

3.1.7 Approximate numerical fluxes

The path-conservative Harten–Lax–Van Leer flux. We denote with $\mathbf{f}$, $\mathbf{g}$ and $\mathbf{h}$ the relevant projections of the physical flux tensor $\mathbf{F}$ onto the Cartesian coordinate directions, i.e. $\mathbf{f} = \mathbf{F} \cdot \hat{\mathbf{e}}_x$, $\mathbf{g} = \mathbf{F} \cdot \hat{\mathbf{e}}_y$ and $\mathbf{h} = \mathbf{F} \cdot \hat{\mathbf{e}}_z$, according to the direction normal to the face/edge along which the solution of the Riemann problem is sought. With reference to two generic input states $\mathbf{q}_{\text{L}}$ and $\mathbf{q}_{\text{R}}$, the HLL [129, 130, 176] flux reads as follows,

$$\mathbf{F}_{\text{RP}}^{\text{HLL}}\left(\mathbf{q}_{\text{L}},\ \mathbf{q}_{\text{R}}\right) = \frac{S_{\text{R}}\,\mathbf{f}\left(\mathbf{q}_{\text{L}}\right) - S_{\text{L}}\,\mathbf{f}\left(\mathbf{q}_{\text{R}}\right)}{S_{\text{R}} - S_{\text{L}}} + \frac{S_{\text{R}}\,S_{\text{L}}}{S_{\text{R}} - S_{\text{L}}}\left(\mathbf{q}_{\text{R}} - \mathbf{q}_{\text{L}}\right), \tag{3.31}$$

and we give the estimates of the minimum and maximum wave speeds as

$$\begin{aligned} S_{\text{L}} &= \min\left[0,\ \lambda_{\min}\left(\mathbf{q}_{\text{L}}\right),\ \lambda_{\min}\left(\overline{\mathbf{q}}\right)\right], \\ S_{\text{R}} &= \max\left[0,\ \lambda_{\max}\left(\mathbf{q}_{\text{R}}\right),\ \lambda_{\max}\left(\overline{\mathbf{q}}\right)\right], \end{aligned} \qquad \text{with} \quad \overline{\mathbf{q}} = \frac{1}{2}\left(\mathbf{q}_{\text{L}} + \mathbf{q}_{\text{R}}\right), \tag{3.32}$$

where $\lambda_{\min}(\mathbf{q})$ and $\lambda_{\max}(\mathbf{q})$ are functions computing, respectively, the minimum and the maximum eigenvalue of the system of equations for a given vector of conserved variables $\mathbf{q}$. Given an outward unit normal vector $\hat{\mathbf{n}}$ such that the scalar product with the positive generic direction vector $\hat{\mathbf{e}}_k$ can be either positive or negative unity, upwinding of the nonconservative terms is accounted for by setting in Eq. (3.27)

$$\omega = \frac{1}{2}\left(1 + \frac{S_{\mathrm{L}} + S_{\mathrm{R}}}{S_{\mathrm{R}} - S_{\mathrm{L}}}\,\hat{\mathbf{e}}_k \cdot \hat{\mathbf{n}}\right). \tag{3.33}$$

This means that, for a given face with jump states $\mathbf{q}_{\mathrm{L}}$ and $\mathbf{q}_{\mathrm{R}}$, in a Cartesian setting, we will compute two weights $\boldsymbol{\omega}$ to associate with the two fluctuation terms, one associated with a positive unit normal, one associated with a negative unit normal.

The Rusanov flux. The Rusanov flux is obtained from the HLL flux under the assumption that $S_{\mathrm{L}} = -S_{\max}$ and $S_{\mathrm{R}} = S_{\max}$ and can be written as

$$\mathbf{F}^{\mathrm{RUS}}_{\mathrm{RP}}\left(\mathbf{q}_{\mathrm{L}},\ \mathbf{q}_{\mathrm{R}}\right) = \frac{1}{2}\Big[\mathbf{f}\left(\mathbf{q}_{\mathrm{L}}\right) + \mathbf{f}\left(\mathbf{q}_{\mathrm{R}}\right)\Big] - \frac{1}{2}\,S_{\max}\left(\mathbf{q}_{\mathrm{R}} - \mathbf{q}_{\mathrm{L}}\right). \tag{3.34}$$

This flux only requires the computation of a single wave speed estimate which is

$$S_{\max} = \max\Big[\left|\lambda_{\min}\left(\mathbf{q}_{\mathrm{L}}\right)\right|,\ \left|\lambda_{\min}\left(\mathbf{q}_{\mathrm{R}}\right)\right|,\ \left|\lambda_{\max}\left(\mathbf{q}_{\mathrm{L}}\right)\right|,\ \left|\lambda_{\max}\left(\mathbf{q}_{\mathrm{R}}\right)\right|\Big]; \tag{3.35}$$

as for the conservative part, the nonconservative fluctuations associated with the Rusanov flux do not account for upwinding and therefore, enforcing the generalized Rankine–Hugoniot consistency condition (3.28) [68, 249] we set $\omega = 1/2$.

3.1.8 A posteriori subcell limiting

The *a posteriori* subcell limiting approach [125] consists in *first* computing a *candidate solution* $\mathbf{u}_h^*(\mathbf{x},\ t^{n+1})$ from the ADER-DG scheme, without applying any precaution for limiting spurious oscillations that are typical of high order linear methods, and subsequently verifying the admissibility of such a solution by means of a relaxed discrete maximum principle and other features that might characterise the solution as locally not valid, such as violations of the positivity of density and pressure or floating-point arithmetic exceptions. This novel *a posteriori* limiting strategy for DG schemes follows the ideas of the MOOD approach, which was forwarded by Clain and Loubère *et al* in [88, 100, 101, 221] within the Finite Volume framework. The *relaxed* discrete maximum principle (DMP) is satisfied if, for all conserved (or primitive) variables, the solution is such that

$$\min_{\mathbf{y}\in\mathcal{N}_i}\Big[\mathbf{u}_h\left(\mathbf{y},\ t^n\right)\Big] - \boldsymbol{\delta} \leq \mathbf{u}_h^*(\mathbf{x},\ t^{n+1}) \leq \max_{\mathbf{y}\in\mathcal{N}_i}\Big[\mathbf{u}_h\left(\mathbf{y},\ t^n\right)\Big] + \boldsymbol{\delta}, \quad \forall \mathbf{x} \in \Omega_i, \tag{3.36}$$

with $\boldsymbol{\delta} = \max\left(\boldsymbol{\delta}_1,\ \boldsymbol{\delta}_2\right)$ computed as the maximum of two tolerances. The first tolerance is based on the absolute magnitude of the conserved quantities

$$\boldsymbol{\delta}_1 = \varepsilon_0 + \varepsilon_1 \min_{\mathbf{y}\in\mathcal{N}_i} \Big[\left|\mathbf{u}_h\left(\mathbf{y},\ t^n\right)\right|\Big] \tag{3.37}$$

with ε_0 and ε_1 small constant parameters set to $\varepsilon_0 = 10^{-4}$ (what is regarded as an acceptable nondimensional overshoot) and $\varepsilon_1 = \varepsilon_0/1000$, the last being intended to prevent excessively restrictive requirements on the oscillations of variables which have typical magnitude much larger than unity: by choosing $\varepsilon_1 = \varepsilon_0/1000$, we are prescribing that if, for a given variable, all the values in the neighborhood $\mathcal{N}_i$ have absolute magnitude larger than 1000, then the dimensionless floor value of $\boldsymbol{\delta}_1$ for that variable will be comparable to that of unit-scaled variables, for which we have chosen $\varepsilon_0 = 10^{-4}$. This will typically be the case for the density of liquids (which can be heavier than interacting gases by a factor of 1000) or internal energy (which for water with stiffened gas equation of state can be larger than the internal energy of ideal gas air by a factor of 10^4), which otherwise might trigger the *a posteriori* limiter unnecessarily due to pure unit scaling issues.

A second tolerance is then scaled according to the variation in the conserved quantities

$$\boldsymbol{\delta}_1 = \varepsilon_2 \left\{ \max_{\mathbf{y}\in\mathcal{N}_i} \Big[\mathbf{u}_h\left(\mathbf{y},\ t^n\right)\Big] - \min_{\mathbf{y}\in\mathcal{N}_i} \Big[\mathbf{u}_h\left(\mathbf{y},\ t^n\right)\Big] \right\}, \tag{3.38}$$

with $\varepsilon_2 = 10^{-3}$, which further increases the independence of the numerical scheme from the specific problem being solved.

All of the cells where the admissibility criteria are not satisfied are marked and the data from the previous timestep is projected on a finer local Finite Volume subgrid; if a given cell was already marked during the previous timestep, such data is recovered from the subcell-average representation directly, while one must compute the local subcell averages of the polynomial data if the limiter state at the previous timestep is not available. Then the solution is recomputed with a more robust Finite Volume scheme and eventually new polynomial data for the original element is reconstructed by solving an overdetermined linear system of conservative reconstruction equations.

By projecting on the subgrid, in addition to the DG polynomial of the troubled cell, also that of the neighbouring cells, one obtains a local subgrid made of conforming Cartesian elements, where some of the elements will be *phantom cells* used only for data reconstruction and for the space-time predictor step. This way, all reconstruction operations of the Finite Volume limiter scheme are well defined also for the subcells that lie on the boundary of the corresponding main grid element, and thus the description of the subcell limiter scheme is identical to that of the main grid scheme, with the obvious difference that the Cartesian mesh is $2\,N+1$ times finer than the main grid, N being the polynomial degree of the data used in the DG method.

For the *a posteriori* subcell limiting of ADER-DG methods, our choice falls on the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ Finite Volume method, or alternatively on a robust second order MUSCL–Hancock TVD Finite Volume method [327]. When employing the MUSCL–Hancock method, we carry out the reconstruction in terms of the primitive variables, with Minmod slope limiter, to obtain a cell-local polynomial representation of the data $\mathbf{v}_h$, and then we recover the conserved variable reconstruction $\mathbf{w}_h$ by extrapolating $\mathbf{v}_h$ to the face midpoint values and then computing in each cell the associated conserved variable slopes.

The MUSCL–Hancock TVD scheme used in this work is entirely equivalent to the ADER-FV $\mathbb{P}_0\mathbb{P}_1$ method, where the predictor step is replaced by a direct application of the governing equations

$$\int_{\Omega_i^\circ} \partial_t\left(\mathbf{q}_h\right) = -\int_{\partial\Omega_i} \mathbf{F}(\mathbf{w}_h)\cdot\hat{\mathbf{n}}\;\mathrm{d}\mathbf{s} - \int_{\Omega_i^\circ} \left[\mathbf{B}(\mathbf{w}_h)\,\nabla\mathbf{w}_h - \mathbf{S}(\mathbf{w}_h)\right]\,\mathrm{d}\mathbf{x}, \tag{3.39}$$

from which an estimate for the time derivative of the space-time predictor data, $\partial_t\left(\mathbf{q}_h\right)$, is computed. The time derivative $\partial_t\left(\mathbf{q}_h\right)$ is computed by using the reconstructed data at time $t = t^n$ and all spatial integrals in (3.39) are evaluated with the midpoint rule. Then the local space-time predictor solution is written as

$$\mathbf{q}_h(\mathbf{x},\ t) = \mathbf{w}_h(\mathbf{x},\ t^n) + (t-t^n)\,\partial_t(\mathbf{q}_h), \tag{3.40}$$

and one may easily evaluate all volume integrals and face integrals appearing in (3.26) by applying the midpoint quadrature rule, thus explicitly evaluating $\mathbf{q}_h(\mathbf{x},\ t)$ at the face-centres of space-time elements and at the barycentre of each space-time control volume. The similarity between the ADER-DG schemes and their Finite Volume counterpart, specifically the presence of a discrete, cell-local, space-time predictor solution that can be evaluated anywhere within the cell, means that fluxes between limited and unlimited cells can be computed (and re-computed if necessary) as they would in an *hp*-adaptive DG method.

In this work we do not use a multiple scheme cascade approach for automatically selecting, within a given hierarchy, the least dissipative scheme, that is still stable for a given timestep, but more simply combine an unlimited $\mathbb{P}_N\mathbb{P}_N$ ADER-DG method on the main grid, with a Finite Volume method of the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ or ADER-TVD $\mathbb{P}_0\mathbb{P}_1$ (MUSCL–Hancock) type, acting on a subgrid that is $2\,N+1$ times finer than the main grid. Once the Finite Volume solution has been computed, we define it to be acceptable by default, even when non-oscillatory DG polynomial data cannot be recovered from it.

Analogously, computations that are carried out using one of these Finite Volume schemes on the main grid directly, have no additional stabilisation on top of their natural nonlinear selection of reconstruction stencils.

We refer the reader to [125] for further details on *a posteriori* subcell limiting and the MOOD approach.

3.2 Direct Arbitrary-Lagrangian–Eulerian ADER schemes

In the present Section we provide a description of the high order fully-discrete one-step predictor-corrector ADER FV-DG method obtained by generalizing the scheme first presented in [105] to moving polygonal meshes with changing connectivity.

As stated and shown in [149], connectivity changes between different time level constitute a valid alternative to remeshing [31, 222, 223] for preserving or restoring mesh quality in a Lagrangian setting. However, in the context of direct ALE schemes based on space–time control volumes C_i^n, spanning the region of space–time occupied by element Ω_i between time levels t^n, when $\Omega_i = \Omega_i^n$ and t^{n+1}, when $\Omega_i = \Omega_i^{n+1}$, an important byproduct of connectivity changes, is the appearance of degenerate, so-called *sliver* elements, which fill space–time holes in the space–time tessellation of the algorithm, for example when an element changes its shape from an hexagon to a pentagon, and require special treatment as detailed in the following.

We recall that high order of accuracy in space is provided by the piecewise polynomial data representation $\mathbf{w}_h^n$, which for $N = M > 0$ coincides with the DG polynomial, i.e. $\mathbf{w}_h^n = \mathbf{u}_h^n$, while, in the Finite Volume case ($N = 0$), $\mathbf{w}_h^n$ is obtained through a reconstruction procedure, which for unstructured grids cannot be carried out in a dimensionally split fashion, and is described in Section 3.2.2.

3.2.1 Spatial representation of the numerical solution

As for the Cartesian case, the numerical solution for the conserved quantities $\mathbf{Q}$ is represented inside each polygonal mesh element polygon Ω_i^n at the current time t^n by piecewise polynomials of degree $N \geq 0$ denoted by $\mathbf{u}_h^n(\mathbf{x},\ t^n)$,

$$\mathbf{u}_h^n(\mathbf{x}, t^n) = \sum_{k=0}^{\mathcal{N}-1} \varphi_k(\mathbf{x}, t^n)\, \hat{\mathbf{u}}_{k,i}^n := \varphi_k(\mathbf{x},\ t^n)\, \hat{\mathbf{u}}_{k,i}^n, \quad \mathbf{x} \in \Omega_i^n, \tag{3.41}$$

where $\varphi_k(\mathbf{x},\ t^n)$ are now *modal* basis functions spanning the space of polynomials of degree degree up to N. Specifically, due to the difficulty of defining a reference element in the context of moving polygonal grids with dynamic connectivity, we employ a simple Taylor monomial basis. For interesting research on nodal basis functions applied to polygonal elements see [45, 157]. In the rest of the Section we will use classical tensor index notation based on the Einstein summation convention, which implies summation over two equal indices. The total number $\mathcal{N}$ of expansion coefficients (degrees of freedom, DOFs) $\hat{\mathbf{u}}_l^n$ for the basis functions depends on the polynomial degree N and is given by $\mathcal{N} = \mathcal{L}(N,\ d)$, with

$$\mathcal{L}(N,\ d) = \frac{1}{d!}\prod_{m=1}^{d}(N+m), \tag{3.42}$$

where $d = 2$ in this Section, since we are dealing only with two-dimensional domains. As basis functions φ_k in (3.41) we employ a Taylor series of degree N in the variables $\mathbf{x} = (x,\ y)$ directly defined on the physical element Ω_i^n, expanded about its current barycenter $\mathbf{x}_i^n$ and normalized by a characteristic length h_i. Formally, such Taylor basis functions for element Ω_i can be written as

$$\varphi_k(\mathbf{x},\ t^n) = \frac{(x - x_i^n)^{p_k}}{p_k!\, h_i^{p_k}}\ \frac{(y - y_i^n)^{q_k}}{q_k!\, h_i^{q_k}}, \qquad k = 0,\ \ldots,\ \mathcal{N}-1, \qquad 0 \le p_k + q_k \le N. \tag{3.43}$$

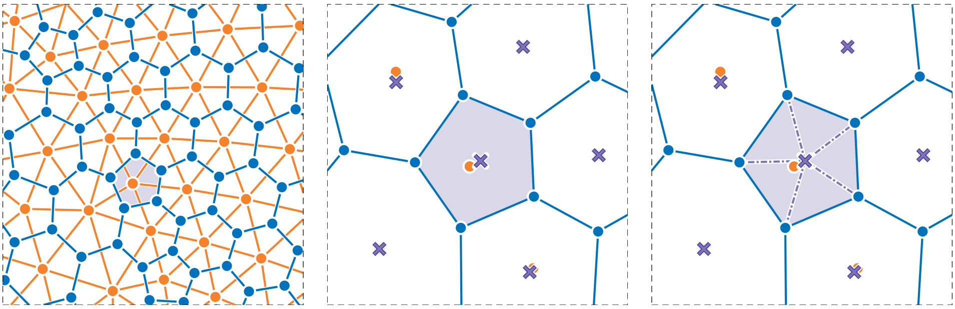


Figure 3.1. Polygonal unstructured grid. The Delaunay triangulation and the generator points are shown in orange, the barycenters of the Delaunay triangles and the dual polygonal tessellation are drawn in blue, the barycenters of the polygons are highlighted with purple crosses. Note that to each generator point corresponds a polygonal dual element which is obtained by connecting the barycenters of the triangles having this generator point as a vertex. Note also that we employ its barycenter to construct the sub–triangulation of each polygonal element (purple dotted line in the right panel).

As a length scale, h_i can be taken to be the radius of gyration of Ω_i^n, that is $h_i = \sqrt{I_{\Omega_i}/|\Omega_i|}$ where I_{Ω_i} is the second moment of area of element Ω_i and $|\Omega_i|$ its area. The unknown expansion coefficients $\hat{\mathbf{u}}_{k,i}^n$ in (3.41) can then be interpreted to be the rescaled derivatives of the Taylor expansion of the solution about $\mathbf{x}_i^n$. The time dependence of $\varphi(\mathbf{x},\ t^n)$ is directly inherited from the time-dependence of the cell barycenter $\mathbf{x}_i^n = (x_i^n,\ y_i^n)$.

The discontinuous finite element data representation (3.41) leads naturally to both a Discontinuous Galerkin (DG) scheme if $N > 0$, but also to a Finite Volume (FV) scheme in the case $N = 0$. This indeed means that for $N = 0$ we have $\varphi_k(\mathbf{x}) = 1$, with $k = 0$ and (3.41) reduces to the classical piecewise constant data representation that is typical of finite volume schemes:

$$\mathbf{u}_h^n(\mathbf{x},\ t^n)= \mathbf{Q}_i^n, \quad \mathbf{x} \in \Omega_i^n, \quad \mathbf{Q}_i^n = \frac{1}{|\Omega_i^n|} \int\limits_{\Omega_i^n} \mathbf{Q}(\mathbf{x},\ t^n)\, \mathrm{d}\mathbf{x}, \tag{3.44}$$

in which the only degree of freedom Ω_i^n is the cell average $\mathbf{Q}_n$. Note also that in the case $N > 0$ the representation given by (3.41) already provides a spatially high order accurate data representation with accuracy $N + 1$, whereas this is not the case when $N = 0$. In order to increase the spatial order of accuracy of the Finite Volume scheme, up to $M + 1$, a data reconstruction that generates a high order polynomial $\mathbf{w}_h^n(\mathbf{x},\ t^n)$ of degree $M > N$ is carried out. The reconstruction polynomial is denoted by

$$\mathbf{w}_h^n(\mathbf{x},\ t^n) = \sum_{k=0}^{\mathcal{M}-1} \psi_k(\mathbf{x},\ t^n)\, \hat{\mathbf{w}}_{k,i}^n := \psi_k(\mathbf{x},\ t^n)\, \hat{\mathbf{w}}_{k,i}^n, \tag{3.45}$$

for $\mathbf{x} \in \Omega_i^n$, with $\mathcal{M} = \mathcal{L}(M,\ d)$, and where we simply employ the same basis functions $\psi_l(\mathbf{x},\ t^n) = \varphi_l(\mathbf{x},\ t^n)$ for the reconstruction according to (3.43), but with $0 \leq k \leq \mathcal{M} - 1$ rather than $0 \leq k \leq \mathcal{N} - 1$, see also [105].

For the sake of uniform notation, in the DG case (when $M = N$), we trivially impose that the reconstruction polynomial is given by the DG polynomial, i.e. $\mathbf{w}_h^n(\mathbf{x},\ t^n) = \mathbf{u}_h^n(\mathbf{x},\ t^n)$, which implies that in the case $N = M$ the reconstruction operator is simply the identity.

3.2.2 CWENO reconstruction

For finite volume schemes ($N = 0$) the reconstruction procedure allows us to compute a high order non-oscillatory polynomial representation $\mathbf{w}_h^n(\mathbf{x},\ t^n)$ of the solution $\mathbf{Q}(\mathbf{x},\ t^n)$ for each polygon Ω_i^n, starting from the values of $\mathbf{u}_h^n(\mathbf{x},\ t^n)$ in Ω_i^n and its neighbours. It should be employed in the case $N = 0,\ M > 0$.

As already stated above, the total number of unknown degrees of freedom $\mathbf{w}_h^n(\mathbf{x},\ t^n)$ is $\mathcal{M} = \mathcal{L}(M,\ d)$, with M denoting the degree of the reconstruction polynomial $\mathbf{w}_h$.

In order to achieve high accuracy, a large stencil centered in Ω_i^n is required, but this choice produces oscillations close to discontinuities, the well-known Gibbs phenomenon. Indeed, for *linear* schemes, high order of accuracy and non-oscillatory behaviour are mutually exclusive properties [162]. In order to fulfill the requirement of non-oscillatory behaviour, a nonlinear reconstruction operator has to be adopted. For ADER-FV schemes on unstructured meshes, we rely on the CWENO reconstruction strategy first introduced in [213, 214, 216], and which can be cast in the general framework described in [92]. Here, we closely follow the work outlined in [127] for unstructured triangular and tetrahedral meshes. For the sake of completeness, we report here the entire algorithm: the differences with respect to [127] are highlighted in the last paragraph of this section.

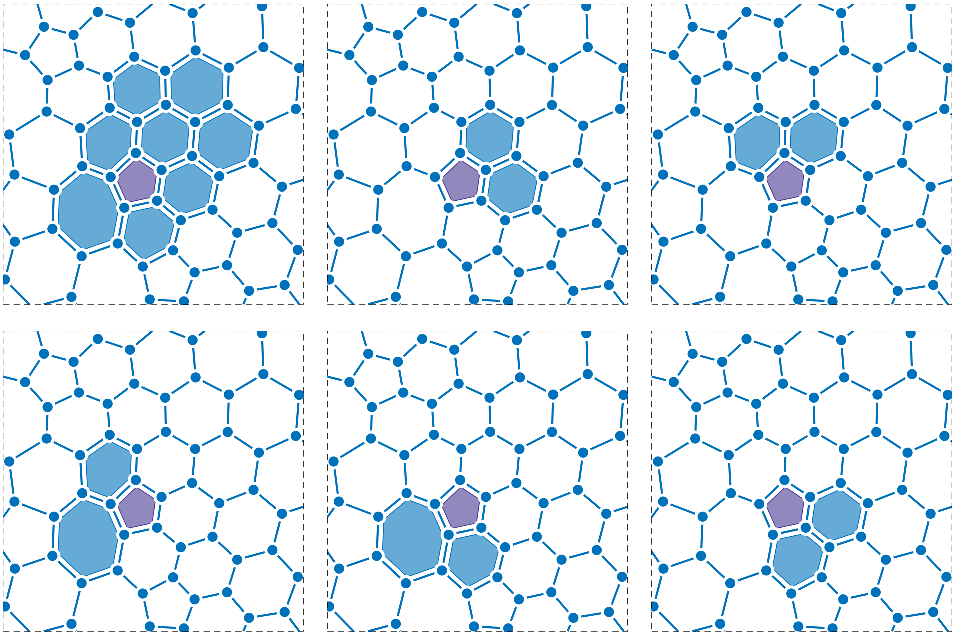


Figure 3.2. Stencils for the CWENO reconstruction of order three ($M = 2$) for a pentagonal element Ω_i^n. Top-left: central stencil made of the element itself Ω_i^n (in purple) and its neighbours (in blue). In the other panels we report the $N_s = 5$ sectorial stencils containing the element itself and two adjacent immediate neighbours.

The reconstruction starts from the computation of a so-called central polynomial $\mathbf{p}_{\mathrm{c}}$ of degree M. In order to define $\mathbf{p}_{\mathrm{c}}$ in a robust manner, following [22, 127, 202, 291], we consider a stencil $\mathcal{S}_i^0$ which is filled with a total number of $n_{\mathrm{s}} = \kappa\,\mathcal{M}$ elements, with κ a specified safety factor greater than unity, aimed at improving the conditioning of the resulting reconstruction equations. In practice we take $\kappa = 1.5$. Stencil S_i^0 includes the current polygon Ω_i^n, the first layer of neighbours of Ω_i, and is then filled by recursively adding neighbours of elements that have been already selected, until the desired number n_e is reached. The central polynomial $\mathbf{p}_{\mathrm{c}}(\mathbf{x}, t^n)$ is then defined by imposing that its average on each cell $P_{i_k}^n$ match the known cell average $\hat{\mathbf{u}}_{0,i_k}^n$. Since $n_e > \mathcal{M}$, this leads to an overdetermined linear system, which is solved using a constrained least-squares technique [117] as

$$\mathbf{p}_{\mathrm{c}}(\mathbf{x},\ t^n) = \underset{\mathbf{p}\,\in\,\mathcal{P}_i}{\operatorname{argmin}} \sum_{\Omega_k^n \in \mathcal{S}_i^0} \left(\mathbf{Q}_k^n - \frac{1}{|\Omega_k^n|} \int_{\Omega_k^n} \mathbf{p}(\mathbf{x},\ t^n)\ \mathrm{d}\mathbf{x} \right)^2 \tag{3.46}$$

with the constraint

$$\mathcal{P}_i = \left\{ \mathbf{p} \in \mathbb{P}_M : \frac{1}{|\Omega_i^n|} \int_{\Omega_i^n} \mathbf{p}(\mathbf{x},\ t^n)\ \mathrm{d}\mathbf{x} = \mathbf{Q}_i^n \right\}, \tag{3.47}$$

where $\mathbb{P}_M$ is the set of all polynomials of degree at most M. In other words, the polynomial $\mathbf{p}_{\mathrm{c}}$ has exactly the cell average $\hat{\mathbf{u}}_{0,i}^n$ on the polygon Ω_i^n and matches all the other cell averages of the remaining stencil elements in the least-square sense. The polynomial $\mathbf{p}_{\mathrm{c}}$ is expressed in terms of the basis functions (3.43) of degree M, hence

$$\mathbf{p}_{\mathrm{c}}(\mathbf{x},\ t^n) = \sum_{k=0}^{\mathcal{M}-1} \psi_k(\mathbf{x},\ t^n)\, \hat{\mathbf{p}}_{k,i}^n, \tag{3.48}$$

and the integrals appearing in (3.46) are computed in each polygon Ω_i by summing the contribution of each of the sub-triangles connecting two adjacent vertices of Ω_i with its barycenter $\mathbf{x}_i$. On the sub-triangles we employ $(M+1)^2$ quadrature points defined by the conical product of the one-dimensional Gauss–Jacobi formula, see [306].

In order to obtain an essentially non oscillatory reconstruction polynomial, the CWENO algorithm makes use of other polynomials of lower degree. In the practical implementation we simply consider a set of N_s linear (first degree) sectorial polynomials associated with N_s stencils S_i^s, $s \in 1,\ \dots,\ N_s$, each one including the central element Ω_i and two adjacent direct neighbours. An illustration of the stencils adopted S_i^0 and S_i^s for a polygon with $N_s = 5$ and $M = 2$ can be found in Figure 3.2.

For each stencil S_i^s, the linear reconstruction polynomial $\mathbf{p}_s(\mathbf{x},\ t^n)$ can be computed by interpreting the cell average values of the three stencil elements as point values collocated at the corresponding cell barycenters. For polygonal elements and linear polynomials, this is entirely equivalent to solving the full conservative reconstruction equations but with reduced computational cost. Following the general framework introduced in [92], we then select a unit-sum set of positive coefficients $\lambda_0,\ \ldots,\ \lambda_{N_s}$ such that and we define a new polynomial

$$\mathbf{p}_0(\mathbf{x},\ t^n) = \frac{1}{\lambda_0}\left(\mathbf{p}_{\mathrm{c}}(\mathbf{x},\ t^n) - \sum_{s=1}^{N_p} \lambda_s\, \mathbf{p}_s(\mathbf{x},\ t^n)\right) \in \mathbb{P}_M, \tag{3.49}$$

so that the linear combination of the polynomials $\mathbf{p}_0,\ \ldots,\ \mathbf{p}_{N_{V_i}^n}$ with coefficients $\lambda_0,\ \ldots,\ \lambda_{N_p}$ is equal to $\mathbf{p}_{\mathrm{c}}$ and conservation is ensured. Specifically, we consider the linear weights used in [117], namely $\lambda_0 = 10^5$ for $\mathcal{S}_i^0$ and $\lambda_s = 1$ for the sectorial stencils. These weights are later normalized in order to sum to unity. Finally, the sectorial polynomials $\mathbf{p}_s$ with $s \in (1,\ N_s)$ are nonlinearly combined with $\mathbf{p}_0$, as it is done also in other WENO schemes [18, 191, 199]. We thus obtain $\mathbf{w}_h(\mathbf{x}, t^n)$ in Ω_i^n as

$$\mathbf{w}_h(\mathbf{x}, t^n) = \sum_{s=0}^{N_p} \omega_s\, \mathbf{p}_s(\mathbf{x},\ t^n), \quad \mathbf{x} \in \Omega_i^n, \tag{3.50}$$

where the normalized nonlinear weights $\boldsymbol{\omega}_s$ are given by

$$\boldsymbol{\omega}_s = \tilde{\boldsymbol{\omega}}_s \left(\sum_{m=0}^{N_p} \tilde{\boldsymbol{\omega}}_m\right)^{-1}, \quad \text{with} \quad \tilde{\boldsymbol{\omega}}_s = \frac{\lambda_s}{(\boldsymbol{\sigma}_s + \varepsilon)^r}. \tag{3.51}$$

In the above expression the non-normalized weights $\tilde{\boldsymbol{\omega}}_s$ depend on the linear weights λ_s and the oscillation indicators $\boldsymbol{\sigma}_s$ with the parameters $\varepsilon = 10^{-14}$ and $r = 4$ chosen according to [117]. Note that in smooth areas, $\omega_s \simeq \lambda_s$ and then $\mathbf{w}_{h_i} \simeq \mathbf{p}_{\mathrm{c}}$, so that optimal accuracy can be recovered. On the other hand, close to a discontinuity, $\mathbf{p}_0$ and some of the low degree polynomials $\mathbf{p}_s$ would be oscillatory and have high oscillation indicators, leading to $\omega_s \simeq 0$ and in these cases only lower order non-oscillatory data are employed in $\mathbf{w}_{h_i}$, guaranteeing the non-oscillatory property of the reconstruction. The oscillation indicators $\boldsymbol{\sigma}_s$ appearing in (3.51) are simply given by

$$\boldsymbol{\sigma}_s = \sum_l \left(\hat{\mathbf{p}}_{l,i}^{n,s}\right)^2, \tag{3.52}$$

which can be seen as a first order approximation of the well known Jiang and Shu oscillation indicator [199].

The CWENO procedure adopted in this work is similar to the one presented in [127]. The adaptation to general polygonal elements is straightforward and only concerns the computation of integrals in (3.46), the number of sectorial polynomials, and the fact that basis functions are rescaled Taylor monomials referred to the physical element and not to a reference element, hence yielding a different and very simple evaluation of the oscillation indicators (3.52).

3.2.3 Evolution of the computational domain

At this point we have a high order spatial representation of the solution $\mathbf{Q}(\mathbf{x},\ t^n)$ at the current time t^n given by the polynomial $\mathbf{w}_h^n = \mathbf{w}_h(\mathbf{x},\ t^n)$ of degree M. We recall that if $N = M > 0$ then $\mathbf{w}_h^n = \mathbf{u}_h^n$; if instead $N = 0$ then $\mathbf{w}_h^n$ is obtained through the reconstruction procedure described in the previous Section 3.2.2.

By evaluating $\mathbf{w}_h^n$ at the generator points $\mathbf{x}_c^n$, i.e. $\mathbf{w}_h^n(\mathbf{x}_c^n,\ t^n)$ with (3.45), we recover the mesh local velocity $\mathbf{v}(\mathbf{x}_c^n)$, that can be used to compute the new coordinates of the generator points simply as

$$\mathbf{x}_{c_i}^{n+1} = \mathbf{x}_{c_i}^n + \Delta t\, \mathbf{v}(\mathbf{x}_{c_i}^n). \tag{3.53}$$

Note that in the direct ALE formalism, the mesh can in principle be moved with any velocity, and in particular high order convergence is achieved even if the above relation (3.53) is not integrated with an order of accuracy matching that of the main quadrature formulas of the scheme.

The Delaunay triangulation connecting the new coordinates of the generator points $\mathbf{x}_c^{n+1}$ is now recomputed, as well as the corresponding updated polygonal tessellation. Note that the only connection between the tessellations at time t^n and t^{n+1} is the number N_P of generator points (i.e. of polygonal elements) and their global numbering. Instead, the shape of each polygon is allowed to change, that is, the number of sides of any polygon may in principle change between two consecutive time levels, and consequently also the connectivity with the neighbouring elements.

This change of the grid topology is actually the strength of the present algorithm, since it allows to maintain a high mesh quality without distorted elements, as opposed to other Lagrangian schemes constrained by a fixed connectivity for which one would have to periodically activate a remeshing procedure in order to eliminate heavily tangled elements.

However, more care is needed in order to update the solution from time t^n to t^{n+1}. In particular, to obtain a high order direct ALE scheme one needs to partition the space-time region spanned by the computational domain bewteen two consecutive time levels, constructing a set of space–time control volumes. For Finite Volume schemes of order up to 2, one could avoid such a geometric space–time approach (see [248, 304]), but in order to achieve higher convergence rates, an explicitly defined partitioning of space–time is necessary in the ADER framework.

The direct Arbitrary-Lagrangian-Eulerian schemes presented in this work constitute a fairly natural extension of the approach forwarded in [39, 41, 42, 46], to a more general tessellation in space, given by polygonal cells, but more importantly with an extended taxonomy of space–time control volumes. Indeed, allowing elements to change shape between time levels gives rise to a new type of peculiar degenerate control volumes, here called *sliver* elements, which are characterized by the fact that they have zero area at both time levels t^n and t^{n+1}, but enclose a nonzero space–time volume. If the tessellation is chosen to be of Voronoi type, i.e. it is constructed by connecting the circumcenters of the dual Delaunay triangulation, then such sliver elements can be seen as a simple tool for carrying out a decomposition of the Scutoids [161] arising from extruded Voronoi diagrams into simpler prismatoids and tetrahedra, which are easier to handle in practice.

A similar approach to space–time partitioning was adopted in [269], where the authors, in order to connect meshes with topology changes (within a different framework w.r.t. the one adopted here), have introduced some pyramidal degenerate elements instead of tetrahedral sliver elements. The strategy proposed in the aforementioned reference is indeed interesting and could in principle be applied also to the framework of explicit high order direct ALE schemes. However, besides presenting the same complexities associated with sliver elements, an additional difficulty would arise, since a degeneracy would occur at the midpoint of the time step.

3.2.4 High order integration of the trajectories of the generator points

Thanks to the ALE framework adopted in this work, the mesh can in principle be moved with an arbitrary velocity, and there is not a specific necessity of moving the grid in a fully Lagrangian fashion. Nevertheless, (3.53) can also be replaced by a high order Taylor method [44, 49, 312], leading to a high order approximation of the Lagrangian trajectories of the generators points. The use of this technique improves mesh quality in vortical flows, and also improves the overall Lagrangian behaviour of the algorithm.

In what follows we detail the high order approach used for the integration of the flow trajectories.

The Taylor expansion of the new position $\mathbf{x}_{c_i}^{n+1}$ of a generator point at time t^{n+1} with respect to its position at time t^n can be written as

$$\mathbf{x}_{c_i}^{n+1} = \mathbf{x}_{c_i}^{n} + \Delta t \, \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} + \frac{\Delta t^2}{2} \, \frac{\mathrm{d}^2\mathbf{x}}{\mathrm{d}t^2} + \frac{\Delta t^3}{6} \, \frac{\mathrm{d}^3\mathbf{x}}{\mathrm{d}t^3} + \mathcal{O}(\Delta t^4), \tag{3.54}$$

which achieves third order of accuracy in time. Now, the high order time derivatives in (3.54) are replaced by high order spatial derivatives, via the Cauchy-Kovalevskaya procedure, using repeatedly the trajectory equation

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{v}(\mathbf{x}(t)), \tag{3.55}$$

and assuming a stationary velocity field one obtains

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = \mathbf{v}, \quad \frac{\mathrm{d}^2\mathbf{x}}{\mathrm{d}t^2} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t}\right) = \frac{\mathrm{d}\mathbf{v}}{\mathrm{d}\mathbf{x}} \, \frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} \tag{3.56}$$

The chain rule, as written in (3.56), can be applied iteratively to obtain the third derivative of the position

$$\frac{\mathrm{d}^3\mathbf{x}}{\mathrm{d}t^3} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\mathrm{d}^2\mathbf{x}}{\mathrm{d}t^2}\right) = \nabla(\nabla\mathbf{v}) \, \mathbf{v} \, \mathbf{v} + (\nabla\mathbf{v}) \, (\nabla\mathbf{v}) \, \mathbf{v} \tag{3.57}$$

Finally, the partial derivatives of $\mathbf{v}$ are recovered from the local fluid velocities $\mathbf{u}$ through the high order polynomials $\mathbf{w}_h$ (3.45) which represent the conserved variables $\mathbf{Q}$ inside each cell with high order of accuracy. Since $\mathbf{w}_h$ is given via modal basis functions, the coefficients $\hat{\mathbf{w}}_{k,i}^n$ already represent the values of the partial derivatives with respect to $\mathbf{x}$ of the conserved variables, if a sufficiently high order accurate $\mathbb{P}_N\mathbb{P}_M$ method is employed. Then, the chain rule is applied in order to recover the partial derivatives of the primitive variable $\mathbf{u}$ from those of the conserved variables $\rho\,\mathbf{u}$ and ρ.

3.2.5 Mesh optimization

One of the key advantages of the direct ALE framework, especially in the variant presented in this work, is that it allows superior unconstrained flexibiliy in the definition in of the mesh motion.

In particular, the choice of type of smoothing scheme is totaly unrestricted, thanks to the possibility of changing the grid connectivity between consecutive time levels.

The polygonal unstructured grids employed in this work are generated and their motion is computed by means of a Fortran code designed and implemented by the author of this thesis, based on the illuminating work of Lawson [210], Chew [84], Ruppert [281], Shewchuck [292–295], Erten and Üngör [133, 324, 325].

In this work, the mesh optimization methods are implemented by slightly modifying, at each time step, the motion of the generator points (that is, the vertexes of the dual Delaunay triangulation). This aims at improving the overall robustness of the method, as well as reducing numerical errors and spurious mesh imprinting.

In general, the target polygonal elements will have a locally uniform edge length (i.e. no small edges for a given element, smoothly graded mesh size) and will not be excessively stretched in one direction only, so that anisotropies due to differences in numerical diffusion are minimized. More importantly, this increases the robustness of the matrix computations involved in the polynomial data reconstruction and in the fully discrete update formulae. Also, we must note that these objectives shall be pursued in conjunction with the interest of preserving an accurate mesh motion that follows the local flow field, maintaining the Lagrangian character of the numerical method as far as possible. This means that one must allow a certain degree of anisotropy in the mesh, which might be desirable to resolve flow discontinuities or strong gradients.

The mesh regularization procedure begins by computing all the new positions $\mathbf{x}_{c_i}^{n+1}$ for the generator points of the polygonal grid and then recovering, for each generator, the position $\mathbf{x}_{c_i}^{*}$ that is prescribed by a simple smoothing technique applied to the candidate positions $\mathbf{x}_{c_i}^{n+1}$. We say that $\mathbf{x}_{c_i}^{*}$ is a location for the generator that is optimal in the sense of mesh quality, as opposed to optimal in following the flow of the fluid, which would be the role taken by $\mathbf{x}_{c_i}^{n+1}$. The candidate position $\mathbf{x}_{c_i}^{n+1}$ is subsequently replaced by a corrected value $\hat{\mathbf{x}}_{c_i}^{n+1}$ that is given by the weighted average $\hat{\mathbf{x}}_{c_i}^{n+1} = (1-\mu)\, \mathbf{x}_{c_i}^{n+1} + \mu\, \mathbf{x}_{c_i}^{*}$, with μ being a blending factor that yields the balance between the amount of mesh motion due to fluid flow with the one due to smoothing.

Concerning the determination of $\mathbf{x}_{c_i}^{*}$, we decided to simply compute it from the application to $\mathbf{x}_{c_i}^{n+1}$ of one iteration of a Lloyd-type algorithm; that is, after updating the Delaunay triangulation of the generator points taking into account their new candidate positions $\mathbf{x}_{c_i}^{n+1}$, we evaluate the quality-optimal position $\mathbf{x}_{c_i}^{*}$ for generator $\mathbf{c}_i$ as

$$\mathbf{x}_{c_i}^{*} = \left(\sum_{a_k \in \mathcal{A}_i} \omega_{a_k} \right)^{-1} \sum_{a_k \in \mathcal{A}_i} \frac{1}{2} \left(\mathbf{x}_{a_k,\, i_2}^{n+1} + \mathbf{x}_{a_k,\, i_3}^{n+1} \right) \omega_{a_k}. \tag{3.58}$$

We define $\mathcal{A}_i$ to indicate the set of Delaunay triangles a_k that share $\mathbf{x}_{c_i}^{n+1}$ as a vertex, while we denote with $\mathbf{x}_{a_k,\, i_2}^{n+1}$ and $\mathbf{x}_{a_k,\, i_3}^{n+1}$ the other two vertexes of the triangle a_k, that is, the two that do not coincide with $\mathbf{x}_{c_i}^{n+1}$. The choice of weights yields different smoothing methods, and in this work we mainly employ $\omega_{a_k} = \left\| \mathbf{x}_{a_k,\,i_2}^{n+1} - \mathbf{x}_{a_k,\,i_3}^{n+1} \right\|$ to obtain an algorithm that is reminiscent of Lloyd smoothing [219], as this would prescribe that each generator shall be moved to the centroid of the polygonal chain obtained by connecting all vertexes $\mathbf{x}_{a_k,\,i_2}^{n+1}$ and $\mathbf{x}_{a_k,\,i_3}^{n+1}$ of all Delaunay triangles in $\mathcal{A}_i$. This choice tends to eliminate small edges just like the algorithms forwarded in [50, 254, 296]. Alternatively, we can set $\omega_{a_k} = 1$ and obtain Laplacian smoothing [141, 180], that is, the generic generator $\mathbf{x}_{c_i}^{n+1}$ is moved to the center of mass of the system of point masses defined by the vertexes of the above described polygonal chain. Laplacian smoothing yields nicely rounded cells and tends to preserve the grading of the mesh.

Once a set of quality-aware node positions $\mathbf{x}_{c_i}^{*}$ has been determined, the algorithm must choose how to compromise between such positions and those prescribed by the fluid motion. Instead of simply fixing the value of μ as a simulation parameter, we chose to let μ vary with time by recomputing it as a function of the solution data and of the current grid configuration, as well as by accounting for the specific explicit time step restriction in use. Specifically, we compute the relaxation parameter μ as

$$\mu = \min\left(1,\ \sqrt{\frac{U_*\,\Delta t}{\Delta s}\,\mathcal{F}}\right), \tag{3.59}$$

with U_* being a rough scaling estimate for the fluid velocity, computed at each time step as the maximum velocity encountered for all generator points, Δt the time step size, and Δs an indicator for the mesh spacing, given by the minimum value of the ratio between the area and the perimeter of all polygonal elements, in analogy to how the time step duration is determined in (3.30) (with a characteristic mesh-size Δs). The underlying idea is that we want to balance, during each time step, the spatial scaling of fluid flow, with a characteristic length representative of the mesh motion due to pure smoothing in the smallest cells of the domain, which we implicitly assume to be the most delicate. In this way, we have replaced the blending factor μ with another non-dimensional *smoothing parameter* $\mathcal{F}$, that fixes the strength of smoothing in the small cells that are those that might otherwise compromise the stability of the computation. The square root is arbitrarily introduced in order to reduce the sensitivity of Equation (3.59) to sudden variations in the flow speed U_*.

Note that, although in a very approximate form, the formula (3.59) scales with the square root of a characteristic Mach number, at least when U_* is negligible with respect to sound speed or vice versa; one can verify this by substituting (3.30) (with a characteristic mesh-size Δs) in (3.59) and noting that it simplifies into an expression that includes the degree of the polynomial data N, the CFL coefficient, and an approximate Mach number. Further investigations on more complex scaling expressions that correct for such residual dependencies, as well as space-dependent formulae, are left for future work.

The results presented in this paper are obtained by moving the generators with the local fluid velocity and by applying one of the smoothing techniques described here, with different values of the smoothing parameter $\mathcal{F}$.

3.2.6 Degenerate space–time elements

Changes in control volume connectivity between two time subsequent levels induce the appearance of degenerate elements in the space–time connectivity.

As is evident from Figures 3.4b-3.4c, some of the sub-space–time control volumes used for the computation of space–time integrals, are triangular prisms with one of their top or bottom faces collapsed to just a line, and with the lateral space–time surface ∂C_{ij}^n being of triangular shape (instead of the standard quadrilateral shape). They do not pose particular problems because they are part of a standard control volume, so everything is naturally well defined on

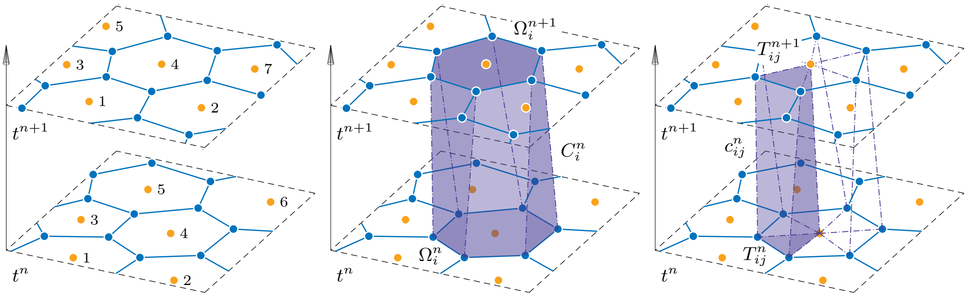


Figure 3.3. General space-time control volumes on a moving polygonal grid with fixed connectivity. (a) The tessellation at time t^n and time t^{n+1}. (b) Ω_i^n is connected with Ω_i^{n+1} to construct the space–time control volume C_i^n. (c) While for fixed Cartesian grids, the control volumes degenerate to rectangular prisms and one may apply the classic tensor product quadrature formulas to the control volume C_i^n directly, for general moving polygonal meshes a geometric decomposition of the space–time control volume is necessary. A trivial decomposition is obtained by splitting both Ω_i^n and Ω_i^{n+1} in sub-triangles T_{ij}^n and T_{ij}^{n+1}, to construct a set of space–time control volumes c_{ij}^n, which can be mapped to a reference element and integrated over.

them (basis functions, quadrature points, values of the numerical solution $\mathbf{u}_h^n$, of the reconstruction polynomials $\mathbf{w}_h^n$, and of the space–time predictor $\mathbf{q}_h^n$ defined below in (3.60)).

On the contrary, the space–time sliver element in Figure 3.4d is a completely new control volume which does not exist at time t^n, or at time t^{n+1}, since it coincides with an edge of the tessellation at the old and at the new time levels, and, as such, has zero area in space at t^n and t^{n+1}. However, it has a nonzero space–time volume. The difficulties related to this kind of elements are due to the fact that $\mathbf{w}_h$ is not clearly defined for them at time t^n and that contributions across them should not be lost at time t^{n+1}, in order to ensure conservation. Space–time sliver elements always have four neighbours, namely the two polygons that share their degenerate bottom face (edge) and the two polygons that share their degenerate top face (edge). For a detailed description of the practical implementation issues related to sliver elements we refer to [148, 149].

3.2.7 Space–time predictor on moving polygonal grids with dynamic connectivity

In what follows, a predictor of the solution is recovered, which is valid locally inside C_i^n and is given by high order piecewise space-time polynomials $\mathbf{q}_h^n(\mathbf{x},\ t)$ of degree M that are expressed as

$$\mathbf{q}_h^n(\mathbf{x},\ t) = \sum_{k=0}^{\mathcal{Q}-1} \vartheta_k(\mathbf{x},\ t)\, \hat{\mathbf{q}}_k^n, \quad (\mathbf{x},\ t) \in C_i^n, \quad \mathcal{Q} = \mathcal{L}(M,\ d+1), \tag{3.60}$$

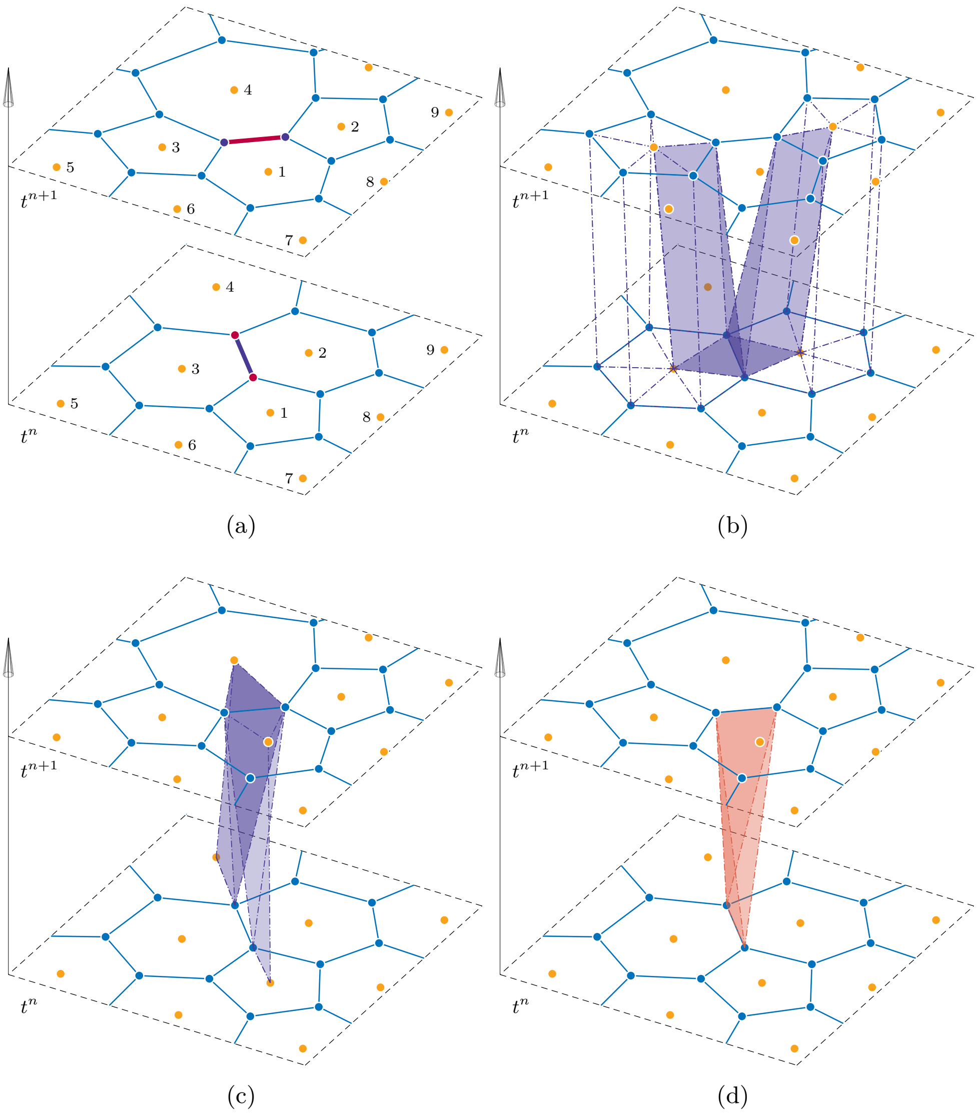
where $\vartheta_k(\mathbf{x},\ t)$ is a modal space–time basis of the polynomials of degree M in $d+1$ dimensions (d space dimensions plus time), which, for the generic spacetime control volume C_i^n, read

$$\vartheta_k(x,\ y,\ t) = \frac{(x-x_i^n)^{p_k}}{p_k!\, h_i^{p_k}}\, \frac{(y-y_i^n)^{q_k}}{q_k!\, h_i^{q_k}}\, \frac{(t-t^n)^{r_k}}{r_k!\, h_i^{r_k}}, \tag{3.61}$$

with

$$k = 0,\ \ldots,\ \mathcal{L}(M,\ d+1), \quad 0 \le p_k + q_k + r_k \le M. \tag{3.62}$$

These space-time basis functions are redefined at the beginning of each time step in function of the current position $\mathbf{x}_i^n$, thus they are directly linked to the current mesh configuration; however, contrarily to the test functions of Equation (3.70) that are used in the corrector step (see the next section), there is no need to move them during each time step, since they allow to represent information at the predictor step, which is only valid locally inside each C_i^n.



(a)

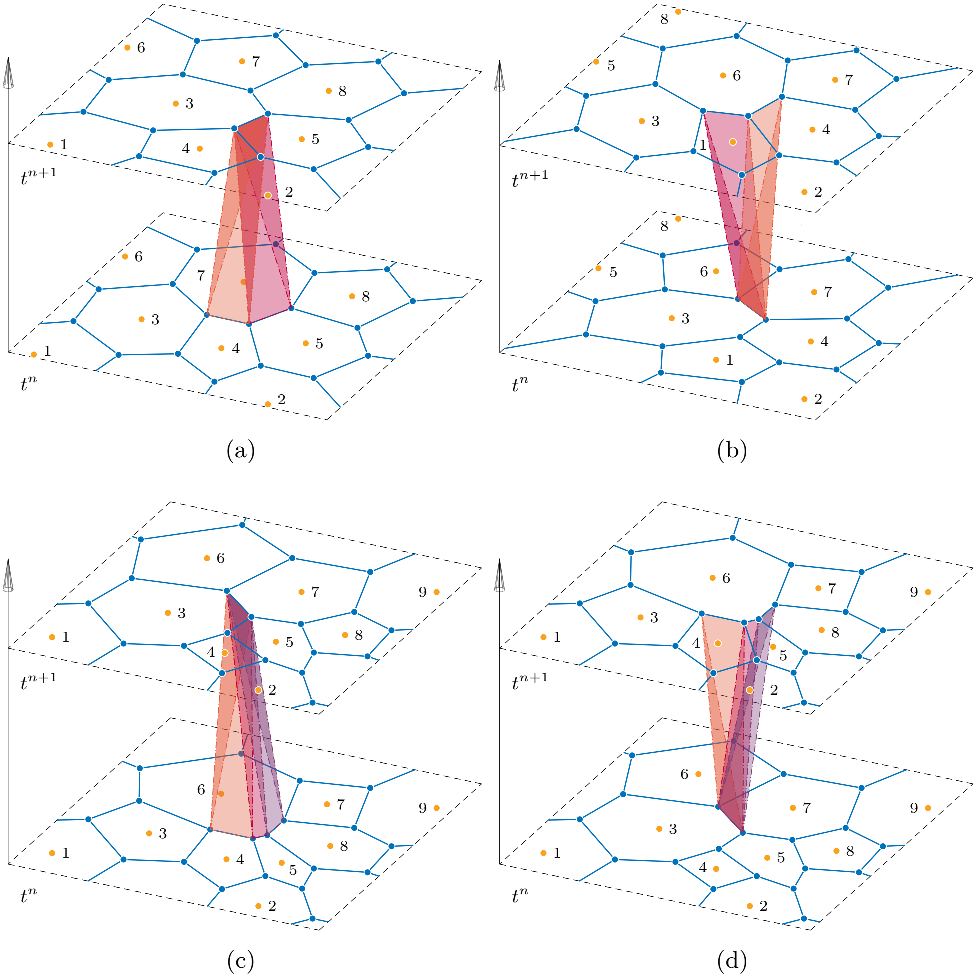
(b)

(c)

(d)

Figure 3.4. Space–time connectivity with topology changes, degenerate sub–space–time control volumes and sliver element. Panel (a): at time t^n the polygons Ω_2^n and Ω_3^n are neighbours and share the highlighted edge, instead at time t^{n+1} they do not touch each other; the opposite situation occurs for polygons Ω_1^n and Ω_4^n. This change of topology causes the appearance of degenerate elements of different types. The first type is given by degenerate sub–space–time control volumes colored in purple in panels (b) and (c). The second type of degenerate elements are called space–time sliver elements, an example is highlighted in panel (d).

The predictor $\mathbf{q}_h^n$ is computed through an iterative procedure that looks for the polynomial satisfying a weak form of (3.1) obtained for any control volume C_i^n as follows. We multiply the governing PDE (3.1) by a test function



(a)

(b)

(c)

(d)

Figure 3.5. Consecutive space–time sliver elements. Refer for example to panel (d): Ω_3^n and Ω_7^n are neighbours at time t^n but this is no longer the case at time t^{n+1} and moreover Ω_4^{n+1}, Ω_5^{n+1}, Ω_6^{n+1} and Ω_8^{n+1} are among them; this complex change of topology triggers the appearance of three space–time sliver elements. A similar situation with three space–time sliver elements is depicted in panel (c). In panels (a) and (b) we show a change of topology with two space–time sliver elements.

ϑ_k, integrate over C_i^n and insert the discrete solution $\mathbf{q}_h^n$ instead of $\mathbf{Q}$, hence obtaining

$$\begin{aligned}\int_{C_i^n} \vartheta_k(\mathbf{x},\ t)\,\frac{\partial \mathbf{q}_h^n}{\partial t}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t + \int_{C_i^n} \vartheta_k(\mathbf{x},\ t)\,\nabla\cdot\mathbf{F}(\mathbf{q}_h^n)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t =\\ = \int_{C_i^n} \vartheta_k(\mathbf{x},\ t)\ [\mathbf{S}(\mathbf{q}_h^n) - \mathbf{B}\,\nabla\mathbf{Q}]\ \mathrm{d}\mathbf{x}\ \mathrm{d}t.\end{aligned} \tag{3.63}$$

Differently from what has been proposed in [40, 41, 105, 112], here we do not integrate the first term in (3.63) by parts in time. Instead, we take into account potential jumps of $\mathbf{q}_h$ on the boundaries of C_i^n in the sense of distributions, combined with upwinding of the fluxes in time. This approach is similar to the path-conservative schemes proposed in [68, 69, 249], but much simpler, since the test functions are only taken from within C_i^n and there is no need to define a non-conservative product on ∂C_i^n. Therefore, the integral containing the time derivative in (3.63) is rewritten as

$$\begin{aligned}\int_{C_i^n} \vartheta_k(\mathbf{x},\ t)\,\frac{\partial \mathbf{q}_h^n}{\partial t}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \int_{C_i^\circ} \vartheta_k(\mathbf{x},\ t)\,\frac{\partial \mathbf{q}_h^n}{\partial t}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t\ +\\ + \int_{\partial C_i^n} \vartheta_k(\mathbf{x},\ t)\ \left(\mathbf{q}_h^{n,+} - \mathbf{q}_h^{n,-}\right)\ \tilde{n}_t^-\ \mathrm{d}S\ \mathrm{d}t,\end{aligned} \tag{3.64}$$

where $C_i^\circ = C_i^n \setminus \partial C_i^n$ denotes the interior of C_i^n. Here, $\mathbf{q}_h^{n,-}$ and $\mathbf{q}_h^{n,+}$ denote the boundary-extrapolated inner and outer states across the interface on ∂C_i^n. Furthermore, we define $\tilde{\mathbf{n}}^- = (\tilde{n}_x^-,\ \tilde{n}_y^-,\ \tilde{n}_t^-)$ to be the space-time outward normal unit vectors on ∂C_i^n that point back in time and $\tilde{n}_t^-$ is their time component, i.e. $\tilde{n}_t^- = \min\,[0,\ \tilde{\mathbf{n}}\cdot(0,\ 0,\ 1)] \leq 0$. Upwinding in time is therefore automatically guaranteed, since we only consider the contributions coming from the past, according to the causality principle. In other words, only time fluxes that enter the space–time control volume C_i^n contribute to the jump term in (3.64), and they are easily identified by checking the sign of the time component of the space–time normal vector $\tilde{\mathbf{n}}$.

The jump term on the bottom surface Ω_i^n of the space–time control volume C_i^n under consideration, where it then simplifies to

$$\left(\mathbf{q}_h^{n,+} - \mathbf{q}_h^{n,-}\right)\ \tilde{n}_t^- = -\left[\mathbf{w}_h^n(\mathbf{x},\ t^n) - \mathbf{q}_h^n(\mathbf{x},\ t^n)\right] = \mathbf{q}_h^n(\mathbf{x},\ t^n) - \mathbf{w}_h^n(\mathbf{x},\ t^n), \tag{3.65}$$

with $\mathbf{q}_h^{n,+} = \mathbf{w}_h(\mathbf{x},\ t^n)$ being simply given by the reconstruction polynomial at time t^n after noting that $\tilde{\mathbf{n}}^- = (0,\ 0,\ -1)$ on Ω_i^n and thus $\tilde{n}_t^- = -1$. In this case, (3.64) reduces to

$$\begin{aligned}\int_{C_i^n} \vartheta_k(\mathbf{x},\ t)\,\frac{\partial \mathbf{q}_h^n}{\partial t}\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \int_{C_i^n \backslash \Omega_i^n} \vartheta_k(\mathbf{x},\ t)\,\frac{\partial \mathbf{q}_h^n}{\partial t}\ \mathrm{d}\mathbf{x}\,\mathrm{d}t\ + \\ \int_{\Omega_i^n} \vartheta_k(\mathbf{x},\ t^n)\,\Big[\mathbf{q}_h^n(\mathbf{x},\ t^n) - \mathbf{w}_h(\mathbf{x},\ t^n)\Big]\ \mathrm{d}\mathbf{x}\end{aligned} \tag{3.66}$$

for standard space–time elements.

The reason for including only jump terms originating at the bottom face Ω_i of the space time element C_i^n is that in this manner, all space–time predictors of the standard elements are decoupled from each other, since they only require the initial data $\mathbf{w}_h^n$ and no information from the neighbour elements. This will not be the case for sliver elements, for which we do not have any reconstruction polynomial available at t^n. If we considered the jump terms also on lateral surfaces of standard space–time elements, the space–time predictors would no longer be independent of each other, since the mesh is moving and there will be in general always a non–empty subset of ∂C_i^n with $\tilde{n}_t^- < 0$.

Doing otherwise would require the imposition of a specific order of execution of the space–time predictor procedure on each element, hindering the parallel performance of the algorithm and adding a further level of complexity to it.

The solution of (3.63)-(3.64) can be found via a simple and fast converging fixed point iteration (a discrete Picard iteration), as detailed in [105, 181]. For linear homogeneous systems, the discrete Picard iteration converges in a finite number of at most $M + 1$ steps, since the involved iteration matrix is nilpotent, see [196]. In practice, one can simply allow a fixed amount of $M + 1$ iterations without affecting the order of convergence $M + 1$ for the scheme, which is what is encountered in practice.

All integrals are evaluated using multidimensional Gaussian quadrature rules of suitable order of accuracy, see [306] and Figure 3.6 for details. Note that for degenerate space–time control sub-volumes, as those of Figures 3.4b and 3.4c, the above quadrature formulae remain well defined, hence the predictor procedure over them does not pose any problem and does not need any adaptation.

It should be emphasized that we first carry out the space–time predictor for all standard elements, which can be computed independently of each other, and only subsequently process the remaining space–time sliver elements.

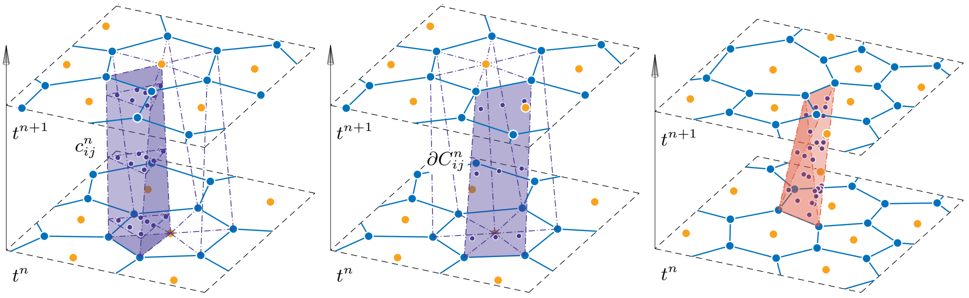
3.2.8 Space–time predictor on degenerate elements

The predictor procedure on space–time sliver elements, as those shown in Figures 3.4d and 3.5, needs particular care. The main problem connected with the space–time sliver elements is the fact that their bottom face is degenerate and consists only in a line segment, hence the spatial integral over Ω_i^n vanishes, i.e. there is no possibility to introduce the initial condition of the local Cauchy problem at time t^n into the predictor for space–time sliver elements.

Furthermore, the degenerate bottom faces are edges of the polygonal tessellation at t^n and are thus at the interface between two adjacent elements, which have in principle a discontinuous solution $\mathbf{w}_h^n$. Therefore, an initial value for a sliver element is in general not easy to define. Thus, in order to couple (3.63) with some known data from the past we have to slightly modify the algorithm detailed previously.

In particular, the upwinding in time approach is not only used for the surface Ω_i^n, as done in (3.65), but on the entire part of the space–time surface ∂C_i^n that encloses a sliver control volume. The information needed to feed the predictor is allowed to come only from the past, i.e. only from those space–time neighbours C_j^n whose common surface $\partial C_{ij}^n = C_i^n \cap C_j^n$ exhibits a negative time component of the outward pointing space–time normal vector ($\tilde{n}_t^- < 0$). In this way, we can introduce information from the past into the space–time sliver elements by considering also its neighbour elements, but respecting at the same time the causality principle in time, hence using again upwinding for the flux evaluation of the jump term in (3.64). As a consequence, the predictor solution $\mathbf{q}_h^n$ is again obtained by means of (3.63), but treating the entire space–time surface ∂C_i^n with the upwind in time approach.

This is slightly different from what is done for standard elements in (3.63)-



(a) (b) (c)

Figure 3.6. Space–time quadrature points for third order methods, i.e. $M = 2$. (a) Quadrature points for the volume integrals and the space–time predictor. (b) Quadrature points for the surface integrals, i.e. for flux computation. (c) Quadrature points for the volume integrals and the space–time predictor for a sliver element.

(3.64), where only the space–time surface at time t^n, i.e. Ω_i^n, is considered for introducing the initial condition $\mathbf{w}_h^n$. In sliver elements, the information from the past requires the knowledge of the predictor solution $\hat{\mathbf{q}}_j^n$ in the neighbour C_j^n. This is the reason why the predictor step must first be performed over all the standard elements, so that the predictor solution $\mathbf{q}_h^n$ is always available to feed the temporal fluxes with the quantities $\hat{\mathbf{q}}_j^n$ that are needed for solving (3.63)-(3.64) in the case of the space–time sliver elements. We underline again that a space–time sliver element has always four standard polygonal elements as neighbours. This closes the description of the predictor step for the space–time sliver elements.

3.2.9 Fully discrete update formulas for the direct ALE ADER-FV-DG scheme

This section contains the core of the direct ALE ADER-FV-DG scheme used to solve the governing equations on regenerating moving meshes.

Following [40–42], the update formula for the direct ALE formulation can be conveniently recovered starting from the space–time divergence form of the PDE

$$\tilde{\nabla} \cdot \tilde{\mathbf{F}}(\mathbf{Q}) + \tilde{\mathbf{B}}(\mathbf{Q})\, \tilde{\nabla}\mathbf{Q} = \mathbf{S}(\mathbf{Q}), \tag{3.67}$$

where we formally defined

$$\tilde{\mathbf{F}} = (\mathbf{F} \cdot \hat{\mathbf{e}}_x,\ \mathbf{F} \cdot \hat{\mathbf{e}}_y,\ \mathbf{Q}), \quad \tilde{\mathbf{B}} = (\mathbf{B} \cdot \hat{\mathbf{e}}_x,\ \mathbf{B} \cdot \hat{\mathbf{e}}_y,\ \mathbf{0}),\ \tilde{\nabla} = (\partial_{\mathbf{x}},\ \partial_t)^{\mathsf{T}}, \tag{3.68}$$

Then, multiplying (3.67) by a set of space–time test functions $\tilde{\varphi}_k$ and integrating over each space–time control volume C_i^n we obtain the weak formulation

$$\int_{C_i^n} \tilde{\varphi}_k(\mathbf{x},\ t)\ \left[\nabla \cdot \tilde{\mathbf{F}}(\mathbf{Q}) + \tilde{\mathbf{B}}(\mathbf{Q})\, \nabla\mathbf{Q}\right]\ \mathrm{d}\mathbf{x}\ \mathrm{d}t = \int_{C_i^n} \tilde{\varphi}_k(\mathbf{x},\ t)\, \mathbf{S}(\mathbf{Q})\ \mathrm{d}\mathbf{x}\ \mathrm{d}t. \tag{3.69}$$

The moving spatial modal test functions $\tilde{\varphi}_k(\mathbf{x},\ t)$, coincide with (3.43) at $t = t^n$ and at $t = t^{n+1}$, i.e. $\tilde{\varphi}_k(\mathbf{x},\ t^n) = \varphi_k(\mathbf{x},\ t^n)$ and $\tilde{\varphi}_k(\mathbf{x},\ t^{n+1}) = \varphi_k(\mathbf{x},\ t^{n+1})$. The test functions are tied to the motion of the generator point $\mathbf{x}_i(t)$ and move together with $\Omega_i(t)$ in such a way that at time $t = t^{n+1}$ they refer to the new barycenter $\mathbf{x}_i^{n+1}$. Thus, the test functions read

$$\tilde{\varphi}_k(x,\ y,\ t) = \frac{[x - x_i(t)]^{p_k}}{p_k!\, h_i^{p_k}}\, \frac{[y - y_i(t)]^{q_k}}{q_k!\, h_i^{q_k}}, \tag{3.70}$$

for indexing defined as $k = 0,\ \dots,\ \mathcal{N}, \quad 0 \leq p_k + q_k \leq N$, and with

$$\mathbf{x}_i(t) = \frac{t - t^n}{\Delta t}\mathbf{x}_i^n + \left(1 - \frac{t - t^n}{\Delta t}\right)\mathbf{x}_i^{n+1}. \tag{3.71}$$

These moving modal basis functions are essential for the approach presented in this paper. They naturally allow for topology changes, without the need of any remapping steps, which is one of the main strengths of the direct ALE formalism.

Now, by applying the Gauss theorem to the flux-divergence term in (3.69) and by splitting the non-conservative products into their volume and surface contribution, the weak formulation becomes

$$\begin{aligned}\int_{\Omega_i^{n+1}} \tilde{\varphi}_k\, \mathbf{u}_h(\mathbf{x},\ t^{n+1})\ \mathrm{d}\mathbf{x} &= \int_{\Omega_i^{n}} \tilde{\varphi}_k\, \mathbf{u}_h(\mathbf{x},\ t^{n})\ \mathrm{d}\mathbf{x}+ \\ &+ \int_{C_i^\circ} \tilde{\varphi}_k(\mathbf{x},\ t)\ [\mathbf{S}(\mathbf{q}_h^n) - \mathbf{B}(\mathbf{q}_h^n)\,\nabla\mathbf{q}_h^n]\ \mathrm{d}\mathbf{x}\ \mathrm{d}t+ \\ &+ \int_{C_i^\circ} \tilde{\nabla}\tilde{\varphi}_k \cdot \tilde{\mathbf{F}}(\mathbf{q}_h)\ \mathrm{d}\mathbf{x}\ \mathrm{d}t - \sum_{j=1}^{N_{st}} \int_{\partial C_{ij}^n} \tilde{\varphi}_k\, \mathbf{D}(\mathbf{q}_h^{n,-},\ \mathbf{q}_h^{n,+}) \cdot \tilde{\mathbf{n}}\ \mathrm{d}S\end{aligned} \tag{3.72}$$

where $\mathbf{Q}$ on Ω_i^{n+1} is represented by the unknown $\mathbf{u}_h^{n+1}$, on P_i^n is taken to be the current representation of the conserved variables $\mathbf{u}_h^n$, in the interior of C_i^n is given by the predictor $\mathbf{q}_h^n$ and on the space–time lateral surfaces ∂C_{ij}^n is given by $\mathbf{q}_h^{n,-}$ and $\mathbf{q}_h^{n,+}$ which are the boundary-extrapolated space-time data, that is the values assumed respectively by the predictors of the two neighbour elements C_i^n and C_j^n on the shared space–time lateral surface ∂C_{ij}^n. Furthermore, we have employed a two-point path-conservative numerical flux function of Rusanov-type

$$\begin{aligned}\mathbf{D}(\mathbf{q}_h^{n,-},\ \mathbf{q}_h^{n,+}) \cdot \tilde{\mathbf{n}} = &\ \frac{1}{2}\left[\tilde{\mathbf{F}}(\mathbf{q}_h^{n,+}) + \tilde{\mathbf{F}}(\mathbf{q}_h^{n,-})\right] \cdot \tilde{\mathbf{n}} - \frac{1}{2}\, s_{\max}\left(\mathbf{q}_h^{n,+} - \mathbf{q}_h^{n,-}\right)\ + \\ &+ \frac{1}{2}\int_0^1 \tilde{\mathbf{B}}\left[\boldsymbol{\Psi}(\mathbf{q}_h^{n,-},\ \mathbf{q}_h^{n,+},\ s)\right] \cdot \mathbf{n}\ \mathrm{d}s\ \left(\mathbf{q}_h^{n,+} - \mathbf{q}_h^{n,-}\right),\end{aligned} \tag{3.73}$$

where $s_{\max}$ is the maximum absolute eigenvalue of the ALE Jacobian matrices $\mathbf{A}_\mathbf{n}^\mathbf{v}(\mathbf{q}_h^{n,+})$ and $\mathbf{A}_\mathbf{n}^\mathbf{v}(\mathbf{q}_h^{n,-})$ computed as

$$\mathbf{A}_\mathbf{n}^\mathbf{v}(\mathbf{Q}) = \sqrt{\tilde{n}_x^2 + \tilde{n}_y^2}\ \left[\left(\frac{\partial \mathbf{F}}{\partial \mathbf{Q}} + \mathbf{B}\right) \cdot \mathbf{n} - (\mathbf{v} \cdot \mathbf{n})\,\mathbf{I}\right], \quad \mathbf{n} = \frac{(\tilde{n}_x,\ \tilde{n}_y)^\intercal}{\sqrt{\tilde{n}_x^2 + \tilde{n}_y^2}}, \tag{3.74}$$

and the path $\boldsymbol{\Psi}(\mathbf{q}_h^-,\ \mathbf{q}_h^+,\ s)$ is a straight-line segment path

$$\boldsymbol{\Psi}(\mathbf{q}_h^-,\ \mathbf{q}_h^+,\ s) = \mathbf{q}_h^- + s\left(\mathbf{q}_h^+ - \mathbf{q}_h^-\right), \quad s \in [0,\ 1]\,, \tag{3.75}$$

connecting $\mathbf{q}_h^{n,-}$ and $\mathbf{q}_h^{n,+}$ which allow to treat the jump of the non-conservative products following the theory introduced in [68, 232, 249], and extended to ADER FV-DG schemes of arbitrary high order in [108, 123].

Finally, using the definitions (3.41) and (3.45), the arbitrary high order one-step direct ALE FV-DG scheme becomes

$$\begin{aligned}
&\left(\int_{\Omega_i^{n+1}} \tilde{\varphi}_k\, \varphi_k \;\mathrm{d}\mathbf{x}\right) \hat{\mathbf{u}}_k^{n+1} = \left(\int_{P_i^n} \tilde{\varphi}_k\, \psi_k \;\mathrm{d}\mathbf{x}\right) \hat{\mathbf{w}}_k^{n} + \\
&+ \int_{C_i^\circ} \tilde{\varphi}_k(\mathbf{x},\ t)\ [\mathbf{S}(\mathbf{q}_h^n) - \mathbf{B}(\mathbf{q}_h^n)\, \nabla \mathbf{q}_h^n] \;\mathrm{d}\mathbf{x}\;\mathrm{d}t + \\
&+ \int_{C_i^n} \tilde{\nabla}\tilde{\varphi}_k \cdot \tilde{\mathbf{F}}(\mathbf{q}_h^n)\;\mathrm{d}\mathbf{x}\,\mathrm{d}t - \sum_{j=1}^{N_{st}} \int_{\partial C_{ij}^n} \tilde{\varphi}_k\, \mathbf{D}(\mathbf{q}_h^{n,-},\ \mathbf{q}_h^{n,+}) \cdot \tilde{\mathbf{n}}\;\mathrm{d}S
\end{aligned} \tag{3.76}$$

The volume integrals in the above expression (3.76) can be easily computed directly on the physical space-time element C_i^n by employing Gaussian quadrature rules of sufficient precision on a simple decomposition of the base polygon into triangles or quadrilaterals, see Figure 3.6a and [306]. The lateral space–time surfaces of ∂C_{ij}^n instead are parameterized using a set of bilinear basis functions [40], that is

$$\partial C_{ij}^n = \tilde{\mathbf{x}}\,(\chi,\ \tau) = \sum_{k=1}^{4} \beta_k(\chi,\ \tau)\, \tilde{\mathbf{x}}_{ij,k}^n, \quad 0 \le \chi \le 1, \quad 0 \le \tau \le 1, \tag{3.77}$$

where the symbol $\tilde{\mathbf{x}}_{ij,k}^n$ represents the space–time coordinates of the four vertexes of ∂C_{ij}^n, and integrals over each space–time face ∂C_{ij}^n are computed by means of the bilinear mapping to a reference square $[0,\ 1] \times [0,\ 1]$. Formally, the mapping is defined by

$$\beta_1 = (1-\chi)\,(1-\tau), \quad \beta_2 = \chi\,(1-\tau), \quad \beta_3 = \chi\,\tau, \quad \beta_4 = (1-\chi)\,\tau. \tag{3.78}$$

with $\tau = t - t^n/\Delta t$.

We close this section remarking that the integration of the governing PDE over the space-time volume C_i^n automatically satisfies the geometric conservation law (GCL) for all test functions $\tilde{\varphi}_k$. This simply follows from Gauss theorem applied to closed space–time control volumes and we refer to [41] for a complete proof.

4 Relaxation processes in the unified model of continuum mechanics

4.1 Problem statement

A necessary element for the successful solution of the unified model of continuum mechanics is the accurate integration of the distortion matrix $\mathbf{A}$.

The evolution dynamics of the distortion matrix $\mathbf{A}$ and of the metric tensor $\mathbf{G} = \mathbf{A}^\mathsf{T}\,\mathbf{A}$ take place on a wide span of timescales: given a fixed evolution speed of the kinematics of distortion (due to flow convection and velocity gradients), one can find anything from infinitely slow strain relaxation in elastic solids, to infinitely fast shear dissipation in perfect fluids, with viscous fluids also being a nontrivial example of fast-acting (stiff) strain relaxation.

From the mathematical standpoint, such timescales can be quantified by means of a relaxation time τ in the evolution equation of the distortion matrix

$$\partial_t \mathbf{A} + (\nabla \mathbf{A})\ \mathbf{u} + (\nabla \mathbf{u})\ \mathbf{A} = \mathbf{Z} = -\frac{3}{\tau}\ (\det \mathbf{A})^{5/3}\ \mathbf{A}\ \operatorname{dev}(\mathbf{A}^\mathsf{T}\,\mathbf{A}) \tag{4.1}$$

and in the corresponding equation for the metric tensor

$$\partial_t \mathbf{G} + (\nabla \mathbf{G})\ \mathbf{u} + \mathbf{G}\,\nabla \mathbf{u} + (\nabla \mathbf{u})^\mathsf{T}\ \mathbf{G} = -\frac{6}{\tau}\,(\det \mathbf{G})^{5/6}\,\mathbf{G}\ \operatorname{dev}\mathbf{G}. \tag{4.2}$$

The relaxation time τ, in principle a function of the state variables, but often a fixed constant, is what defines the stiff nature of the algebraic source terms governing the relaxation towards an equilibrium state of the material strain. To highlight the connection between the distortion matrix $\mathbf{A}$, the metric tensor $\mathbf{G}$ and what we generically call *strain*, it should be recalled that, in a purely elastic context, for small deformations, the linear strain $\boldsymbol{\varepsilon}$ can be directly expressed as $\boldsymbol{\varepsilon} = (\mathbf{I} - \mathbf{G})\,/2$, for this reason we refer to the above right hand side terms as *strain relaxation sources.*

One of the major difficulties in the solution of the unified model of continuum mechanics is indeed the presence of these nonlinear source terms. In the past, the locally implicit ADER treatment of source terms has proven effective, as well as the splitting or fractional step approach, in conjunction with the implicit Euler scheme for stable time integration. However, we found that a new approach has to be adopted for certain choices of the material parameters, for example for extremely fast relaxation times in complex flows, or for the nonlinearly stress-dependent timescales encountered in the application of the model to material failure dynamics.

4.2 Preliminary developments with the Baer–Nunziato model of two-phase flow

In the context of this thesis, the first important development towards an efficient and robust solution method for the strain relaxation sources in the unified model of continuum mechanics has been achieved by first tackling a similar but simpler problem: that of finite-rate pressure relaxation in Baer–Nunziato two-phase flow.

The technique is based on the fractional step approach coupled with an exponential type integrator which obtains the solutions to a relaxation ordinary differential equation system as a sequence of solutions of linearised problems, for which analytical expressions can be retrieved.

The solutions to a linear system of ordinary differential equations

$$\frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} + \mathbf{J}\,\mathbf{V} = \mathbf{B}, \tag{4.3}$$

are expressed in terms of exponentials of the matrix of coefficients of the system itself. Matrix exponentials can be costly to compute as one must resort to either carrying out a spectral decomposition of the matrix or, to the application of the ad-hoc approximants of Al-Mohy and Higham [5, 6, 182], which in any case involve the solution of a linear system of equations.

Moreover, for non-homogeneous systems of linear ODEs, an additional matrix inversion operation is needed, and in general the matrix of coefficients could be difficult to invert.

However, as shown in this Section, carefully constructed approximations of the Jacobian matrix $\mathbf{J}$ admit closed-form analytical solutions which can be evaluated without the use of matrix exponentials. This allows the scheme to handle arbitrarily stiff finite-rate pressure and velocity relaxation processes in Baer–Nunziato models in a robust and efficient manner.

Robustness is given by the adaptive timestepping scheme that explicitly checks for physical admissibility of the computed solution, reducing or increasing the integration step size according to the needs.

Efficiency is due to the fact that the scheme does not require the numerical solution of any systems of coupled algebraic equations, even element-local ones.

Accuracy is achieved by construction, since instead of representing solutions in time of the governing nonlinear ODE as piecewise linear or piecewise polynomial, they can be rather rich functions that conform as closely as possible to the structure of the exact solution of the problem.

4.2.1 A semi-analytic integrator for the relaxation sources of the Baer–Nunziato system

Stiff algebraic source terms, accounting for mechanical relaxation and phase transition in two-phase flow models of the Baer–Nunziato type [10, 253, 284], are one of the key difficulties in computing solutions to these systems of hyperbolic partial differential equations (PDE). Their accurate solution is relevant for the study of droplet dynamics with Baer–Nunziato models. These weakly compressible phenomena can be accurately described by the reduced models that assume instantaneous pressure and velocity equilibrium like the one forwarded by Kapila et al. [201]. Solving more general sets of equations like [10, 253, 284] in the stiff relaxation limit gives results that are similar to those obtained from the instantaneous equilibrium model, while allowing more modelling flexibility, since less physical assumptions have to be made.

A simple computational strategy for dealing with stiff sources is the *splitting* approach [305, 332]. The procedure consists of two steps: at each timestep, first one solves the homogeneous part of the PDE

$$\partial_t \mathbf{Q} + \nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q})\, \nabla \mathbf{Q} = \mathbf{S}(\mathbf{Q}), \tag{4.4}$$

for example with a path-conservative [68, 249] MUSCL–Hancock [212] method, obtaining a preliminary solution $\mathbf{Q}_{\mathrm{H}}$ and then one can use this state vector as initial condition for the Cauchy problem

$$\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} = \mathbf{S}(\mathbf{Q}), \quad \mathbf{Q}(t^n) = \mathbf{Q}_{\mathrm{H}}, \quad t \in (t^n,\ t^{n+1}), \tag{4.5}$$

of which the solution will then yield the updated quantities at the new time level t^{n+1}. This way, the problem is reduced to the integration of a system of ordinary differential equations (ODE), and general-purpose ODE solvers or more specialised tools can be employed for this task.

It is often the case that the time scales associated with relaxations sources are much shorter than those given by the stability condition of the PDE scheme, thus one must be able to deal with source terms that are potentially stiff. In order to integrate stiff ODEs with conventional explicit solvers, one has to impose very severe restrictions on the maximum timestep size, and for this reason implicit methods are commonly preferred [317]. Unfortunately implicit solvers, are, on a per-timestep basis, much more expensive than explicit integrators, and they still might require variable sub-timestepping in order to avoid under-resolving complex transients in the solution.

In this work, we will develop a technique for constructing a solver for stiff finite-rate mechanical relaxation sources, specifically those encountered in models of the Baer–Nunziato type.

The proposed method overcomes the issues typical of explicit solvers with three concurrent strategies: first, the update formula is based on exponential integration [80, 190, 264], in order to mimic at the discrete level the behaviour of the differential equation; second, information at the new time level t^{n+1} is taken into account by iteratively updating a linearisation of the ODE system, this is achieved without resorting to a fully implicit method like those introduced in [63], and for which one would need to solve a system of nonlinear algebraic equations at each timestep t^n; third, the method incorporates a simple and effective adaptive timestepping criterion, which is crucial for capturing abrupt changes in the state variables and dealing with the different time scales that characterise the equations under investigation.

4.2.2 Model equations

We are interested in the solution of two-phase flow models of the Baer–Nunziato family, which can be written in the general form (4.4), with a vector of conserved variables defined as

$$\mathbf{Q} = [\alpha_1\,\rho_1,\ \alpha_2\,\rho_2,\ \alpha_1\,\rho_1\,\mathbf{u}_1,\ \alpha_2\,\rho_2\,\mathbf{u}_2,\ \alpha_1\,\rho_1\,E_1,\ \alpha_2\,\rho_2\,E_2,\ \alpha_1]^{\mathsf{T}}, \tag{4.6}$$

a conservative flux $\mathbf{F}$ and a non-conservative term $\mathbf{B}\,\nabla\mathbf{Q}$ written as

$$\mathbf{F}(\mathbf{Q}) = \begin{pmatrix} \alpha_1\,\rho_1\,\mathbf{u}_1 \\ \alpha_2\,\rho_2\,\mathbf{u}_2 \\ \alpha_1\,(\rho_1\,\mathbf{u}_1\otimes\mathbf{u}_1 + p_1\,\mathbf{I}) \\ \alpha_2\,(\rho_2\,\mathbf{u}_2\otimes\mathbf{u}_2 + p_2\,\mathbf{I}) \\ \alpha_1\ (\rho_1\,E_1 + p_1)\ \mathbf{u}_1 \\ \alpha_2\ (\rho_2\,E_2 + p_2)\ \mathbf{u}_2 \\ 0 \end{pmatrix}, \quad \mathbf{B}(\mathbf{Q})\,\nabla\mathbf{Q} = \begin{pmatrix} 0 \\ 0 \\ -p_{\mathrm{I}}\,\nabla\alpha_1 \\ +p_{\mathrm{I}}\,\nabla\alpha_1 \\ -p_{\mathrm{I}}\,\mathbf{u}_{\mathrm{I}}\cdot\nabla\alpha_1 \\ +p_{\mathrm{I}}\,\mathbf{u}_{\mathrm{I}}\cdot\nabla\alpha_1 \\ \mathbf{u}_{\mathrm{I}}\cdot\nabla\alpha_1 \end{pmatrix}, \tag{4.7}$$

and a source term vector written as

$$\mathbf{S}(\mathbf{Q}) = \begin{pmatrix} 0 \\ 0 \\ \lambda\,(\mathbf{u}_2 - \mathbf{u}_1) \\ \lambda\,(\mathbf{u}_1 - \mathbf{u}_2) \\ \lambda\,(\mathbf{u}_2 - \mathbf{u}_1)\cdot\mathbf{u}_\mathrm{I} + \nu\,p_\mathrm{I}\,(p_2 - p_1) \\ \lambda\,(\mathbf{u}_1 - \mathbf{u}_2)\cdot\mathbf{u}_\mathrm{I} + \nu\,p_\mathrm{I}\,(p_1 - p_2) \\ \nu\,(p_1 - p_2) \end{pmatrix}. \tag{4.8}$$

Here we indicate with α_1 and α_2 the volume fractions of the first phase and of the second phase respectively, with ρ_1 and ρ_2 the phase densities, $\mathbf{u}_1 = (u_1,\ v_1,\ w_1)^\mathsf{T}$ and $\mathbf{u}_2 = (u_2,\ v_2,\ w_2)^\mathsf{T}$ indicate the velocity vectors, $\alpha_1\,\rho_1\,E_1$ and $\alpha_2\,\rho_2\,E_2$ are the partial energy densities. The pressure fields are denoted with p_1 and p_2, and the interface pressure and velocity are named p_I and $\mathbf{u}_\mathrm{I} = (u_\mathrm{I},\ v_\mathrm{I},\ w_\mathrm{I})^\mathsf{T}$. Finally, the parameters λ and ν control the time scales for friction and pressure relaxation kinetics respectively.

In the following, we will study the system of ordinary differential equations arising from the source term (4.8) only, that is, the one constructed as given in equation (4.5) and specifically its one-dimensional simplification in terms of the primitive variables $\mathbf{V} = (u_1,\ u_2,\ p_1,\ p_2,\ \alpha_1)^\mathsf{T}$, with an initial condition $\mathbf{V}_0 = (u_1^0,\ u_2^0,\ p_1^0,\ p_2^0,\ \alpha_1^0)^\mathsf{T}$. Since no source is present in the mass conservation equations, they have a trivial solution, that is, $\alpha_1\,\rho_1$ and $\alpha_2\,\rho_2$ remain constant in time; for compactness, these quantities will be included in our analysis as constant parameters, rather than as variables of the ODE system.

The one-dimensional ODE system is written as

$$\frac{\mathrm{d}u_1}{\mathrm{d}t} = \frac{\lambda}{\alpha_1\,\rho_1}\,(u_2 - u_1), \tag{4.9}$$

$$\frac{\mathrm{d}u_2}{\mathrm{d}t} = \frac{\lambda}{\alpha_2\,\rho_2}\,(u_1 - u_2), \tag{4.10}$$

$$\frac{\mathrm{d}p_1}{\mathrm{d}t} = \frac{\nu\,(p_I + k_{1a}\,p_1 + k_{1b})}{\alpha_1\,k_{1a}}\,(p_2 - p_1) + \frac{\lambda\,(u_I - u_1)}{\alpha_1\,k_{1a}}\,(u_2 - u_1), \tag{4.11}$$

$$\frac{\mathrm{d}p_2}{\mathrm{d}t} = \frac{\nu\,(p_I + k_{2a}\,p_2 + k_{2b})}{\alpha_2\,k_{2a}}\,(p_1 - p_2) + \frac{\lambda\,(u_I - u_2)}{\alpha_2\,k_{2a}}\,(u_1 - u_2), \tag{4.12}$$

$$\frac{\mathrm{d}\alpha_1}{\mathrm{d}t} = \nu\,(p_1 - p_2). \tag{4.13}$$

The choices for interface pressure and velocity are $p_I = p_2$ and $u_I = u_1$. Finally, one can verify that, using the stiffened-gas equation of state for both phases, we have $k_{1a} = 1/(\gamma_1 - 1)$, $k_{2a} = 1/(\gamma_2 - 1)$, $k_{1b} = \gamma_1\,\Pi_1/(\gamma_1 - 1)$, and $k_{2b} = \gamma_2\,\Pi_2/(\gamma_2 - 1)$.

4.3 Description of the numerical method

The methodology is described in the following with reference to a generic nonlinear first order Cauchy problem

$$\frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} = \mathbf{S}(\mathbf{V},\ t), \qquad \mathbf{V}(t^n) = \mathbf{V}_n, \tag{4.14}$$

for which the ODE can be linearised about a given state $\mathbf{Q}^*$ and time t^* as

$$\frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} = \mathbf{B}^* + \mathbf{J}^*(\mathbf{Q}^*,\ t^*)\,(\mathbf{V} - \mathbf{Q}^*). \tag{4.15}$$

Here we defined the Jacobian matrix of the source $\mathbf{J}^* = \mathbf{J}(\mathbf{Q}^*,\ t^*)$ and analogously the source vector evaluated at the linearisation state is $\mathbf{B}^* = \mathbf{S}(\mathbf{Q}^*,\ t^*)$. We then introduce the vector

$$\mathbf{C}^* = \mathbf{C}^*(\mathbf{B}^*,\ \mathbf{J}^*) = \mathbf{C}^*(\mathbf{Q}^*,\ t^*), \tag{4.16}$$

which will be used as an indicator for the adaptive timestepping algorithm and may be constructed for example listing all of the components of the matrix $\mathbf{J}^*$ together with all the components of the vector $\mathbf{B}^*$ and the state $\mathbf{Q}^*$, or only with a selection of these variables, or any other relevant combination of the listed variables, that is, any group indicative of changes in the nature or the magnitude of the linearised source operator.

It is then necessary to compute an accurate analytical solution of the nonhomogeneous linear Cauchy problem

$$\frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} = \mathbf{S}^*(\mathbf{V};\ \mathbf{Q}^*,\ t^*) = \mathbf{B}^* + \mathbf{J}^*(\mathbf{Q}^*,\ t^*)\,(\mathbf{V} - \mathbf{Q}^*), \qquad \mathbf{V}(t^n) = \mathbf{V}_n. \tag{4.17}$$

We will denote the analytical solution of the IVP (4.17) as $\mathbf{Q}_\mathrm{e}(t;\ \mathbf{S}^*,\ t^n,\ \mathbf{V}_n)$. As for $\mathbf{S}^*(\mathbf{V};\ \mathbf{Q}^*,\ t^*)$, the semicolon separates the variable on which $\mathbf{Q}_\mathrm{e}$ and $\mathbf{S}^*$ continuously depend (t or $\mathbf{V}$) from the parameters used in the construction of the operators. The state vector at a generic time level t^n is written as $\mathbf{V}_n$, the variable timestep size is $\Delta t^n = t^{n+1} - t^n$.

4.3.1 Timestepping

Marching from a start time t_0 to an end time t^{end} is carried out as follows. First, an initial timestep size Δt^0 is chosen, then, at each time iteration, the state

$\mathbf{V}_{n+1}$ at the new time level t^{n+1} is computed by means of the iterative procedure described below. The iterative procedure will terminate by computing a value for $\mathbf{V}_{n+1}$, together with a new timestep size $\Delta t^{n+1} = t_{n+2} - t^{n+1}$ based on an estimator which is embedded in the iterative solution algorithm. There is also the possibility that, due to the timestep size Δt being too large, the value of $\mathbf{V}_{n+1}$ be flagged as not acceptable. In this case, the procedure will return a new shorter timestep size for the current timestep $\Delta t^n = t^{n+1} - t_n$ and a new attempt at the solution for $\mathbf{V}_{n+1}$ will be carried out. Specifically, in practice we choose the new timestep size to be half of the one used in the previous attempt.

4.3.2 Iterative computation of the timestep solution

At each iteration (denoted by the superscript k) we define an average state vector ${\mathbf{Q}^*}^k_{n+1/2} = (\mathbf{V}_n + {\mathbf{Q}^*}^{k-1}_{n+1})/2$ to be formally associated with an intermediate time level $t^{n+1/2} = \left(t^n + t^{n+1}\right)/2$. For the first iteration we need a guess value for ${\mathbf{Q}^*}^{k-1}_{n+1}$, with the simplest choice being ${\mathbf{Q}^*}^{k-1}_{n+1} = \mathbf{V}_n$. Then the coefficients ${\mathbf{C}^*}^k_{n+1/2}$ are computed as

$$ {\mathbf{C}^*}^k_{n+1/2} = {\mathbf{C}^*}^k_{n+1/2}({\mathbf{Q}^*}^k_{n+1/2},\ t^{n+1/2}). \tag{4.18}$$

In a joint way, one can build the affine source operator

$$ {\mathbf{S}^*}^k_{n+1/2} = {\mathbf{S}^*}^k_{n+1/2}(\mathbf{V};\ {\mathbf{Q}^*}^k_{n+1/2},\ t^{n+1/2}). \tag{4.19}$$

Then one can solve analytically

$$ \frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} = {\mathbf{S}^*}^k_{n+1/2}(\mathbf{V};\ {\mathbf{Q}^*}^k_{n+1/2},\ t^{n+1/2}), \qquad \mathbf{V}(t^n) = \mathbf{V}_n, \tag{4.20}$$

by computing

$$ {\mathbf{Q}^*}^k_{n+1} = \mathbf{V}_\mathrm{e}\left(t^{n+1};\ {\mathbf{S}^*}^k_{n+1/2},\ t^n,\ \mathbf{V}_n\right). \tag{4.21}$$

It is then checked that the state vector ${\mathbf{Q}^*}^k_{n+1}$ be physically admissible: in our case this means verifying that internal energy of each phase be positive and that the volume fraction be bounded between 0 and 1. Also one can check for absence of floating-point exceptions. Additionally, one must evaluate

$$ {\mathbf{C}^*}^k_{n+1} = {\mathbf{C}^*}^k_{n+1}\left({\mathbf{Q}^*}^k_{n+1},\ t^{n+1}\right). \tag{4.22}$$

This vector of coefficients will not be employed for the construction of an affine source operator ${\mathbf{S}^*}^k_{n+1}$, but only for checking the validity of the solution obtained from the approximate problem (4.20) by comparing the coefficients vector ${\mathbf{C}^*}_{n+1}^{k}$ to ${\mathbf{C}^*}_{n}$, as well as comparing the coefficients ${\mathbf{C}^*}_{n+1/2}^{k}$ used in the middle-point affine operator for the initial coefficients ${\mathbf{C}^*}_{n}$. At the end of the iterative procedure, one will set ${\mathbf{C}^*}_{n+1} = {\mathbf{C}^*}_{n+1}^{k}$, so that this will be the new reference vector of coefficients for the next timestep. The convergence criterion for stopping the iterations is implemented by computing

$$r = \max\left(\frac{\left|{\mathbf{Q}^*}_{n+1}^{k} - {\mathbf{Q}^*}_{n+1}^{k-1}\right|}{\left|{\mathbf{Q}^*}_{n+1}^{k}\right| + \left|{\mathbf{Q}^*}_{n+1}^{k-1}\right| + \varepsilon_r}\right), \tag{4.23}$$

and checking if $r \le r_{\max}$, with $r_{\max}$ and ε_r given tolerances, or if the iteration count k has reached a fixed maximum value $k_{\max}$. Note that in principle any norm may be used to compute the error metric given in equation (4.23), as this is just a measure of the degree to which ${\mathbf{Q}^*}_{n+1}^{k}$ was corrected in the current iteration. Moreover, we found convenient to limit the maximum number of iterations allowed, and specifically here we set $k_{\max} = 8$, but stricter bounds can be used. For safety, we decide to flag the state vector ${\mathbf{Q}^*}_{n+1}^{k}$ as not admissible, as if a floating-point exception had been triggered, whenever the iterative procedure terminates by reaching the maximum iteration count.

After the convergence has been obtained, in order to test if the IVP (4.14) is well approximated by its linearised version (4.20), we compute

$$\delta_{n+1/2} = \max\left(\frac{\left|{\mathbf{C}^*}_{n+1/2} - {\mathbf{C}^*}_{n}\right|}{\left|{\mathbf{C}^*}_{n+1/2}\right| + \left|{\mathbf{C}^*}_{n}\right| + \varepsilon_\delta}\right), \tag{4.24}$$

$$\delta_{n+1} = \max\left(\frac{\left|{\mathbf{C}^*}_{n+1} - {\mathbf{C}^*}_{n}\right|}{\left|{\mathbf{C}^*}_{n+1}\right| + \left|{\mathbf{C}^*}_{n}\right| + \varepsilon_\delta}\right), \tag{4.25}$$

and we verify if $\delta = \max(\delta_{n+1/2},\ \delta_{n+1}) \le \delta_{\max}$. The user should specify a tolerance $\delta_{\max}$ as well as the floor value ε_δ, which is used in order to prevent that excessive precision requirements be imposed in those situations when all the coefficients are so small than even large relative variations expressed by equations (4.24) and (4.25) do not affect the solution in a significant manner. If $\delta \le \delta_{\max}$ we confirm the state vector at the new time level to be $\mathbf{V}_{n+1} = {\mathbf{Q}^*}_{n+1}^{k}$ and a new timestep size is computed as

$$\Delta t^{n+1} = \lambda\,\frac{\delta_{\max}}{\delta + \varepsilon}, \quad \text{with} \quad \lambda = 0.8, \quad \varepsilon = 10^{-14}, \tag{4.26}$$

otherwise the solution of the IVP (4.20) is attempted again with a reduced timestep size, specifically one that is obtained by halving the timestep used in the current attempt. The same happens if at any time the admissibility test on ${\mathbf{Q}^*}_{n+1}^{k}$ fails.

4.3.3 Analytical solution of the linearised problem

The general solution to an initial value problem like (4.20) can be written as

$$\mathbf{V}(t) = \exp\left[\mathbf{J}^{*}\,(t - t^{n})\right]\,\Big[\mathbf{V}(t^{n}) + \mathbf{J}^{*-1}\,\mathbf{B}^{*} - \mathbf{V}^{*}\Big] - \mathbf{J}^{*-1}\,\mathbf{B}^{*} + \mathbf{V}^{*}. \tag{4.27}$$

Note that, in addition to evaluating the matrix exponential $\exp\left[\mathbf{J}^{*}\,(t - t^{n})\right]$, one must also compute the inverse Jacobian matrix $\mathbf{J}^{*-1}$. Computation of matrix exponentials can be carried out rather robustly in double precision arithmetic with the aid of the algorithms of Higham [182] and Al-Mohy and Higham [5, 6], while inversion of the Jacobian matrix can be an arbitrarily ill-conditioned problem, to be carefully treated or avoided if possible.

For this reason we propose the following strategy for choosing a more suitable linearisation and computing analytical solutions of the linearised problem for the ODE system (4.9)–(4.13). First, it is easy to see that the velocity sub-system (equations for u_1 and u_2) can be fully decoupled from the other equations, as the partial densities $\alpha_1\,\rho_1$ and $\alpha_2\,\rho_2$ remain constant in the relaxation step. Then the solution of the velocity sub-system can be immediately obtained as

$$u_1(t) = \frac{\lambda}{k}\left\{\frac{u_1^0}{\alpha_2\,\rho_2} + \frac{u_2^0}{\alpha_1\,\rho_1} + \frac{u_1^0 - u_2^0}{\alpha_1\,\rho_1}\,\exp\left[-k\,(t - t^{n})\right]\right\}, \tag{4.28}$$

$$u_2(t) = \frac{\lambda}{k}\left\{\frac{u_1^0}{\alpha_2\,\rho_2} + \frac{u_2^0}{\alpha_1\,\rho_1} + \frac{u_2^0 - u_1^0}{\alpha_2\,\rho_2}\,\exp\left[-k\,(t - t^{n})\right]\right\}, \tag{4.29}$$

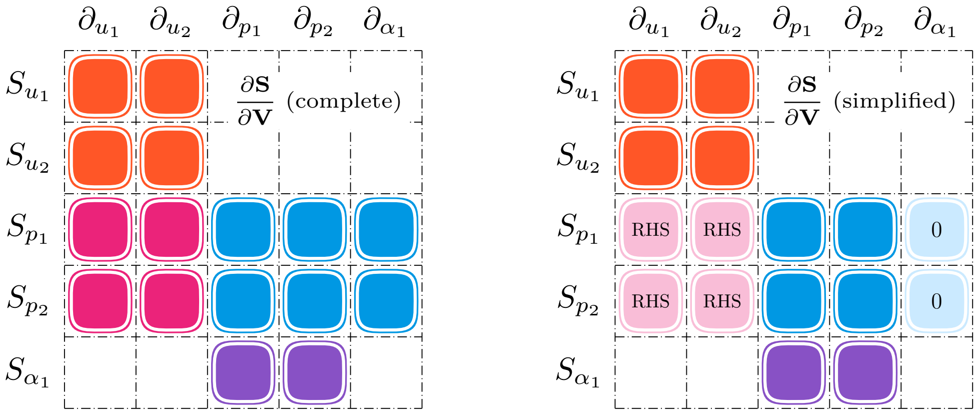

∂_{u_1} ∂_{u_2} ∂_{p_1} ∂_{p_2} ∂_{α_1}
S_{u_1}
S_{u_2}
S_{p_1}
S_{p_2}
S_{α_1}
$\frac{\partial \mathbf{S}}{\partial \mathbf{V}}$ (complete)
∂_{u_1} ∂_{u_2} ∂_{p_1} ∂_{p_2} ∂_{α_1}
S_{u_1}
S_{u_2}
S_{p_1}
S_{p_2}
S_{α_1}
$\frac{\partial \mathbf{S}}{\partial \mathbf{V}}$ (simplified)
RHS
RHS
0
RHS
RHS
0

Figure 4.1. Visual comparison between the structure of the complete Jacobian matrix for the ODE system (4.9)–(4.13) and the proposed three-step simplified structure. The RHS label indicates dependencies that are accounted for as non-homogeneous terms in the pressure sub-system, while the zeros mark dependencies that are suppressed entirely.

with $k = 1/\alpha_1\,\rho_1 + 1/\alpha_2\,\rho_2$. In a second step, the pressure sub-system (4.11)–(4.12) is linearised as

$$\frac{\mathrm{d}p_1}{\mathrm{d}t} = k_p\,(p_2 - p_1) + k_u\,(u_I - u_1)\,(u_2 - u_1), \tag{4.30}$$

$$\frac{\mathrm{d}p_2}{\mathrm{d}t} = k_p\,(p_1 - p_2) + k_u\,(u_I - u_2)\,(u_1 - u_2), \tag{4.31}$$

where k_p and k_u are constant coefficients directly obtained from equations (4.11)–(4.12). This way, at the cost of suppressing the dependence on α_1 in the Jacobian of the pressure sub-system, the homogeneous part of equations (4.30)–(4.31) has the same simple structure found in the velocity sub-system, with the addition of a non-homogeneous term, which is known, as $u_1(t)$ and $u_2(t)$ already have been computed. The solution can again be evaluated using standard scalar exponential functions, which are fast and robust, compared to matrix exponentials and especially so, because one no longer needs to perform the inversion of the Jacobian matrix of the full system. Finally, the solution to equation (4.13) can be integrated analytically from the expressions of $p_1(t)$ and $p_2(t)$. Full coupling of the system is restored in the successive iterations by recomputing the constant coefficients k_p and k_u using an updated midpoint value for α_1. See Figure 4.1 for a graphical description of the proposed simplified solution structure.

4.4 Treatment of strain relaxation source terms with exponential integrators

The idea developed on the Baer–Nunziato model can be ported quite easily to deal with the much more complex strain relaxation dynamics of the unified model of continuum mechanics. Unfortunately one can no longer avoid the fairly expensive process of computing matrix exponential and inverting the Jacobian matrix of the ODE system, which in this case reads

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{pmatrix}\xi \\ \mathbf{A}\end{pmatrix} = \begin{pmatrix}\psi_\xi \\ -3\,\tau_{\mathrm{m}}^{-1}\,(\det\mathbf{A})^{5/3}\,\mathbf{A}\,\mathrm{dev}\,G\end{pmatrix}. \tag{4.32}$$

Note that (4.32) is a system of 10 fully coupled ordinary differential equations, since $\tau_{\mathrm{m}} = \tau_{\mathrm{m}}(\xi)$ depends on the damage variable ξ and clearly the structure of the relaxation source is such that all components of $\mathbf{A}$ depend on all other components of $\mathbf{A}$ during the relaxation process. Similarly, the source term ψ_ξ depends on the stress tensor, which in turn is a function of all components of the distortion matrix $\mathbf{A}$. Nonetheless, several simplifications can be carried out to minimize the computational expense of the method. In the following we denote for brevity $\mathbf{Q} = (\xi,\ \mathbf{A})$ and

$$\mathbf{S} = \left(\psi_\xi,\ -3\,\tau_{\mathrm{m}}^{-1}\,(\det \mathbf{A})^{5/3}\ \mathbf{A}\ \mathrm{dev}\,G\right). \tag{4.33}$$

If the system of equations in linearised form about a given state $\mathbf{Q}^*$ and time t^* as

$$\frac{\mathrm{d}\mathbf{V}}{\mathrm{d}t} = \mathbf{B}^* + \mathbf{J}^*(\mathbf{Q}^*,\ t^*)\,(\mathbf{V} - \mathbf{Q}^*) \tag{4.34}$$

is large (10 ODE in this case is sufficiently large), it might be infeasible to explicitly evaluate a closed form expression for the solution of the linearised initial value problem (4.17), which formally can be written as

$$\mathbf{Q}(t) = \exp\left[\mathbf{J}^*\,(t - t^n)\right]\left[\mathbf{Q}(t^n) + (\mathbf{J}^*)^{-1}\ \mathbf{B}^* - \mathbf{Q}^*\right] - (\mathbf{J}^*)^{-1}\ \mathbf{B}^* + \mathbf{Q}^*, \tag{4.35}$$

due to the fact that the evaluation of the matrix exponential and the computation of the inverse Jacobian $(\mathbf{J}^*)^{-1}$ in general need to be carried out numerically, and in particular the inversion of $\mathbf{J}^*$ may constitute an ill-conditioned problem. Specifically, the timescale associated with strain relaxation can be significantly different from the reaction speed of material failure. This issue can be treated by detecting those rows of the Jacobian that have much smaller entries than

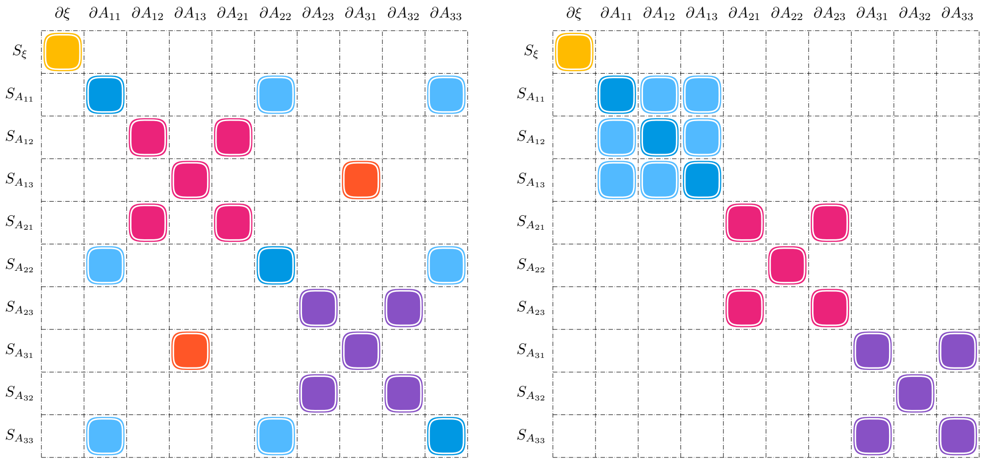

$\partial\xi$ ∂A_{11} ∂A_{12} ∂A_{13} ∂A_{21} ∂A_{22} ∂A_{23} ∂A_{31} ∂A_{32} ∂A_{33}
S_ξ
$S_{A_{11}}$
$S_{A_{12}}$
$S_{A_{13}}$
$S_{A_{21}}$
$S_{A_{22}}$
$S_{A_{23}}$
$S_{A_{31}}$
$S_{A_{32}}$
$S_{A_{33}}$
$\partial\xi$ ∂A_{11} ∂A_{12} ∂A_{13} ∂A_{21} ∂A_{22} ∂A_{23} ∂A_{31} ∂A_{32} ∂A_{33}
S_ξ
$S_{A_{11}}$
$S_{A_{12}}$
$S_{A_{13}}$
$S_{A_{21}}$
$S_{A_{22}}$
$S_{A_{23}}$
$S_{A_{31}}$
$S_{A_{32}}$
$S_{A_{33}}$

Figure 4.2. Structure of the two-block split (material failure and strain relaxation) of the source Jacobian (on the left) and of the four-block split (material failure and three small sub-matrices for strain relaxation) of the system (on the right).

a given global scale for the full system and removing them from the matrix, then inverting a reduced system of equations with better conditioning.

An alternative approach, used in this work, consists in constructing an approximate linearisation of the source Jacobian so that it is structured in two independent blocks. One is a scalar equation for ξ obtained by suppressing all the off-diagonal entries of $\mathbf{J}^*$. The other, analogously, can be obtained from removing all dependencies on the damage variable of the strain-relaxation subsystem. This way, one can compute the analytical solution for the two subsystems separately, and since the two blocks are built so that they are independent on each other, these will constitute an *approximate* solution of (4.17). Moreover, being the fracture kinetics subsystem composed of a single scalar equation, the evaluation of the analytical solution can use the standard scalar exponential and division operations, instead of the more delicate procedures for computing matrix exponentials and especially inverting matrices. As an added benefit, if one approximates the relaxation timescale τ_m with a constant value (to be recomputed iteratively), the Jacobian for the strain relaxation subsystem can be evaluated analytically in a very efficient manner, without resorting to numerical differentiation.

In some particular cases, another approximation step can be taken: whenever the off diagonal entries of the distortion matrix are small in comparison to the diagonal ones, many of the entries of the the strain-relaxation source Jacobian become negligible and one can exploit the sparsity of the strain-relaxation Jacobian matrix to further split it into three blocks. In detail, we take the material-damage/strain-relaxation source vector

$$\mathbf{S}(\mathbf{Q}) = \left(S_\xi,\ S_{A_{11}},\ S_{A_{12}},\ S_{A_{13}},\ S_{A_{21}},\ S_{A_{22}},\ S_{A_{23}},\ S_{A_{31}},\ S_{A_{32}},\ S_{A_{33}}\right)^\mathsf{T} \tag{4.36}$$

and rearrange it as four independent sources

$$\mathbf{S}_\mathrm{a} = (S_\xi), \tag{4.37}$$

$$\mathbf{S}_\mathrm{b} = \left(S_{A_{11}},\ S_{A_{22}},\ S_{A_{33}}\right)^\mathsf{T}, \tag{4.38}$$

$$\mathbf{S}_\mathrm{c} = \left(S_{A_{12}},\ S_{A_{13}},\ S_{A_{21}}\right)^\mathsf{T}, \tag{4.39}$$

$$\mathbf{S}_\mathrm{d} = \left(S_{A_{23}},\ S_{A_{31}},\ S_{A_{32}}\right)^\mathsf{T}. \tag{4.40}$$

Then, of each one of the sources, we compute the Jacobian with respect to only the variables in the corresponding block, that is, for example $\mathbf{S}_\mathrm{b}$ will be differentiated only with respect to the diagonal components of $\mathbf{A}$ and all off-diagonal-block contributions to the global Jacobian like $\partial S_{A_{11}}/\partial A_{12}$ will be assumed null. This approximation is justified (under the assumption that off-diagonal components of $\mathbf{A}$ be small, which, albeit not valid as an approximation in fluids, is often verified in elasto-plasticity) for all off-diagonal-block derivatives except for $\partial S_{A_{13}}/\partial A_{31}$ and $\partial S_{A_{31}}/\partial A_{13}$, which maintain a large magnitude even when $\mathbf{A}$ is almost diagonal. These elements can be suppressed regardless, as we already do for all the derivatives like $\partial S_\xi/\partial A_{ij}$ and $\partial S_{A_{ij}}/\partial \xi$, relying on the adaptive timestepping method and on the iterative re-linearisation for the task of reintroducing the lost coupling terms. A visual representation of the two-block split (material failure and strain relaxation) and the four-block split (material failure and three small sub-matrices for strain relaxation) of the source Jacobian is given in Figure 4.2.

4.5 Semi-analytic solution of the strain relaxation source

A final major step forward in the development of a robust solver for the strain relaxation system (4.1), in particular allowing to accurately capture the Navier–Stokes limits regardless of the timestep size, is based on three key observations, discussed in the following.

4.5.1 Observation 1: The splitting approach is not always adequate

In previously discussed techniques [86, 313] for the solution of relaxation processes we have adopted the fractional step (or splitting) approach. The technique is very useful as it allows to separate the solution of the relaxation source from all other dynamics, and attack the resulting ordinary differential equation system with ad hoc techniques. However, the relaxation processes in the unified model of continuum mechanics, besides complex nonlinear dynamics, also feature nontrivial equilibrium states that *must* be reliably captured and preserved. If not, important properties of the continuum model, like the convergence to the Navier–Stokes–Fourier system [120], may be lost in its discrete transposition.

To quantitatively argue the point, consider the thermal impulse equation

$$\partial_t (\mathbf{J}) + \nabla (\mathbf{J} \cdot \mathbf{u} + T) + [\nabla \mathbf{J} - (\nabla \mathbf{J})^{\mathsf{T}}]\, \mathbf{u} = -\mathbf{J}/\tau_{\mathrm{H}}, \tag{4.41}$$

and assume available a generic numerical scheme by which one can compute for each cell/degree of freedom an update $\mathbf{P}_* = (\mathbf{J}^* - \mathbf{J}^n)/\Delta t$ such that $\mathbf{J}^*$ is the solution to the update of the left hand side of (4.41), i.e. the homogeneous system that is solved by application of the splitting approach. In this particular case $\mathbf{P}_*$ can be seen as a discretisation of $-\nabla T$. Then, a straightforward application of the splitting method would find the solution at time $t^{n+1} = t^n + \Delta t$ of

$$\frac{\mathrm{d}\mathbf{J}}{\mathrm{d}\mathbf{t}} = -\frac{1}{\tau_{\mathrm{H}}}\,\mathbf{J}, \quad t \in [t^n,\ t^{n+1}], \quad \mathbf{J}(t^n) = \mathbf{J}^*. \tag{4.42}$$

If Equation (4.42) is integrated via the implicit Euler method, then one can prove that the final value of $\mathbf{J}^{n+1}$ will indeed yield an asymptotic preserving discretization of the PDE (provided obviously that the discretisation of the left hand side is compatible). However, it is easy to see that, isolated from the left hand side of (4.41), the ordinary differential problem (4.42) asymptotically relaxes $\mathbf{J}$ to zero if the relaxation time is sufficiently small with respect to the timestep size, regardless of the value of the initial condition $\mathbf{J}^*$. This implies that if one were to integrate (4.42) *exactly*, then the updated value of the thermal impulse $\mathbf{J}$ would be $\mathbf{J}^{n+1} = \mathbf{0}$ instead of $\mathbf{J}^{n+1} = -\tau_{\mathrm{H}}\, \nabla T$

In order to overcome this issue, a simple modification to the ordinary differential problem (4.42) allows to account for the left hand side of (4.41) and thus converge to the correct asymptotic state $\mathbf{J} = -\tau_{\mathrm{H}}\, \nabla T$ in the stiff limit $\tau_h \to 0$.

An alternative ordinary differential problem to be solved is then

$$\frac{\mathrm{d}\mathbf{J}}{\mathrm{d}t} = \mathbf{P}_* - \frac{1}{\tau_{\mathrm{H}}}\,\mathbf{J}, \quad t \in [t^n,\ t^{n+1}], \quad \mathbf{J}(t^n) = \mathbf{J}^*, \tag{4.43}$$

where, as stated, $\mathbf{P}_*$ accounts for the discrete update from the left hand side of (4.41).

Again, (4.43) can be seen as a system of three uncoupled first order linear ordinary differential equations (ODEs) and an exact solution is indeed found thanks to the linearity and independence of the three equations. Explicitly, the solution is

$$\mathbf{J}^{n+1} = (\mathbf{J}^n - \tau_{\mathrm{H}}\, \mathbf{P}_*)\, \exp(-\Delta t/\tau_{\mathrm{H}}) + \tau_{\mathrm{H}}\, \mathbf{P}_*. \tag{4.44}$$

The only degenerate case to be considered is that if $\Delta t/\tau_{\mathrm{H}}$ is very small (of the order of 10^{-8}), i.e. if the source term is not stiff at all, then (4.44) might yield inaccurate results, due to floating point representation issues. In this case, one may simply opt to to switch to explicit Euler integration, which for such mild (vanishing) sources yields perfectly valid solutions.

4.5.2 Observation 2: The structure of the problem can be significantly simplified by choosing the appropriate reference frames

The nine components of the distortion matrix/basis triad $\mathbf{A}$ encode two different kinds of information: six degrees of freedom are directly linked to the stress tensor $\boldsymbol{\sigma} = -\rho\, c_{\mathrm{s}}^2\, \mathbf{G}\ \mathrm{dev}\, \mathbf{G}$, while the remaining three are associated with an angular orientation which does not influence stresses or energies but are nonetheless part of the structure of the governing equations. This can be formalized by means of the polar decomposition of $\mathbf{A}$, by which we can highlight the six stress-inducing components of $\mathbf{A}$, identifying them as the square root $\mathbf{G}^{1/2}$ of the metric tensor $\mathbf{G}$. Moreover, one can easily see that, if an appropriate *fixed* transformation of the reference frame is applied to (4.1) (a polar decomposition followed by a spectral decomposition), such that at time $t = t^n$ one has $\mathbf{A}$ in diagonal form, and if the convection/production term on the left hand side of (4.1) is null (if the flow field is uniform, i.e. $\nabla\mathbf{u} = \mathbf{0}$, or if formally we want to study the invariance properties of the strain relaxation source), then the diagonality of $\mathbf{A}$ is maintained for all $t \geq t^n$. This means that the relaxation source on the right hand side of (4.1) does not alter the rotational component of $\mathbf{A}$. In the following we establish the notation for the polar decomposition procedure enabling separate treatment of rotational degrees of freedom of $\mathbf{A}$ and volumetric/shear/relaxation effects and provide some formal justification of the validity of the approach.

4.5.2.1 Polar decomposition of the distortion matrix

Given the definition of the metric tensor $\mathbf{G} = \mathbf{A}^{\mathsf{T}}\, \mathbf{A}$, the distortion matrix $\mathbf{A}$ can always be expressed as

$$\mathbf{A} = \mathbf{R}\, \mathbf{G}^{1/2}, \quad \text{with} \quad \mathbf{G}^{1/2} = \mathbf{E}\, \hat{\mathbf{G}}^{1/2}\, \mathbf{E}^{-1}, \tag{4.45}$$

with $\mathbf{R}$ an orthogonal transformation with positive unitary determinant, i.e. a rotation matrix. Numerically, the matrix square root $\mathbf{G}^{1/2}$ can be evaluated by means of the Denman–Beavers algorithm, or alternatively, thanks to the symmetry of $\mathbf{G}$, one can reliably and accurately compute the eigenvectors $\mathbf{E}$ and diagonal form $\hat{\mathbf{G}}$ from the eigen-decomposition of the metric tensor

$$\mathbf{G} = \mathbf{E}\, \hat{\mathbf{G}}\, \mathbf{E}^{-1} \tag{4.46}$$

with the Jacobi eigenvalue algorithm. In the work reported in the present thesis, the latter is indeed the method of choice for the task. This allows, for any given state $\mathbf{A}$, to compute a rotation matrix

$$\mathbf{R} = \mathbf{A}\, \mathbf{G}^{-1/2}, \quad \text{with} \quad \mathbf{G} = \mathbf{A}^{\mathsf{T}}\, \mathbf{A} \tag{4.47}$$

which allows to apply any operation to $\mathbf{G} = \mathbf{A}^{\mathsf{T}}\, \mathbf{A}$, or its square root $\mathbf{G}^{-1/2}$, exploiting their symmetric and positive definite properties, or, temporarily in the solution procedure and locally in space, adopting a simpler form of the governing equations, written in terms of $\mathbf{G}$ instead of $\mathbf{A}$. Then the effects of such operations can be mapped onto $\mathbf{A}$ directly via the rotation $\mathbf{A} = \mathbf{R}_* \, \mathbf{G}^{-1/2}$, with $\mathbf{R}_*$ a rotation matrix that can be computed independently from the nonlinear source terms as $\mathbf{R}_* = \mathbf{A}_* \, \mathbf{G}_*^{-1/2}$, having defined $\mathbf{A}_*$ the distortion matrix obtained from the left hand side of the evolution equation as customarily done per the fractional step method. To summarize, one may first compute $\mathbf{R}_n = \mathbf{A}_n \, \mathbf{G}_n^{-1/2}$, use it to map to an auxiliary frame in which one can easily integrate the source term as applied to the symmetric positive definite metric tensor $\mathbf{G}$, and then map back to $\mathbf{A}$ by means of a *different* rotation matrix $\mathbf{R}_* = \mathbf{A}_* \, \mathbf{G}_*^{-1/2}$, already obtained as a function of the left hand side only.

4.5.2.2 Invariance under strain relaxation of the rotational component of the distortion matrix

We can study the effects of the relaxation source on the distortion matrix $\mathbf{A}$ and the metric tensor $\mathbf{G}$ in isolation from those of flow gradients by formally setting $\nabla \mathbf{u} = \mathbf{0}$ obtaining the uniform flow equations

$$\partial_t \mathbf{A} + (\nabla \, \mathbf{A}) \; \mathbf{u} = -k \, \mathbf{A} \; \operatorname{dev} \mathbf{G}, \tag{4.48}$$

$$\partial_t \mathbf{G} + (\nabla \, \mathbf{G}) \; \mathbf{u} = -2 \, k \, \mathbf{G} \; \operatorname{dev} \mathbf{G}, \tag{4.49}$$

with, $k = 3 \, \tau^{-1} \; (\det \mathbf{G})^{5/6}$, or, in the co-moving reference, equivalently

$$\frac{\mathrm{d}\mathbf{A}}{\mathrm{d}t} = -k \, \mathbf{A} \; \operatorname{dev} \mathbf{G}, \tag{4.50}$$

$$\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} = -2 \, k \, \mathbf{G} \; \operatorname{dev} \mathbf{G}, \tag{4.51}$$

where $\mathrm{d}/\,\mathrm{d}t$ is the customary notation for the total/convective/Lagrangian derivative. In the following paragraphs we derive a set of evolution equations, valid in uniform flow, where only strain relaxation effects can be observed, which are provided as an argument for the separate computation of rotations and volumetric/shear effects in the proposed semi-analytical solver.

Evolution equation for the square root of the metric tensor. The metric tensor equation (4.51) can be mapped to a principal reference frame, by setting $\mathbf{G} = \mathbf{E} \, \hat{\mathbf{G}} \, \mathbf{E}^{-1}$, with $\mathbf{E}$ the matrix of eigenvectors of $\mathbf{G}$ computed at a fixed time $t = t_*$ and $\hat{\mathbf{G}}$ a diagonal matrix defined at any time t by $\hat{\mathbf{G}}(t) = \mathbf{E}^{-1} \, \mathbf{G}(t) \, \mathbf{E}$. Then (4.51) can be rewritten as

$$\frac{\mathrm{d}}{\mathrm{d}t} \left(\mathbf{E} \, \hat{\mathbf{G}} \, \mathbf{E}^{-1} \right) = -2 \, k \, \mathbf{E} \, \hat{\mathbf{G}} \, (\operatorname{dev} \hat{\mathbf{G}}) \, \mathbf{E}^{-1}, \tag{4.52}$$

and, since $\mathbf{E}$ is fixed at time $t = t_*$, we obtain

$$\frac{\mathrm{d}\hat{\mathbf{G}}}{\mathrm{d}t} = -2\,k\,\hat{\mathbf{G}}\,\mathrm{dev}\,\hat{\mathbf{G}}, \tag{4.53}$$

which implies that for a fixed orthonormal transformation $\mathbf{E}$ such that at a given time t_* one has a diagonal $\hat{\mathbf{G}}(t_*) = \mathbf{E}^{-1}\,\mathbf{G}(t_*)\,\mathbf{E}$, the matrix $\hat{\mathbf{G}}$ will remain diagonal for all times, implying that the strain relaxation source does not affect the eigenvectors of $\mathbf{G}$ but only its eigenvalues. Moreover, since in the fixed principal axes reference frame $\hat{\mathbf{G}}$ and its square root are guaranteed to be diagonal at any time t, then one can write

$$\begin{aligned}\frac{\mathrm{d}\hat{\mathbf{G}}}{\mathrm{d}t} = \frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{\mathbf{G}}^{1/2}\,\hat{\mathbf{G}}^{1/2}\right) = \hat{\mathbf{G}}^{1/2}\,\frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{\mathbf{G}}^{1/2}\right) + \frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{\mathbf{G}}^{1/2}\right)\hat{\mathbf{G}}^{1/2} = \\ = 2\,\hat{\mathbf{G}}^{1/2}\,\frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{\mathbf{G}}^{1/2}\right),\end{aligned} \tag{4.54}$$

since, in this specific reference frame, all matrix involved are diagonal and products are commutative. Therefore the evolution equation for the square root of $\hat{\mathbf{G}}$, under the effects of strain relaxation only, and in the principal frame reads

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{\mathbf{G}}^{1/2}\right) = \frac{1}{2}\,\hat{\mathbf{G}}^{-1/2}\,\frac{\mathrm{d}\hat{\mathbf{G}}}{\mathrm{d}t} = -k\,\hat{\mathbf{G}}^{1/2}\;\mathrm{dev}\,\hat{\mathbf{G}}. \tag{4.55}$$

Evolution equation for the rotational component of the distortion matrix. A simple governing equation for the rotational component $\mathbf{R}$ of the distortion matrix $\mathbf{A}$ can be derived, in uniform flow $\nabla\mathbf{u} = \mathbf{0}$, or equivalently under the effect of the strain relaxation source only, by substituting the polar decomposition

$$\mathbf{A} = \mathbf{R}\,\mathbf{G}^{1/2} = \mathbf{R}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,\mathbf{E}^{-1} \tag{4.56}$$

in (4.50), obtaining

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathbf{R}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,\mathbf{E}^{-1}\right) = -k\,\mathbf{R}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,(\mathrm{dev}\,\hat{\mathbf{G}})\,\mathbf{E}^{-1}. \tag{4.57}$$

Then, expanding the derivative, we have

$$\mathbf{R}\,\mathbf{E}\,\frac{\mathrm{d}}{\mathrm{d}t}\left(\hat{\mathbf{G}}^{1/2}\right)\mathbf{E}^{-1} + \frac{\mathrm{d}\mathbf{R}}{\mathrm{d}t}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,\mathbf{E}^{-1} = -k\,\mathbf{R}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,(\mathrm{dev}\,\hat{\mathbf{G}})\,\mathbf{E}^{-1}, \tag{4.58}$$

and by substituting (4.55) in (4.58) we recover

$$\mathbf{R}\,\mathbf{E}\left(-k\,\hat{\mathbf{G}}^{1/2}\;\mathrm{dev}\,\hat{\mathbf{G}}\right)\mathbf{E}^{-1} + \frac{\mathrm{d}\mathbf{R}}{\mathrm{d}t}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,\mathbf{E}^{-1} = -k\,\mathbf{R}\,\mathbf{E}\,\hat{\mathbf{G}}^{1/2}\,(\mathrm{dev}\,\hat{\mathbf{G}})\,\mathbf{E}^{-1}, \tag{4.59}$$

and hence the simple (non)-evolution law for the rotations of $\mathbf{A}$ reads

$$\frac{\mathrm{d}\mathbf{R}}{\mathrm{d}t} = \partial_t \mathbf{R} + (\nabla \mathbf{R})\, \mathbf{u} = \mathbf{0}, \tag{4.60}$$

which means that the rotational component $\mathbf{R}$ is not affected by the relaxation source and can only change due to non-uniformity of the flow, unlike the principal components of the metric tensor $\mathbf{G}$, accounting for volumetric and shear effects.

As early as 2010, in [167], S. Godunov and I. Peshkov, recognised that the structure of the governing equations of the cobasis $\mathbf{A}$ could be exploited to simplify the relaxation procedure. Similarly, in [194, 195], Jackson and Nikiforakis develop and employ an approximate semi-analytical integration technique for the relaxation source of the UMCM, which however is tied to a periodic reset procedure for the cobasis $\mathbf{A}$ and for strain energy, restoring a new local *small deformations* configuration whenever necessary. Hence the motivation for developing a novel integration methodology capable of handling arbitrary deformations, extreme parameter choices potentially varying between $\tau = 10^{20}$ and $\tau = 10^{-14}$ as a function of stress as in [47], without modification to the structure of the PDE system.

4.5.3 Observation 3: Equilibrium states can be computed algebraically without time integration

A final welcome fact about the particular structure of the strain production/relaxation equations (4.1) and (4.2) is that, if the source term is very stiff, one may easily compute the asymptotic solution without necessarily resolving the stiff dynamics: the asymptotic state does not depend on the initial conditions and can be computed by means of a simple and quickly convergent fixed point iteration scheme.

Moreover, from the numerical standpoint, time-dependent closed form solutions of the strain relaxation equations can be less accurate than expected for certain initial conditions, parameter choices, or flow configurations. In such cases, the availability of the solution for the sought equilibrium state is not only a matter of efficiency, but also of accuracy and robustness of the computations.

Additionally, the fixed point iteration allows to prove that the integration scheme for the relaxation source will indeed provide, in the Navier–Stokes limit of the model, the correct unique asymptotic stress tensor, reflecting the convergence limit of the continuum model to the Navier–Stokes equations.

4.6 Semi-analytic solution of the strain relaxation equations with finite relaxation time

The solver for the strain relaxation source is based on the exponential integrator developed in [313] for the computation of diffuse interface fractures and material failure, but exploits in a deeper manner the particular structure of the equation being solved, following the technique presented in [86] for finite-rate pressure relaxation.

We recall that the solver employed in [313] required, in general, the solution of a sequence of a nonhomogeneous nine-by-nine systems of linear ordinary differential equations for the nine independent components of the distortion matrix $\mathbf{A}$, which involves the numerical computation of matrix exponentials and the inversion of the Jacobian matrix of the ODE system. Both these operations constitute delicate tasks in linear algebra that require special care to be carried out in an efficient and accurate manner.

The approach used in this work entirely foregoes the solution of such nine-by-nine systems (six-by-six, in the case of the symmetric tensor $\mathbf{G}$) and the associated linear algebra intricacies. Instead, we compute the analytical solution to one of several different linearised equations that approximate the nonlinear ordinary differential equation

$$\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} = \mathbf{L}_* - \frac{6}{\tau}\left(\det \mathbf{G}\right)^{5/6} \mathbf{G}\ \operatorname{dev} \mathbf{G}, \tag{4.61}$$

while admitting simple solutions that can be computed in a robust fashion. Here with $\mathbf{L}_*$ we denote a *constant convective/productive forcing term* to be given in the following paragraphs, in analogy to the previously defined $\mathbf{P}_*$ discrete time derivative of the thermal impulse.

An important aspect of the scheme is that it avoids fractional-step-type splitting, so that the Navier–Stokes stress tensor and the Fourier heat flux can be recovered regardless of the ratio between the computational timestep size and the relaxation timescales. This means that the global timestep size need not be adjusted to accommodate for the fast dynamics of the relaxation sources. This is achieved by first computing, cell by cell, the update to $\mathbf{G}$ (or to $\mathbf{J}$) associated with the left hand side of (4.2), or of (4.43) for heat conduction, and then including its effects in (4.61), in the form of the constant forcing term $\mathbf{L}_*$. Formally, the first step amounts to computing the solution $\mathbf{G}_* = \mathbf{G}(t^{n+1})$ to the initial value problem

$$\begin{cases} \dfrac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} + (\nabla \mathbf{G})\ \mathbf{u} + \mathbf{G}\,\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathsf{T}}\ \mathbf{G} = \mathbf{0}, \\ \mathbf{G}(t^n) = \mathbf{G}_n. \end{cases} \tag{4.62}$$

In our case, instead of Eq. (4.62), we solve the more general equation for the distortion matrix

$$\begin{cases} \dfrac{\mathrm{d}\mathbf{A}}{\mathrm{d}t} + (\nabla \mathbf{A})\ \mathbf{u} + (\nabla \mathbf{u})\ \mathbf{A} = \mathbf{0}, \\ \mathbf{A}(t^n) = \mathbf{A}_n. \end{cases} \tag{4.63}$$

from which we obtain a pointwise update to the cell averages or degrees of freedom, which is in turn used to define the constant convective/productive forcing term

$$\mathbf{L}_* = \frac{\mathbf{G}_* - \mathbf{G}_n}{\Delta t}, \text{ with } \mathbf{G}_* = \mathbf{A}_*^{\mathsf{T}}\,\mathbf{A}_*,\ \mathbf{G}_n = \mathbf{A}_n^{\mathsf{T}}\,\mathbf{A}_n. \tag{4.64}$$

Then a sub-timestepping loop with adaptive step size δt^m is entered in order to approximate the solution of (4.61) with a sequence of solutions of linearised ODEs. Such a sub-timestepping loop is useful for ensuring the robustness and accuracy of the solver in complex flow configurations, but generally in our computations the solver achieves convergence in one single sub-timestep $\delta t^m = \Delta t$. We refer to [86, 313] for more details on the sub-timestepping approach, and we carry on our presentation of the method by listing three possible approximation choices for the solution of (4.61).

4.6.1 Approximate analytical solution for strain, approach 1

When dealing with fluid flows (that is, when the source term acts on fast timescales), rather often one may assume that $\mathbf{G}$ be a perturbation of a spherical tensor, that is, $\operatorname{dev}\mathbf{G}$ can be assumed small. Then, it is advantageous to rewrite (4.61) as

$$\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} = \mathbf{L}_* - k\ \operatorname{dev}\mathbf{G}\ \operatorname{dev}\mathbf{G} + k\left(\frac{\operatorname{tr}\mathbf{G}}{3}\right)^2 \mathbf{I} - k\,\frac{\operatorname{tr}\mathbf{G}}{3}\,\mathbf{G}, \tag{4.65}$$

with $k = 6\det(\mathbf{G})^{5/6}/\tau$ taken constant for the sub-timestep. This splits the source in four pieces. The first is the constant $\mathbf{L}_*$, associated with convection which, by definition, cannot be stiff as its size is limited by the CFL constraint of the global timestepping scheme. The second is a (small by hypothesis) quadratic term in $\operatorname{dev}\mathbf{G}$ which can be safely approximated as constant. The third is a function of $\mathrm{tr}\,\mathbf{G}$ only, again formally taken constant. This assumption can be justified by writing the evolution equation for the trace of $\mathbf{G}$

$$\frac{\mathrm{d}}{\mathrm{d}t}\,(\mathrm{tr}\,\mathbf{G}) = \mathbf{L}_* - k\ \mathrm{tr}\,(\mathrm{dev}\,\mathbf{G}\ \mathrm{dev}\,\mathbf{G}), \tag{4.66}$$

which shows that either $\mathrm{tr}\,\mathbf{G}$ varies on a timescale associated with convection (by definition, slow), or as a quadratic function of $\mathrm{dev}\,\mathbf{G}$ (small by assumption). The approximate equation (4.65) then admits the simple exact solution

$$\mathbf{G}_{m+1} = \mathbf{G}(t^m + \delta t^m) = \exp\left(-k\,\frac{\mathrm{tr}\,\mathbf{G}}{3}\,\delta t^m\right)(\mathbf{G}_m + \mathbf{F}_0) - \mathbf{F}_0, \tag{4.67}$$

with

$$\mathbf{F}_0 = -\frac{3}{k\ \mathrm{tr}\,\mathbf{G}}\left[\mathbf{L}_* + k\left(\frac{\mathrm{tr}\,\mathbf{G}}{3}\right)^2\mathbf{I} - k\ \mathrm{dev}\,\mathbf{G}\ \mathrm{dev}\,\mathbf{G}\right]. \tag{4.68}$$

We should remark that *nowhere* in this approximate solution we neglected the contributions due to $\mathrm{dev}\,\mathbf{G}$, they only have taken to be constant for a sub-timestep. Specifically, our constant approximations are initially set to $\mathrm{tr}\,\mathbf{G} = \mathrm{tr}\,\mathbf{G}_m$ and $\mathrm{dev}\,\mathbf{G} = \mathrm{dev}\,\mathbf{G}_m$ and then updated within a fixed point iteration as $\mathrm{tr}\,\mathbf{G} = \mathrm{tr}(\mathbf{G}_m + \mathbf{G}_{m+1})/2$ and $\mathrm{dev}\,\mathbf{G} = \mathrm{dev}(\mathbf{G}_m + \mathbf{G}_{m+1})/2$.

4.6.2 Approximate analytical solution for strain, approach 2

Whenever the deviatoric part of $\mathbf{G}$ cannot be assumed small, i.e. in practice when

$$\sqrt{\mathrm{tr}\,(\mathrm{dev}\,\mathbf{G}_m\ \mathrm{dev}\,\mathbf{G}_m)} > (\det\mathbf{G}_m)^{1/3}/5, \tag{4.69}$$

better accuracy in the approximation of (4.61) can be obtained by observing that it is possible to switch the order of the operands of the matrix product $\mathbf{G}\ \mathrm{dev}\,(\mathbf{G})$ appearing in (4.61). Thus we rewrite (4.61) as

$$\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} = \mathbf{L}_* - k\ \mathrm{dev}\,(\mathbf{G})\,\mathbf{G}, \tag{4.70}$$

where we will take $\mathbf{L}_*$, $k = 6\,(\det\mathbf{G})^{5/6}/\tau$, and $\mathrm{dev}\,(\mathbf{G})$ to be constant at each sub-timestep. In order to simplify the solution of (4.70), we work in the principal reference frame which diagonalizes $\mathbf{G}$ and $\mathrm{dev}\,(\mathbf{G})$, i.e. we compute the orthonormal matrix $\mathbf{E}$ such that $\hat{\mathbf{G}} = \mathbf{E}^{-1}\,\mathbf{G}\,\mathbf{E}$ is a diagonal matrix and apply the associated change of basis to all vectors and tensors in our equation. This way the exact solution to Eq. (4.70) is

$$\begin{aligned}\mathbf{G}_{m+1} = \mathbf{G}(t^m + \delta t^m) = \mathbf{E}\,\Big\{&\exp\left(-k\ \mathrm{dev}\,\hat{\mathbf{G}}\,\delta t^m\right)\cdot\\ \cdot\left[\mathbf{E}^{-1}\,\mathbf{G}_n + \frac{1}{k}\left(\mathrm{dev}\,\hat{\mathbf{G}}\right)^{-1}\mathbf{E}^{-1}\,\mathbf{L}_*\right] &- \frac{1}{k}\left(\mathrm{dev}\,\hat{\mathbf{G}}\right)^{-1}\mathbf{E}^{-1}\,\mathbf{L}_*\Big\}.\end{aligned} \tag{4.71}$$

The three-by-three matrix $\mathbf{E}$ having for columns the eigenvectors of $\mathbf{G}$ can be quickly and robustly computed to arbitrary precision by means of Jacobi's method for the eigenstructure of symmetric matrices, and its inverse is simply given by $\mathbf{E}^{-1} = \mathbf{E}^{\mathsf{T}}$. Furthermore, $\operatorname{dev}\hat{\mathbf{G}}$ can be inverted trivially in the principal reference frame by just taking the reciprocal of each diagonal entry. Like for the previous solution we iteratively update the estimate $\operatorname{dev}\hat{\mathbf{G}} = \operatorname{dev}(\hat{\mathbf{G}}_m + \hat{\mathbf{G}}_{m+1})/2$.

Determinant constraint. In the solution of the equation for the metric tensor $\mathbf{G}$, specifically when the computation involves fluid-type behaviour, special care must be paid to preserve the nonlinear algebraic constraint $\det\mathbf{G}(t,\ \mathbf{x}) = [\rho(t,\ \mathbf{x})/\rho_0]^2 = D(t,\ \mathbf{x})$. The constraint must be actively enforced since the discretisation scheme may in principle introduce significant errors that over time could let the solution drift away from a compatible state in which $\det\mathbf{A} \neq \rho/\rho_0$. A simple approach to the problem constitutes in uniformly multiplying all components of $\mathbf{G}$ by $(D/\det\mathbf{G})^{1/3}$ so that the resulting determinant is D.

The specific numerical value of the determinant D is clearly known (from density) at the time levels t^n and t^{n+1}, however it must be somehow approximated for all the in-between times during which we operate our sub-timestepping procedure. In this work we impose that for a given sub-timestep indexed by m, connecting t^m and t^{m+1} the determinant D be computed as

$$D = \beta_s\, D_s + (1 - \beta_s)\, D_f, \tag{4.72}$$

where we define $D_s = \det\left[\mathbf{G} + (t^m + \delta t^m - t^n)\,\mathbf{L}_*\right]$ to be the value that the determinant would have following a linear segment path connecting the two states $\mathbf{G}$ and $\mathbf{G}_*$, that is, the value that would allow preserve exact integration of the (zero) source term in the solid case. Instead, $D_f = \det\mathbf{G} + (t^m + \delta t^m - t^n)\,(\det\mathbf{G}_* - \det\mathbf{G}_n)$ is a second order approximation of the determinant in the fluid limit. The mixing ratio β_s for the two approximations D_s and D_f is an heuristic measure of how close to a solid can the material be considered and its expression is

$$\beta_s = \min\left[1,\ \frac{||\mathbf{L}_*||_2^2}{||6/\tau\,(\det\mathbf{G}_m)^{5/6}\,\mathbf{G}_m\ \operatorname{dev}\mathbf{G}_m||_2^2 + 10^{-14}}\right]^4, \tag{4.73}$$

with $||\mathbf{A}||_2^2$ denoting the square of the Frobenius norm of a given tensor $\mathbf{A}$.

4.6.3 Approximate analytical solution for strain, approach 3: Fixed point iteration for the Navier–Stokes equilibrium state

Oftentimes the timescale τ of strain relaxation is so fast that one may decide to just compute the strain state for which the forcing term due to convection $\mathbf{L}_*$ and the relaxation source are balanced yielding an equilibrium state corresponding to the Navier–Stokes limit of the GPR model. Such an equilibrium state can be easily computed by means of a fixed point iteration in the form

$$\mathbf{G}_{i+1} = \widetilde{\mathbf{G}}_i \left(\frac{D}{\det \widetilde{\mathbf{G}}_i} \right)^{1/3}, \text{ with } \widetilde{\mathbf{G}}_i = \frac{\tau \ \mathrm{dev} \left(\mathbf{G}_i^{-1} \, \mathbf{L}_* \right)}{6 \, (\det \mathbf{G}_i)^{5/6}} + \frac{\mathrm{tr} \, \mathbf{G}_i}{3} \, \mathbf{I}, \tag{4.74}$$

with D the target determinant as defined in (4.72) and i the iteration index in the fixed point procedure. We found that the fixed point iteration (4.74) is always convergent regardless of the initial guess, but nonetheless we care to provide a simple and efficient choice in the form

$$\mathbf{G}_1 = \widetilde{\mathbf{G}}_0 \left(\frac{D}{\det \widetilde{\mathbf{G}}_0} \right)^{1/3}, \text{ with } \widetilde{\mathbf{G}}_0 = \mathbf{I} + \frac{\tau}{6 \ \det(\mathbf{G}_m)^{5/6}} \, \mathrm{dev} \, \mathbf{L}_*. \tag{4.75}$$

Details on the derivation of this fixed point iteration scheme, as well as a proof of convergence are given in Section 4.7

4.6.4 Summary of the selection procedure for the approximation method

At each sub-timestep between t^m and t^{m+1}, our solver for the equation of the elastic metric tensor $\mathbf{G}$ has to select the optimal approximation method for the specific distortion configuration at hand. The selection procedure is carried out as follows:

1. If the source is not stiff, i.e. if $\beta_s > 1 - 10^{-14}$, then we use explicit Euler integration and compute the solution at the next time sub-level $\mathbf{G}_{m+1} = \mathbf{G}_m + \Delta t \left(\mathbf{L}_* - \dfrac{6}{\tau \ \det \left(\mathbf{G}_m \right)^{5/6} \mathbf{G} \ \mathrm{dev} \, \mathbf{G}} \right)$.

2. Else define the indicator matrix $\boldsymbol{\Lambda} = \left| \left(\mathbf{G}_m^{-1} \, \mathbf{L}_* - k \ \mathrm{dev} \, \mathbf{G}_m \right) \right|$ and if the sum of the off-diagonal components of $\boldsymbol{\Lambda}$ is less than $\mathrm{tr} \, \boldsymbol{\Lambda}/5$ and $\delta t^m > \tau$ then the scheme selects the fixed point iteration (4.74).

3. Else if $\sqrt{\mathrm{tr} \left(\mathrm{dev} \, \mathbf{G}_m \ \mathrm{dev} \, \mathbf{G}_m \right)} < (\det \mathbf{G}_m)^{1/3}/5$ or if any of the diagonal entries of $\mathrm{dev} \, \hat{\mathbf{G}}_m$ has magnitude smaller than $\mathrm{tr} \, \hat{\mathbf{G}}_m/1000$ then the scheme uses (4.65).

4. If none of the above, then we apply approximation (4.70).

Regardless of the chosen approximation method, at the end of each sub-timestep, the result $\mathbf{G}_{m+1} = \mathbf{G}(t^{m+1})$ must be multiplied by $(D/\det\mathbf{G}_{m+1})^{1/3}$ so that the determinant constraint is satisfied.

4.7 Equilibrium state for the nonlinear strain relaxation process

In this section, Specifically, we are seeking a steady-state solution to the ordinary differential equation

$$\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} = \mathbf{L}_* - \frac{6}{\tau}\,|\det\mathbf{G}|^{5/6}\;\mathbf{G}\;\operatorname{dev}\mathbf{G}, \tag{4.76}$$

under the algebraic constraint

$$\det\mathbf{G} = D = \frac{\rho^2}{\rho_0^2}, \tag{4.77}$$

which expresses the compatibility of the material distortion given by $\mathbf{A}$ with the mass conservation law, and where we point out that, from the point of view of the semi-analytical integration technique introduced in this Section, the quantity D is a *fixed parameter*, that is obtained directly by the discrete solution of the mass conservation equation. We can include the constraint (4.77) into the ODE (4.76) by writing a modified ODE of the type

$$\frac{\mathrm{d}\mathbf{G}}{\mathrm{d}t} = \mathbf{L}_* - \frac{6}{\tau}\,|\det\mathbf{G}|^{5/6}\;\mathbf{G}\;\operatorname{dev}\mathbf{G} - \frac{1}{\tau_D}\;(\det\mathbf{G} - D)\;\mathbf{G}, \tag{4.78}$$

for an appropriately small value of τ_D, so that the relaxation towards the constraint $\det\mathbf{G} = D$ can be considered instantaneous even with respect to the already very fast strain relaxation process.

4.7.1 Designing a fixed point iteration method for the computation of the steady state solution

The aim of this section is to construct an iterative process of the type

$$\mathbf{G}_{r+1} = f(\mathbf{G}_r;\;\mathbf{L}_*,\;\tau,\;D) \tag{4.79}$$

such that the limit $\mathbf{G}_r,\ r \to \infty$ converges to the steady state solution of (4.78). More specifically, the iterative update formula will take the general form

$$\begin{cases} \widetilde{\mathbf{G}}_{r+1} = \operatorname{dev} \widetilde{\mathbf{G}}_{r+1} + \dfrac{\operatorname{tr} \widetilde{\mathbf{G}}_{r+1}}{3} \, \mathbf{I}, \\ \mathbf{G}_{r+1} = \widetilde{\mathbf{G}}_{r+1} \left(\dfrac{D}{\det \widetilde{\mathbf{G}}_{r+1}} \right)^{1/3}, \end{cases} \tag{4.80}$$

and we will compute, at each iteration r, an estimate for $\operatorname{dev} \widetilde{\mathbf{G}}_{r+1}$ and one for $\operatorname{tr} \widetilde{\mathbf{G}}_{r+1}$. The two-stage form of (4.80), with the second stage being a simple uniform rescaling of all components of $\widetilde{\mathbf{G}}_{r+1}$ is such that each iteration yields a state $\mathbf{G}_{r+1}$ that satisfies the determinant constraint $\det \mathbf{G}_{r+1} = D$ exactly.

At equilibrium, the time derivative of $\mathbf{G}$ will be null, then we can write

$$\mathbf{L}_* - k \, \mathbf{G} \, \operatorname{dev} \mathbf{G} = \frac{1}{\tau_D} \, (\det \mathbf{G} - D) \, \mathbf{G}, \tag{4.81}$$

having defined the shorthand notation $k = 6 \, (\det \mathbf{G})^{5/6} / \tau = 6 \, D^{5/6} / \tau$. From (4.81) we can note that

$$\mathbf{L}_* - k \, \mathbf{G} \, \operatorname{dev} \mathbf{G} = \lambda \, \mathbf{G} \tag{4.82}$$

must hold for some scalar λ. This is nothing but a formal statement that, at steady-state, the combined effects of flow convection, given from $\mathbf{L}_*$, and of strain relaxation, proportional to $\mathbf{G} \operatorname{dev} \mathbf{G}$, must translate to a uniform rescaling of the metric tensor $\mathbf{G}$, with proportionality constant λ.

An estimate for such a proportionality constant can be computed by multiplying from the left (4.82) by $\mathbf{G}^{-1}$ to obtain

$$\mathbf{G}^{-1} \, \mathbf{L}_* - k \, \mathbf{G}^{-1} \, \mathbf{G} \, \operatorname{dev} \mathbf{G} = \lambda \, \mathbf{G}^{-1} \, \mathbf{G} \tag{4.83}$$

and simplifying to

$$\mathbf{G}^{-1} \, \mathbf{L}_* - k \, \operatorname{dev} \mathbf{G} = \lambda \, \mathbf{I} = \mathbf{\Lambda}, \tag{4.84}$$

from which one can take the trace and express λ as

$$\lambda = \frac{\operatorname{tr} \mathbf{\Lambda}}{3} = \frac{\operatorname{tr} \left(\mathbf{G}^{-1} \, \mathbf{L}_* - k \, \operatorname{dev} \mathbf{G} \right)}{3} = \frac{\operatorname{tr} \left(\mathbf{G}^{-1} \, \mathbf{L}_* \right)}{3}. \tag{4.85}$$

At steady state, any one of the three nonzero components of $\mathbf{\Lambda}$ will clearly be exacty equal to λ, as implied by (4.82). However, the looser estimate given by applying (4.85) to any state $\mathbf{G}$ will be useful in the design of the fixed point iteration, which converges to the steady-state value of the metric tensor

$\mathbf{G}$ through a sequence of approximate solutions, each one progressively more accurate than the previous with respect to the exact equilibrium balances (4.78), (4.82), or (4.85), but never computed with a state $\mathbf{G}$ for which $\boldsymbol{\Lambda}$ is necessarily required to be a perfectly spherical matrix.

The estimate of $\operatorname{dev}\widetilde{\mathbf{G}}_{r+1}$ is readily obtained by taking the deviator of both sides of (4.84) and simplifying to

$$\operatorname{dev}\left(\mathbf{G}^{-1}\,\mathbf{L}_*)\right) - k\,\operatorname{dev}\mathbf{G} = \operatorname{dev}(\lambda\,\mathbf{I}) = \mathbf{0}. \tag{4.86}$$

This means, since the deviator of a spherical tensor, such as $\lambda\,\mathbf{I}$, is vanishing, we can compute a very simple estimate, independent from λ itself, for the deviatoric part of the updated metric tensor as

$$\operatorname{dev}\widetilde{\mathbf{G}}_{r+1} = \frac{1}{k}\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right). \tag{4.87}$$

The estimate of $\operatorname{tr}\widetilde{\mathbf{G}}_{r+1}$ is computed in a similar fashion by rearranging (4.82) and applying the trace operator to get

$$\operatorname{tr}\mathbf{G} = \frac{1}{\lambda}\left[\operatorname{tr}\mathbf{L}_* - k\,\operatorname{tr}(\mathbf{G}\,\operatorname{dev}\mathbf{G})\right], \tag{4.88}$$

with λ given by (4.85) as $\lambda = \operatorname{tr}\left(\mathbf{G}^{-1}\,\mathbf{L}_*\right)/3$. If then we further plug (4.87) in (4.88), then the suggested approximation for $\operatorname{tr}\widetilde{\mathbf{G}}_{r+1}$ is

$$\operatorname{tr}\widetilde{\mathbf{G}}_{r+1} = 3\,\frac{\operatorname{tr}\left[\mathbf{L}_* - \mathbf{G}_r\,\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\right]}{\operatorname{tr}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)}, \tag{4.89}$$

which simplifies entirely to

$$\operatorname{tr}\widetilde{\mathbf{G}}_{r+1} = \operatorname{tr}\mathbf{G}_r. \tag{4.90}$$

The fixed point recursion formula for the steady-state equilibrium of the metric tensor $\mathbf{G}$ is then written as

$$\mathbf{G}_{r+1} = \widetilde{\mathbf{G}}_{r+1}\left(\frac{D}{\det\widetilde{\mathbf{G}}_{r+1}}\right)^{1/3}, \tag{4.91}$$

with

$$\widetilde{\mathbf{G}}_{r+1} = \frac{1}{k}\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right) + \frac{\operatorname{tr}\mathbf{G}_r}{3}\,\mathbf{I}. \tag{4.92}$$

It is to be noted that, despite the extreme simplicity of the trace estimate (4.90), and the fact that only a deviatoric tensor (4.87) is added to the spherical tensor

$\mathbf{I}\,\mathrm{tr}\,\mathbf{G}_r/3$ to form $\widetilde{\mathbf{G}}_{r+1}$, the fixed point iteration still features a mechanism to modfiy the trace of the metric tensor at each iteration, by means of the rescaling operation enforcing the determinant constraint $\det \mathbf{G}_{r+1} = D$, and the trace of the metric tensor is not stuck to its first guess value as could initially appear. A very efficient initial guess is obtained by taking $\mathbf{G}_{(-1)} = \mathbf{I}$ and analytically applying one iteration of the algorithm to obtain

$$\mathbf{G}_0 = D^{1/3}\,\frac{\mathbf{I} + k^{-1}\ \mathrm{dev}\,\mathbf{L}_*}{\left[\det\left(\mathbf{I} + k^{-1}\ \mathrm{dev}\,\mathbf{L}_*\right)\right]^{1/3}}, \tag{4.93}$$

which, as will be shown, experimentally proves to be a fairly effective choice.

4.7.2 Proof of convergence to a unique fixed point

In this Section we provide a fomal proof that the fixed point iteration scheme (4.91)–(4.92) for the steady-state asymptotic solution of the strain relaxation equation (4.78) converges to a unique fixed point.

As a notational convenience, we rewrite (4.92) as

$$\widetilde{\mathbf{G}}_{r+1} = \varepsilon\ \mathrm{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right) + \frac{\mathrm{tr}\,\mathbf{G}_r}{3}\,\mathbf{I}, \tag{4.94}$$

having defined the parameter

$$\varepsilon = \frac{1}{k} = \frac{\tau}{6\,D^{5/6}}, \tag{4.95}$$

which can be considered small as it is associated with a fast timescale τ, which is much smaller than the timescale Δt appearing in $\mathbf{L}_*$. If this were not the case, simply we would not be seeking a steady-state solution for $\mathbf{G}$ but rather a transient-state one.

Asymptotic expansion of the fixed point iteration. In order to simplify the calculations, we explicitly write the determinant of $\widetilde{\mathbf{G}}_{r+1}$, which appears at the denominator of (4.91), collecting all the ε-scaled terms so that the a Taylor series can be used to give a simple explicit expression of the fixed point iteration up to quadratic terms in ε. The determinant of $\widetilde{\mathbf{G}}_{r+1}$ is readily expressed as

$$\begin{aligned}\det \widetilde{\mathbf{G}}_{r+1} = {}& \left(\frac{\mathrm{tr}\,\mathbf{G}_r}{3}\right)^3 + \varepsilon\ \mathrm{tr}\left[\mathrm{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\right]\ \left(\frac{\mathrm{tr}\,\mathbf{G}_r}{3}\right)^2 + \\ & + \varepsilon^2\ \left\{\mathrm{tr}\left[\mathrm{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\right]^2 - \mathrm{tr}\left[\mathrm{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)^2\right]\right\}\left(\frac{\mathrm{tr}\,\mathbf{G}_r}{6}\right) + \\ & + \varepsilon^3\ \det\left[\mathrm{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\right],\end{aligned} \tag{4.96}$$

The application of $\operatorname{tr}[\operatorname{dev}(\bullet)] = 0$ simplifies (4.96) into

$$\det \widetilde{\mathbf{G}}_{r+1} = \left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^3 - \varepsilon^2 \operatorname{tr}\left[\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\,\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\right]\left(\frac{\operatorname{tr}\mathbf{G}_r}{6}\right) + \\ + \varepsilon^3 \det\left[\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\right], \tag{4.97}$$

which clearly shows that the determinant of $\widetilde{\mathbf{G}}_{r+1}$ can be approximated as

$$\det \widetilde{\mathbf{G}}_{r+1} = \left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^3 + 0\,\varepsilon + o(\varepsilon) \tag{4.98}$$

since the first order term in the small parameter ε vanishes and terms that are quadratic or cubic in ε are collected in the remainder $o(\varepsilon)$.

Then the division of $\widetilde{\mathbf{G}}_{r+1}$ by $(\det \widetilde{\mathbf{G}}_{r+1})^{1/3}$ appearing in (4.91) can be carried out exploiting the Taylor series expansion (4.98) as

$$\begin{aligned}\mathbf{G}_{r+1} &= D^{1/3}\,\widetilde{\mathbf{G}}_{r+1}\left(\det\widetilde{\mathbf{G}}_{r+1}\right)^{-1/3} = D^{1/3}\,\widetilde{\mathbf{G}}_{r+1}\left[\left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^3 + o(\varepsilon)\right]^{-1/3} = \\ &= D^{1/3}\,\widetilde{\mathbf{G}}_{r+1}\left[\left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^{-3/3} - \frac{0\,\varepsilon}{3}\left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^{-12/3} + o(\varepsilon)\right] = \\ &= D^{1/3}\,\widetilde{\mathbf{G}}_{r+1}\left[\left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^{-1} + o(\varepsilon)\right].\end{aligned} \tag{4.99}$$

Contractive nature of the fixed point iteration. Finally the fixed point iteration can be approximated up to second order terms as

$$\mathbf{G}_{r+1} = \left[\varepsilon\,\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right) + \frac{\operatorname{tr}\mathbf{G}_r}{3}\,\mathbf{I}\right]\left[D^{1/3}\left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^{-1} + o(\varepsilon)\right], \tag{4.100}$$

or equivalently can be expanded into,

$$\mathbf{G}_{r+1} = D^{1/3}\left[\mathbf{I} + \varepsilon\,\operatorname{dev}\left(\mathbf{G}_r^{-1}\,\mathbf{L}_*\right)\left(\frac{\operatorname{tr}\mathbf{G}_r}{3}\right)^{-1}\right] + o(\varepsilon). \tag{4.101}$$

The distance, in the Frobenius norm, between two generic states $\mathbf{G}_r^A$ and $\mathbf{G}_r^B$, at iteration r is

$$d_r = \left\|\mathbf{G}_r^A - \mathbf{G}_r^B\right\|_2, \tag{4.102}$$

while the distance of their images through the next iteration is

$$d_{r+1} = 3\,\varepsilon\,D^{1/3}\left\|\frac{\operatorname{dev}\left[\left(\mathbf{G}_r^A\right)^{-1}\mathbf{L}_*\right]}{\operatorname{tr}\mathbf{G}_r^A} - \frac{\operatorname{dev}\left[\left(\mathbf{G}_r^B\right)^{-1}\mathbf{L}_*\right]}{\operatorname{tr}\mathbf{G}_r^B}\right\|_2 + o(\varepsilon). \quad (4.103)$$

Then we can write the distance ratio

$$\vartheta_{r+1} = \frac{3\,\varepsilon\,D^{1/3}\left\|\frac{\operatorname{dev}\left[\left(\mathbf{G}_r^A\right)^{-1}\mathbf{L}_*\right]}{\operatorname{tr}\mathbf{G}_r^A} - \frac{\operatorname{dev}\left[\left(\mathbf{G}_r^B\right)^{-1}\mathbf{L}_*\right]}{\operatorname{tr}\mathbf{G}_r^B}\right\|_2}{\left\|\mathbf{G}_r^A - \mathbf{G}_r^B\right\|_2} + o(\varepsilon). \quad (4.104)$$

Recalling that the parameter ε is small by assumption, justified by the fact that the timescale it is associated with (the relaxation time τ) is much faster than the timescale of observation Δt contained in $\mathbf{L}_*$, which means that, compared, to ε, both norms in (4.104) are unit-scaled.

Note that, in order to obtain (4.104), we made use of the fact that the determinant of the metric tensor is strictly constrained to $\det\mathbf{G} = D$, with D computed as $D = (\rho/\rho_0)^2$. This means that the enforcement of the determinant constraint $\det\mathbf{G} = (\rho/\rho_0)^2$ is not only fundamental for driving the evolution of the spherical part of the metric tensor $\mathbf{G}$ in the fixed point iteration (4.91)–(4.92), but also an important condition for its proof of convergence.

This then means that $\vartheta_{r+1} < 1$ for sufficiently small ε, and hence, per Banach's Fixed Point Theorem, that the iteration is contractive and convergent to a unique fixed point, namely the Navier–Stokes limit for infinitely small ε, but in general the unique steady-state for a given a forcing term due to convection $\mathbf{L}_*$ in the unified model of continuum mechanics.

5 Semi-implicit, curl-preserving schemes

In this Chapter we present a novel numerical method designed to overcome several of the challenges presented by the equations of the unified model of continuum mechanics.

The scheme is based on a *multiply staggered* Cartesian discretisation introduced in [43] that admits special gradient and curl operators that can be used to evolve sensitive involution-constrained quantities such as the interface field that is used to track material interfaces and compute surface tension forces. This allows to solve the weakly hyperbolic surface tension model given in [286] without curl cleaning procedures, due to the fact that, if curl involutions are enforced exactly, the weakly hyperbolic model and its augmented hyperbolic variants collapse onto the same stable system.

The staggered grid also allows to discretely recover a numerical scheme for the incompressible Navier–Stokes equations in the low Mach limit [205–207, 238, 241], and at the same time eliminate the accuracy and efficiency issues of Godunov-type methods: the timestep restriction due to acoustic waves is lifted at the rather limited cost of solving a sequence of sparse symmetric-positive-definite systems for the scalar pressure at each timestep. Together with a suitable low-dissipative discretisation of convective fluxes, this enables the second-order spatially accurate method to achieve results that are comparable with schemes that formally feature a much higher order of accuracy.

Finally, the method employs a semi-analytical integration technique for the strain relaxation sources of the unified model of continuum mechanics, introduced in Section 4.5, in order to robustly compute viscous forces in flows with high spatial and parametric variability.

5.1 Flux-splitting approach

The semi-implicit curl-preserving numerical method presented in this work is based on a three way split of the governing equations, such that convection, capillarity, strain, and acoustic effects are treated separately, each with an ad-hoc discretisation. In particular: (i) a MUSCL–Hancock scheme with primitive variable reconstruction and positivity preserving limiting is adopted for the solution of the convective part of the system; (*ii*) an implicit staggered conservative scheme is used to compute the unknown pressure field as the solution of a simple discrete wave equation leading to a well-behaved symmetric positive definite system of linear equations. This lifts timestep restrictions due to acoustic waves and preserves the accuracy of the method in the low-Mach regime; (*iii*) the evolution of geometric involution constrained fields associated with material distortion **A** and material interfaces **b** is carried out with ad-hoc discrete differential operators that can preserve the curl involutions of the governing equations exactly up to machine precision.

In sequence, at each timestep, first, the convective update of the conserved variable is computed via a path-conservative MUSCL–Hancock method, and at the same time the quantities **A** and **b**, that are endowed with curl constraints, are evolved in time with a simple two-stage Runge–Kutta scheme, which adopts the semi-analytical solver introduced in Section 4.5 for strain relaxation. Following this, corner fluxes due to viscous forces and capillarity can be computed. Then, a discrete wave equation, derived from a staggered discretisation of the momentum-energy system, can be solved for the unknown scalar pressure and finally, since a robust predictor for all state variables has been obtained, momentum and energy interface fluxes can be computed and used to update the conserved variables to the next time level in a conservative fashion.

The system (2.11) can be written with matrix-vector notation as

$$\partial_t \mathbf{Q} + \nabla \cdot \mathbf{F}(\mathbf{Q}) + \mathbf{B}(\mathbf{Q}) \cdot\, \mathbf{Q} = \mathbf{S}(\mathbf{Q}), \tag{5.1}$$

with the state vector

$$\mathbf{Q} = (\alpha_1 \,\rho_1,\ \alpha_2 \,\rho_2,\ \rho \,\mathbf{u},\ \rho \,E,\ \alpha_1,\ \mathbf{b},\ \mathbf{A})^{\mathsf{T}}, \tag{5.2}$$

a flux tensor $\mathbf{F}(\mathbf{Q})$, a non-conservative product $\mathbf{B}(\mathbf{Q})\,\nabla\mathbf{Q}$, and an algebraic source term $\mathbf{S}(\mathbf{Q})$. As proposed in [43, 104, 109] the flux tensor is *split* into a pressure part, and a convective part, to be treated partially by means of a path-conservative MUSCL–Hancock scheme, partially with a special *compatible* and structure-preserving discretization using a vertex-based staggered grid. Hence, eqn. (5.1) is rewritten as

$$\begin{aligned} \partial_t \mathbf{Q} + \nabla\cdot \Big[\mathbf{F}_c(\mathbf{Q}) + \mathbf{F}_p(\mathbf{Q}) + \mathbf{F}_v(\mathbf{Q})\Big] + \nabla \Big[\mathbf{K}_v(\mathbf{Q})\Big] + \\ + \Big[\mathbf{B}_c(\mathbf{Q}) + \mathbf{B}_v(\mathbf{Q})\Big] \cdot \nabla\mathbf{Q} = \mathbf{S}_p(\mathbf{Q}) + \mathbf{S}_v(\mathbf{Q}), \end{aligned} \tag{5.3}$$

where $\mathbf{F}_c(\mathbf{Q})$ and $\mathbf{B}_c(\mathbf{Q})$ contain the convective terms that will be discretized explicitly; $\mathbf{F}_p(\mathbf{Q})$ are the pressure fluxes that will be discretized implicitly using an edge-based staggered grid. The resulting splitting into pressure and convective fluxes is identical to the flux-vector splitting scheme of Toro and Vázquez-Cendón recently forwarded in [322]. The terms $\mathbf{F}_v(\mathbf{Q})$, $\nabla \mathbf{K}_v(\mathbf{Q})$ and $\mathbf{B}_v(\mathbf{Q}) \cdot \nabla \mathbf{Q}$ are discretized in a structure-preserving manner using an explicit scheme on a vertex-based staggered mesh. The split fluxes read

$$\mathbf{F}_c = \begin{pmatrix} \alpha_1\,\rho_1\,\mathbf{u} \\ \alpha_2\,\rho_2\,\mathbf{u} \\ \rho\,\mathbf{u}\otimes\mathbf{u} \\ (\rho\,E - \rho\,e)\;\mathbf{u} \\ 0 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \; \mathbf{F}_p = \begin{pmatrix} 0 \\ 0 \\ p\,\mathbf{I} \\ (\rho\,e + p)\,\mathbf{u} \\ 0 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \; \mathbf{F}_v = \begin{pmatrix} 0 \\ 0 \\ -\boldsymbol{\Sigma}_{\mathrm{t}} - \boldsymbol{\Sigma}_{\mathrm{s}} \\ (-\boldsymbol{\Sigma}_{\mathrm{t}} - \boldsymbol{\Sigma}_{\mathrm{s}})\;\mathbf{u} \\ 0 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \tag{5.4}$$

The convective part of the non-conservative product is given by

$$\mathbf{B}_c(\mathbf{Q})\,\nabla\mathbf{Q} = (0,\; 0,\; \mathbf{0},\; 0,\; \mathbf{u}\cdot\nabla\alpha_1 - K\,\nabla\cdot\mathbf{u},\; \mathbf{0},\; \mathbf{0})^{\mathsf{T}}\,, \tag{5.5}$$

The evolution of the curl-free vector field $\mathbf{b}$ and the distortion matrix $\mathbf{A}$ is governed by the terms $\nabla \mathbf{K}_v(\mathbf{Q})$, $\mathbf{B}_v(\mathbf{Q})\,\nabla\mathbf{Q}$, and $\mathbf{S}_v(\mathbf{Q})$, with

$$\mathbf{K}_v(\mathbf{Q}) = \begin{pmatrix} 0 \\ 0 \\ \mathbf{0} \\ 0 \\ 0 \\ \mathbf{b}\cdot\mathbf{u} \\ \mathbf{A}\,\mathbf{u} \end{pmatrix}, \quad \mathbf{B}_v(\mathbf{Q})\,\nabla\mathbf{Q} = \begin{pmatrix} 0 \\ 0 \\ \mathbf{0} \\ 0 \\ 0 \\ (\nabla\mathbf{b} - \nabla\mathbf{b}^{\mathsf{T}})\;\mathbf{u} \\ (\nabla\mathbf{A} - \nabla\mathbf{A}^{\mathsf{T}})\;\mathbf{u} \end{pmatrix}, \tag{5.6}$$

and

$$\mathbf{S}_v(\mathbf{Q}) = (0,\; 0,\; \mathbf{0},\; 0,\; 0,\; \mathbf{0},\; \mathbf{Z})^{\mathsf{T}}\,, \tag{5.7}$$

which accounts for the strain relaxation effects. For clarity, we specify that the tensor notation $(\nabla\mathbf{A} - \nabla\mathbf{A}^{\mathsf{T}})\;\mathbf{u}$ yields a three by three matrix whose rows $\mathbf{a}_k$ are the obtained by computing $(\nabla\mathbf{a}_k - \nabla\mathbf{a}_k^{\mathsf{T}})\;\mathbf{u}$ for each row $\mathbf{a}_k$ from $\mathbf{a}_1$ to $\mathbf{a}_3$. The remaining source terms, corresponding to gravity forces, are included in the pressure sub-system as

$$\mathbf{S}_p(\mathbf{Q}) = (0,\; 0,\; \mathbf{g},\; \mathbf{g}\cdot\mathbf{u},\; 0,\; \mathbf{0},\; \mathbf{0})^{\mathsf{T}}\,. \tag{5.8}$$

As already mentioned before, the subsystem

$$\partial_t\mathbf{Q} + \nabla\cdot\Big[\mathbf{F}_c(\mathbf{Q}) + \mathbf{F}_v(\mathbf{Q})\Big] + \nabla\Big[\mathbf{K}_v(\mathbf{Q})\Big] + \Big[\mathbf{B}_c(\mathbf{Q}) + \mathbf{B}_v(\mathbf{Q})\Big]\;\nabla\mathbf{Q} = \mathbf{S}_v(\mathbf{Q}), \tag{5.9}$$

will be discretized explicitly. The discretization method presented in the next section is a combination of a classical second order MUSCL–Hancock type [317, 326, 327] TVD finite volume scheme for the convective fluxes $\mathbf{F}_c$ and the nonconservative term $\mathbf{B}_c \nabla\mathbf{Q}$, a curl-free discretization for the terms $\mathbf{K}_v$, and $\mathbf{B}_v \nabla\mathbf{Q}$ using compatible gradient and curl operators, a semi-analytical integration technique for the relaxation source $\mathbf{S}_v$, as well as a vertex-based discretization of the terms $\mathbf{F}_v$. The pressure subsystem

$$\partial_t \mathbf{Q} + \nabla \cdot \mathbf{F}_p(\mathbf{Q}) = \mathbf{S}_p(\mathbf{Q}) \tag{5.10}$$

is formally identical to the Toro–Vázquez pressure system [322], with the simple addition of gravity sources $\mathbf{S}_p$ and is discretized implicitly.

5.2 Explicit discretisation of the convective subsystem

5.2.1 Data reconstruction and slope limiting

In order to achieve second order spatial accuracy for the convective fluxes, a data reconstruction yielding a piecewise first-degree polynomial representation of the state variables, denoted $\mathbf{W}_{ij}(x,\ y)$, must be carried out. For our PDE system, it is convenient to compute such a reconstruction in the primitive variable space, specified by choosing as a primitive state vector

$$\mathbf{V} = (\rho_1,\ \rho_2,\ \mathbf{u},\ p,\ \alpha_1,\ \mathbf{b},\ \mathbf{A})^{\mathsf{T}}, \tag{5.11}$$

which is related to the conserved state $\mathbf{Q}$ by

$$\mathbf{V}(x,\ y) = \mathcal{P}\left[\mathbf{Q}(x,\ y)\right], \text{ and } \mathbf{Q}(x,\ y) = \mathcal{C}\left[\mathbf{V}(x,\ y)\right]. \tag{5.12}$$

Then we denote the primitive-variable reconstruction polynomial as $\mathbf{W}^{\mathrm{p}}(x,\ y) = \mathcal{P}\left[\mathbf{W}(x,\ y)\right]$, and, complementarily, the conserved variable reconstruction polynomial is $\mathbf{W}(x,\ y) = \mathcal{C}\left[\mathbf{W}^{\mathrm{p}}(x,\ y)\right]$. For the sake of clarity, we remark that the primitive-to-conserved operator $\mathcal{C}$ and conserved-to-primitive operator $\mathcal{P}$, due to their nonlinear nature, are to be read as *pointwise* operations, and the evaluation points will be explicitly stated in the following whenever conversion of state variables is necessary.

For each cell of index i, the left and right differences are computed in the primitive variable space as $\Delta\mathbf{V}_L = \mathbf{V}_i - \mathbf{V}_{i-1}$ and $\Delta\mathbf{V}_R = \mathbf{V}_{i+1} - \mathbf{V}_i$ respectively. They are then combined in a nonlinear fashion to ensure non-oscillatory property of the resulting scheme. In particular, we employ a simple slope limiter that can be computed as

$$\widetilde{\Delta \mathbf{V}}_i = \frac{\Delta \mathbf{V}_R \max\left[0,\ \min\left(\beta\,\Delta \mathbf{V}_R^2,\ \Delta \mathbf{V}_R\,\Delta \mathbf{V}_L\right)\right]}{2\,\Delta \mathbf{V}_R^2 + \varepsilon^2} + \frac{\Delta \mathbf{V}_L \max\left[0,\ \min\left(\beta\,\Delta \mathbf{V}_L^2,\ \Delta \mathbf{V}_L\,\Delta \mathbf{V}_R\right)\right]}{2\,\Delta \mathbf{V}_L^2 + \varepsilon^2}, \tag{5.13}$$

where $\varepsilon = 10^{-14}$ is a small constant that avoids division by zero and all operations are to be taken componentwise.

The slope limiter (5.13) yields the minmod slope for $\beta = 1$ and reduces to the MUSCL–Barth–Jespersen limiter for $\beta = 3$. In all our numerical tests we set $\beta = 2$.

The preliminary (undivided) slope $\widetilde{\Delta \mathbf{V}}_i$ is then corrected to enforce the positivity of the reconstructed values of density and pressure, as well as the unit-sum constraints on the volume fractions α_1 and α_2. This is achieved by rescaling the slope with

$$\Delta \mathbf{V}_i = \widetilde{\Delta \mathbf{V}}_i \min\left(1,\ \varphi_i^+,\ \varphi_i^-\right), \tag{5.14}$$

having set

$$\begin{aligned} \varphi_i^+ &= \frac{\left[\left(|\widetilde{\Delta \mathbf{V}}_i| + \widetilde{\Delta \mathbf{V}}_i\right)(\mathbf{V}_{\max} - \mathbf{V}_i) + \left(|\widetilde{\Delta \mathbf{V}}_i| - \widetilde{\Delta \mathbf{V}}_i\right)(\mathbf{V}_{\min} - \mathbf{V}_i)\right]\widetilde{\Delta \mathbf{V}}_i}{2\,|\widetilde{\Delta \mathbf{V}}_i^3| + \varepsilon^3}, \\ \varphi_i^- &= \frac{\left[\left(|\widetilde{\Delta \mathbf{V}}_i| - \widetilde{\Delta \mathbf{V}}_i\right)(\mathbf{V}_i - \mathbf{V}_{\max}) + \left(|\widetilde{\Delta \mathbf{V}}_i| + \widetilde{\Delta \mathbf{V}}_i\right)(\mathbf{V}_i - \mathbf{V}_{\min})\right]\widetilde{\Delta \mathbf{V}}_i}{2\,|\widetilde{\Delta \mathbf{V}}_i^3| + \varepsilon^3}, \end{aligned} \tag{5.15}$$

where, with reference to the primitive variable state vector (5.11), we have set the lower and upper bounds for each variable as

$$\begin{aligned} \mathbf{V}_{\min} &= (0,\ 0,\ -\mathbf{h},\ 0,\ 0,\ -\mathbf{h},\ -\mathbf{H})^{\mathsf{T}}, \\ \mathbf{V}_{\max} &= (H,\ H,\ \mathbf{h},\ H,\ 1,\ \mathbf{h},\ \mathbf{H})^{\mathsf{T}}. \end{aligned} \tag{5.16}$$

The values of H, $\mathbf{h}$, and $\mathbf{H}$ are set to a large arbitrary scalar, vector or matrix (like $H = 10^{40}$) to represent the absence of an upper or lower bound for the corresponding variable. The same sequence of operations is carried out in the y-direction to compute $\Delta \mathbf{V}_j$. Then the primitive reconstruction polynomial can be evaluated at any point in space as

$$\mathbf{W}_{ij}^{\mathrm{p}}(x,\ y) = \mathbf{V}_{ij} + (x - x_{ij})\,\frac{\Delta \mathbf{V}_i}{\Delta x} + (y - y_{ij})\,\frac{\Delta \mathbf{V}_j}{\Delta y}. \tag{5.17}$$

5.2.2 Computation of convective fluxes

The convective terms are explicitly integrated by means of a path-conservative MUSCL–Hancock scheme. The fully discrete one-step update formula reads

$$\begin{aligned}\mathbf{Q}_{ij}^{n+1} = \mathbf{Q}_{ij}^{n} &- \frac{\Delta t}{\Delta x}\left(\mathbf{F}^{\mathrm{RS}}_{i+1/2,\,j} - \mathbf{F}^{\mathrm{RS}}_{i-1/2,\,j} + \mathbf{D}^{+}_{i-1/2,\,j} + \mathbf{D}^{-}_{i+1/2,\,j}\right) + \\ &- \frac{\Delta t}{\Delta y}\left(\mathbf{F}^{\mathrm{RS}}_{i,\,j+1/2} - \mathbf{F}^{\mathrm{RS}}_{i,\,j-1/2} + \mathbf{D}^{+}_{i,\,j-1/2} + \mathbf{D}^{-}_{i,\,j+1/2}\right) + \\ &- \frac{\Delta t}{\Delta x}\,\mathbf{B}_1^{\mathrm{p}}\left[\mathbf{v}_{ij}(t^{n+1/2},\ x_i,\ y_j)\right]\,\Delta\mathbf{V}_i + \\ &- \frac{\Delta t}{\Delta y}\,\mathbf{B}_2^{\mathrm{p}}\left[\mathbf{v}_{ij}(t^{n+1/2},\ x_i,\ y_j)\right]\,\Delta\mathbf{V}_j + \\ &+ \Delta t\,\mathbf{S}\left[\mathbf{v}_{ij}\left(t^{n+1/2},\ x_i,\ y_j\right)\right],\end{aligned} \tag{5.18}$$

which is then applied to the convective subsystem and used to formally define a convective state $\mathbf{Q}_{ij}^{*} = \mathbf{Q}_{ij}^{n+1}$, which in particular is

$$\mathbf{Q}_{ij}^{*} = \left((\alpha_1\,\rho_1)^*_{ij},\ (\alpha_2\,\rho_2)^*_{ij},\ (\rho\,\mathbf{u})^*_{ij},\ (\rho\,E)^*_{ij},\ (\alpha_1)^*_{ij},\ (\mathbf{b})^*_{ij},\ (\mathbf{A})^*_{ij}\right)^{\mathsf{T}} \tag{5.19}$$

For the computation of the conservative numerical fluxes we employ the simple Rusanov Riemann solver

$$\begin{aligned}\mathbf{F}^{\mathrm{RS}}_{i+1/2,\,j}(\mathbf{v}_L,\ \mathbf{v}_R) &= \frac{1}{2}\left[\mathbf{F}_1(\mathbf{v}_L) + \mathbf{F}_1(\mathbf{v}_R)\right] - \frac{1}{2}\,s_1^{\max}\left[\mathcal{C}(\mathbf{v}_R) - \mathcal{C}(\mathbf{v}_L)\right], \\ \mathbf{F}^{\mathrm{RS}}_{i,\,j+1/2}(\mathbf{v}_L,\ \mathbf{v}_R) &= \frac{1}{2}\left[\mathbf{F}_2(\mathbf{v}_L) + \mathbf{F}_2(\mathbf{v}_R)\right] - \frac{1}{2}\,s_2^{\max}\left[\mathcal{C}(\mathbf{v}_R) - \mathcal{C}(\mathbf{v}_L)\right],\end{aligned} \tag{5.20}$$

where the signal speed estimates $s_1^{\max}$ and $s_2^{\max}$, i.e. the maximum absolute eigenvalues of the convective subsystem, in the first or second space direction, associated with a pair of states $\mathbf{v}_L$ and $\mathbf{v}_R$, are given by

$$\begin{aligned}s_1^{\max} = s_1^{\max}(\mathbf{v}_L,\ \mathbf{v}_R) = \max\left[\,\max\left(|\boldsymbol{\lambda}_1(\mathbf{v}_L)\,|\right),\ \max\left(|\boldsymbol{\lambda}_1(\mathbf{v}_R)\,|\right)\right], \\ s_2^{\max} = s_2^{\max}(\mathbf{v}_L,\ \mathbf{v}_R) = \max\left[\,\max\left(|\boldsymbol{\lambda}_2(\mathbf{v}_L)\,|\right),\ \max\left(|\boldsymbol{\lambda}_2(\mathbf{v}_R)\,|\right)\right].\end{aligned} \tag{5.21}$$

An important consideration is that, in order to achieve a compatible discretisation of density, momentum, and kinetic energy in uniform flows, the jump of conserved variables $\mathcal{C}(\mathbf{v}_R) - \mathcal{C}(\mathbf{v}_L)$ must *exclude* the difference of internal energies. Thus, instead of $\Delta E = \rho\,E(\mathbf{v}_R) - \rho\,E(\mathbf{v}_L)$, the jump in the energy conservation equation will be

$$\begin{aligned}\Delta E'(\mathbf{v}_L,\ \mathbf{v}_R) &= \left[\rho\,E|_{\mathbf{v}_R} - \rho\,e|_{\mathbf{v}_R}\right] - \left[\rho\,E|_{\mathbf{v}_L} - \rho\,e|_{\mathbf{v}_R}\right] = \\ &= \rho\,(e_{\mathrm{k}} + e_{\mathrm{s}} + e_{\mathrm{t}})|_{\mathbf{v}_R} - \rho\,(e_{\mathrm{k}} + e_{\mathrm{s}} + e_{\mathrm{t}})|_{\mathbf{v}_L},\end{aligned} \tag{5.22}$$

where we recall $e_\mathrm{k} = \|\mathbf{u}\|^2/2$, $e_\mathrm{s} = c_\mathrm{s}^2\,\mathrm{tr}\,(\mathrm{dev}\,\mathbf{G}\ \mathrm{dev}\,\mathbf{G})/4$, $e_\mathrm{t} = \sigma\,\|\mathbf{b}\|/\rho$.

Nonconservative products are discretised within the path-conservative formalism [68, 249]. This means that, at each cell interface, generically indexed with $i+1/2, j$, in addition to numerical fluxes, two so-called fluctuations, denoted $\mathbf{D}^-_{i+1/2,j}\,(\mathbf{v}_L,\ \mathbf{v}_R)$ and $\mathbf{D}^+_{i+1/2,j}\,(\mathbf{v}_L,\ \mathbf{v}_R)$, have to be computed. Our choice of for the discrete jump terms (fluctuations) is such that, the left and right fluctuations have the same value and may be denoted as $\mathbf{D}_{i+1/2,j}\,(\mathbf{v}_L,\ \mathbf{v}_R)$. The same holds in the y-direction for $\mathbf{D}^-_{i,j+1/2}\,(\mathbf{v}_L,\ \mathbf{v}_R)$ and $\mathbf{D}^+_{i,j+1/2}\,(\mathbf{v}_L,\ \mathbf{v}_R)$. The fluctuations are computed with a three-point point quadrature rule over a segment path $\boldsymbol{\Psi}(\mathbf{v}_L,\ \mathbf{v}_R,\ s) = (1-s)\,\mathbf{v}_L + s\,\mathbf{v}_R$ in the primitive state space. Their discrete expressions read

$$\begin{aligned}
\mathbf{D}_{i+1/2,j}\,(\mathbf{v}_L,\ \mathbf{v}_R) &= \frac{1}{2}\sum_{k=1}^{3}\omega_k\,\mathbf{B}_1^\mathrm{p}\left[\boldsymbol{\Psi}\,(\mathbf{v}_L,\ \mathbf{v}_R,\ s_k)\right]\,(\mathbf{v}_R - \mathbf{v}_L)\,,\\
\mathbf{D}_{i,j+1/2}\,(\mathbf{v}_L,\ \mathbf{v}_R) &= \frac{1}{2}\sum_{k=1}^{3}\omega_k\,\mathbf{B}_2^\mathrm{p}\left[\boldsymbol{\Psi}\,(\mathbf{v}_L,\ \mathbf{v}_R,\ s_k)\right]\,(\mathbf{v}_R - \mathbf{v}_L)\,,
\end{aligned}\tag{5.23}$$

where the nonconservative products for the convective subsystem are given by

$$\begin{aligned}
\mathbf{B}_1^\mathrm{p}(\mathbf{v})\,\Delta\mathbf{V}_i &= (0,\ 0,\ \mathbf{0},\ 0,\ u_1(\mathbf{v})\,(\Delta\alpha_1)_i - K(\mathbf{v})\,(\Delta u_1)_i,\ \mathbf{0},\ \mathbf{0})^\mathsf{T}\,,\\
\mathbf{B}_2^\mathrm{p}(\mathbf{v})\,\Delta\mathbf{V}_j &= (0,\ 0,\ \mathbf{0},\ 0,\ u_2(\mathbf{v})\,(\Delta\alpha_1)_j - K(\mathbf{v})\,(\Delta u_2)_j,\ \mathbf{0},\ \mathbf{0})^\mathsf{T}\,,
\end{aligned}\tag{5.24}$$

with $K = \left(\rho_2\,a_2^2 - \rho_1\,a_1^2\right)\,\alpha_1\,\alpha_2/(\alpha_1\,\rho_2\,a_2^2 + \alpha_2\,\rho_1\,a_1^2)$. For clarity, we explicitly give also the expressions for the fluxes of the convective subsystem

$$\mathbf{F}_1 = \begin{pmatrix} \alpha_1\,\rho_1\,u_1 \\ \alpha_2\,\rho_2\,u_1 \\ \rho\,\mathbf{u}\,u_1 \\ (\rho\,E - \rho\,e)\ u_1 \\ 0 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix},\quad \mathbf{F}_2 = \begin{pmatrix} \alpha_1\,\rho_1\,u_2 \\ \alpha_2\,\rho_2\,u_2 \\ \rho\,\mathbf{u}\,u_2 \\ (\rho\,E - \rho\,e)\ u_2 \\ 0 \\ \mathbf{0} \\ \mathbf{0} \end{pmatrix},\tag{5.25}$$

while the source term is simply $\mathbf{S} = \mathbf{0}$.

At each cell interface, of generic index $i+\frac{1}{2}, j$ in the x-direction or $i, j+\frac{1}{2}$ in the y-direction, the boundary-extrapolated states $\mathbf{v}_L$ and $\mathbf{v}_R$ are taken from a cell-local space-time predictor solution $\mathbf{v}_{ij}(t,\ x,\ y)$. In particular, the space-time midpoint values for each face are

$$\begin{aligned}
(\mathbf{v}_L)_{i+1/2,j} &= \mathbf{v}_{i,j}(t^{n+1/2},\ x_{i+1/2},\ y_j),\\
(\mathbf{v}_R)_{i+1/2,j} &= \mathbf{v}_{i+1,j}(t^{n+1/2},\ x_{i+1/2},\ y_j),\\
(\mathbf{v}_L)_{i,j+1/2} &= \mathbf{v}_{i,j}(t^{n+1/2},\ x_i,\ y_{j+1/2}),\\
(\mathbf{v}_R)_{i,j+1/2} &= \mathbf{v}_{i,j+1}(t^{n+1/2},\ x_i,\ y_{j+1/2}),
\end{aligned} \tag{5.26}$$

and they are explicitly computed as

$$\begin{aligned}
\mathbf{v}_{ij}(t^{n+1/2},\ x,\ y) &= \mathcal{P}\left(\mathbf{q}_{ij}(t^{n+1/2},\ x,\ y)\right)\\
&= \mathcal{P}\left[\mathcal{C}\left(\mathbf{W}^{\mathrm{p}}_{ij}(x,\ y)\right) + \Delta\mathbf{Q}_{ij}\right],
\end{aligned} \tag{5.27}$$

where

$$\begin{aligned}
\Delta\mathbf{Q}_{ij} = &-\frac{\Delta t}{2\,\Delta x}\left[\mathbf{F}_1\left(\mathbf{W}^{\mathrm{p}}_{ij}(x_{i+1/2},\ y_j)\right) - \mathbf{F}_1\left(\mathbf{W}^{\mathrm{p}}_{ij}(x_{i-1/2},\ y_j)\right)\right] +\\
&-\frac{\Delta t}{2\,\Delta y}\left[\mathbf{F}_2\left(\mathbf{W}^{\mathrm{p}}_{ij}(x_i,\ y_{j+1/2})\right) - \mathbf{F}_2\left(\mathbf{W}^{\mathrm{p}}_{ij}(x_i,\ y_{j-1/2})\right)\right] +\\
&-\frac{\Delta t}{2\,\Delta x}\mathbf{B}^{\mathrm{p}}_1\left[\mathbf{W}^{\mathrm{p}}_{ij}(x_i,\ y_j)\right]\Delta\mathbf{V}_i - \frac{\Delta t}{2\,\Delta y}\mathbf{B}^{\mathrm{p}}_2\left[\mathbf{W}^{\mathrm{p}}_{ij}(x_i,\ y_j)\right]\Delta\mathbf{V}_j +\\
&+\frac{\Delta t}{2}\mathbf{S}\left[\mathbf{W}^{\mathrm{p}}_{ij}(x_i,\ y_j)\right].
\end{aligned} \tag{5.28}$$

For the sake of clarity, it should be pointed out that the primitive-to-conserved and conserved-to-primitive conversion operators in Eq. (5.27) are to be read as *pointwise* operations, or equivalently the formula can be taken as a projection between two different manifolds, but it is not *strictly* satisfied in any point except those where the conversion of state variables has taken place, i.e. the space-time barycenters of each face.

5.3 Staggered mesh and discrete divergence, curl and gradient operators

The numerical scheme is presented in a two-dimensional context, however it is necessary and beneficial to retain all components of three-dimensional vectors to simplify the treatment of the relaxation source term, which acts on all components of the distortion matrix even regardless of whether derivatives in any direction vanish or not. Again, we consider a two-dimensional physical domain Ω covered by a set of uniformly sized and non-overlapping Cartesian control volumes $\Omega_{ij} = [x_{i-1/2},\ x_{i+1/2}] \times [y_{j-1/2},\ y_{j+1/2}]$ with mesh spacings $\Delta x = x_{i+1/2} - x_{i-1/2}$ and $\Delta y = y_{j+1/2} - y_{j-1/2}$ in x and y direction respectively. With $x_{i\pm1/2} = x_i \pm \Delta x/2$ and $y_{j\pm1/2} = y_j \pm \Delta y/2$, we denote the barycenter coordinates of the control volumes. We will furthermore use the notation $\hat{\mathbf{e}}_x = (1,\ 0,\ 0)$, $\hat{\mathbf{e}}_y = (0,\ 1,\ 0)$ and $\hat{\mathbf{e}}_z = (0,\ 0,\ 1)$ for the unit vectors pointing into the directions of the Cartesian coordinate axes.

The set of discrete times will be denoted by t^n. For a sketch of the employed staggered grid arrangement of the main quantities, see Fig. 5.2 and Fig. 5.1.

The main ingredients of the structure-preserving staggered semi-implicit scheme proposed in this Chapter are the definitions of appropriate discrete divergence, gradient and curl operators acting on quantities that are arranged in different and judiciously chosen locations on the staggered mesh. The discrete pressure field at time t^n is denoted by p_{ij} and its degrees of freedom are located in the center of each control volume as $p_{ij}^n = p(t^n, x_i,\ y_j)$.

The discrete velocities u_1^n and u_2^n are arranged in an edge-based staggered fashion, i.e. $(u_1^n)_{i+1/2}^j = u_1(t^n,\ x_{i+1/2},\ y_j)$ and $(u_2^n)_i^{j+1/2} = u_2(t^n,\ x_i,\ y_{j+1/2})$. The discrete vector field $\mathbf{b}^n$ is defined on the *vertices* of each spatial control volume as $\mathbf{b}_{i+1/2,j+1/2}^n = \mathbf{b}(t^n,\ x_{i+1/2},\ y_{j+1/2})$. For clarity, see again Fig. 5.2.

The *discrete divergence operator*, $\nabla_h\cdot$, acting on a discrete vector field $\mathbf{u}^n$ is

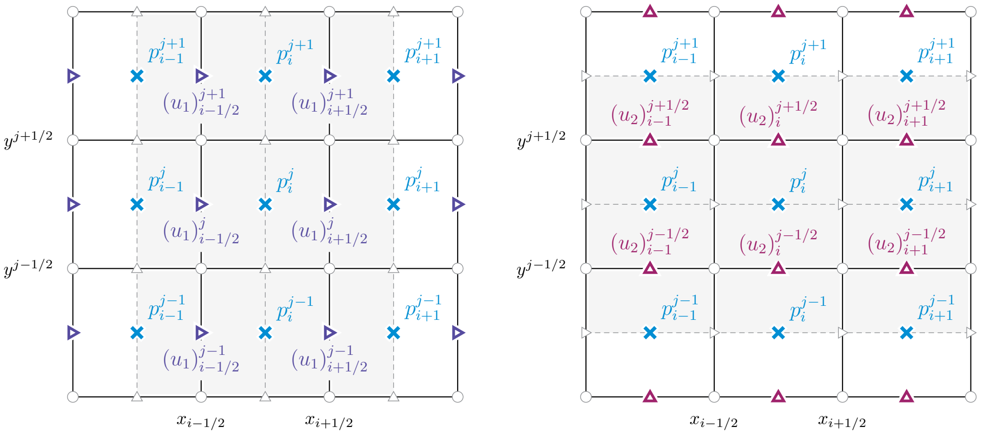


Figure 5.1. Staggered mesh configuration with the pressure field p_i^j defined in the cell barycenters and the velocity field components $(u_1)_{i+1/2}^j$ and $(u_2)_i^{j+1/2}$ defined on the edge-based staggered dual grids.

abbreviated by $\nabla_h \cdot \mathbf{u}^n$ and its degrees of freedom are given by

$$\nabla_h^{i,j} \cdot \mathbf{u}^n = (\nabla_h \cdot \mathbf{u}^n)_{i,j} = \frac{(u_1^n)_{i+1/2}^{j} - (u_1^n)_{i-1/2}^{j}}{\Delta x} + \frac{(u_2^n)_{i}^{j+1/2} - (u_2^n)_{i}^{j-1/2}}{\Delta y}, \tag{5.29}$$

i.e. it is based on the *edge-based* staggered values of the field $\mathbf{u}^n$. It defines a discrete divergence on the control volume Ω_{ij} via the Gauss theorem,

$$\nabla_h \cdot \mathbf{u}^n = \frac{1}{\Delta x\, \Delta y} \int\limits_{\Omega_{ij}} \nabla \cdot \mathbf{u}\ \mathrm{d}\mathbf{x} = \frac{1}{\Delta x\, \Delta y} \int\limits_{\partial\Omega_{ij}} \mathbf{u} \cdot \hat{\mathbf{n}}\ \mathrm{d}S, \tag{5.30}$$

based on the mid-point rule for the computation of the integrals along each edge of Ω_{ij}. In (5.30) the outward pointing unit normal vector to the boundary $\partial\Omega_{ij}$ of Ω_{ij} is denoted by $\hat{\mathbf{n}}$.

In a similar manner, the z component of the *discrete curl*, $\nabla_h \times$, of a discrete vector field $\mathbf{b}^n$ (or $\mathbf{a}_1^n = (1,\ 0,\ 0)\, \mathbf{A}^n$ for example) is denoted by $(\nabla_h \times \mathbf{b}^n) \cdot \hat{\mathbf{e}}_z$

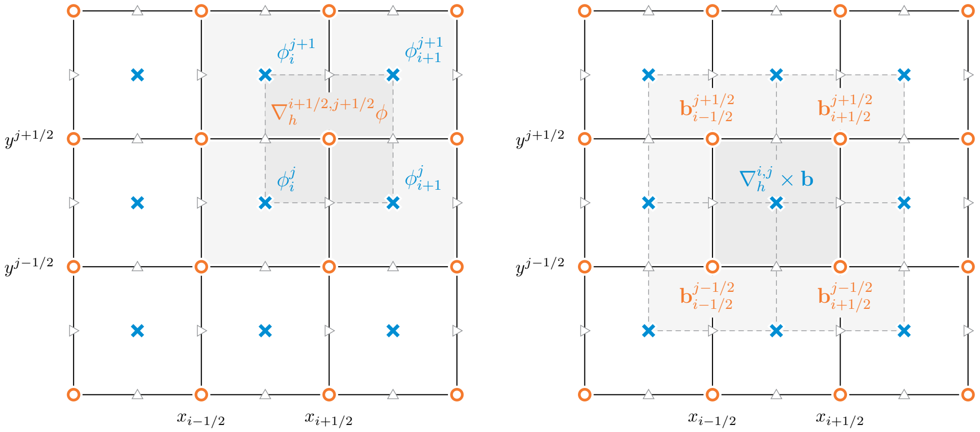

ϕ_i^{j+1}
ϕ_{i+1}^{j+1}
$\nabla_h^{i+1/2,j+1/2}\phi$
ϕ_i^j
ϕ_{i+1}^j
$y^{j+1/2}$
$y^{j-1/2}$
$x_{i-1/2}$
$x_{i+1/2}$
$\mathbf{b}_{i-1/2}^{j+1/2}$
$\mathbf{b}_{i+1/2}^{j+1/2}$
$\nabla_h^{i,j} \times \mathbf{b}$
$\mathbf{b}_{i-1/2}^{j-1/2}$
$\mathbf{b}_{i+1/2}^{j-1/2}$
$y^{j+1/2}$
$y^{j-1/2}$
$x_{i-1/2}$
$x_{i+1/2}$

Figure 5.2. Staggered mesh configuration with a scalar field field φ_i^j defined in the cell barycenters, and the interface field field $\mathbf{b}_{i+1/2}^{j+1/2}$ defined on the vertices of the main grid. The shaded control volumes indicate the stencil for the computation of the corner gradients $\nabla_h^{i+1/2,j+1/2}\varphi$ and for the cell-centred curl operator $\nabla_h^{i,j} \times \mathbf{b}$.

and its degrees of freedom are naturally defined as

$$\begin{gathered}\left(\nabla_h^{i,j} \times \mathbf{b}^n\right) \cdot \hat{\mathbf{e}}_z = \Big[(\nabla_h \times \mathbf{b}^n) \cdot \hat{\mathbf{e}}_z \Big]_{i,j} = \\ \frac{(b_2^n)_{i+1/2}^{j+1/2} - (b_2^n)_{i-1/2}^{j+1/2} + (b_2^n)_{i+1/2}^{j-1/2} - (b_2^n)_{i-1/2}^{j-1/2}}{2\,\Delta x} + \\ \frac{(b_1)_{i+1/2}^{j+1/2} - (b_1^n)_{i+1/2}^{j-1/2} + (b_1^n)_{i-1/2}^{j+1/2} - (b_1^n)_{i-1/2}^{j-1/2}}{2\,\Delta y}\end{gathered} \tag{5.31}$$

making use of the *vertex based* staggered values of the field $\mathbf{b}^n$, see Fig. 5.2. In the present two-dimensional description of the scheme the first and second components of the discrete curl $\nabla_h^{i,j} \times \mathbf{b}^n$ vanish identically. Eqn. (5.31) defines a discrete curl on the control volume Ω_{ij} via the Stokes theorem

$$(\nabla_h \times \mathbf{b}^n) \cdot \hat{\mathbf{e}}_z = \frac{1}{\Delta x\,\Delta y} \int\limits_{\Omega_{ij}} (\nabla \times \mathbf{b}) \cdot \hat{\mathbf{e}}_z \, \mathrm{d}\mathbf{x} = \frac{1}{\Delta x\,\Delta y} \int\limits_{\partial\Omega_{ij}} \mathbf{b} \cdot \hat{\mathbf{t}} \, \mathrm{d}S, \tag{5.32}$$

based on the trapezoidal rule for the computation of the integrals along each edge of Ω_{ij}.

Last but not least, we need to define a discrete gradient operator that is compatible with the discrete curl, so that the continuous identity

$$\nabla \times \nabla\varphi = \mathbf{0} \tag{5.33}$$

also holds on the discrete level. If we define a scalar field in the barycenters of the control volumes Ω_{ij} as $\varphi_{ij}^n = \varphi(t^n,\ x_i,\ y_j)$ then the corner gradient generates a natural discrete gradient operator ∇_h of the discrete scalar field φ^n that defines a discrete gradient in all vertices of the mesh. The corresponding degrees of freedom generated by $\nabla_h\varphi^n$ read (see Fig. 5.2)

$$\nabla_h^{i+1/2,j+1/2}\varphi^n = (\nabla_h\varphi^n)_{i+1/2}^{j+1/2} = \begin{pmatrix} \dfrac{\varphi_{i+1,j+1}^n - \varphi_{i,j+1}^n + \varphi_{i+1,j}^n - \varphi_{i,j}^n}{2\,\Delta x} \\[2ex] \dfrac{\varphi_{i+1,j+1}^n - \varphi_{i+1,j}^n + \varphi_{i,j+1}^n - \varphi_{i,j}^n}{2\,\Delta y} \\[2ex] 0 \end{pmatrix}. \tag{5.34}$$

It is then straightforward to verify that an immediate consequence of Equations (5.31) and (5.34) is

$$\nabla_h \times \nabla_h\varphi^n = \mathbf{0}, \tag{5.35}$$

i.e. one obtains a discrete analogue of (5.33). This can be easily seen by computing

$$
\begin{aligned}
&\left(\nabla_h^{i,j} \times \nabla^{i+1/2,j+1/2} \varphi^n\right) \cdot \hat{\mathbf{e}}_z = \\
&\frac{\left(\varphi_{i+1}^{j+1} - \varphi_{i+1}^{j} + \varphi_{i}^{j+1} - \varphi_{i}^{j}\right) + \left(\varphi_{i+1}^{j} - \varphi_{i+1}^{j-1} + \varphi_{i}^{j} - \varphi_{i}^{j-1}\right)}{4\,\Delta x\,\Delta y} - \\
&\frac{\left(\varphi_{i}^{j+1} - \varphi_{i}^{j} + \varphi_{i-1}^{j+1} - \varphi_{i-1}^{j}\right) + \left(\varphi_{i}^{j} - \varphi_{i}^{j-1} + \varphi_{i-1}^{j} - \varphi_{i-1}^{j-1}\right)}{4\,\Delta x\,\Delta y} - \\
&\frac{\left(\varphi_{i+1}^{j+1} - \varphi_{i}^{j+1} + \varphi_{i+1}^{j} - \varphi_{i}^{j}\right) + \left(\varphi_{i}^{j+1} - \varphi_{i-1}^{j+1} + \varphi_{i}^{j} - \varphi_{i-1}^{j}\right)}{4\,\Delta y\,\Delta x} + \\
&\frac{\left(\varphi_{i+1}^{j} - \varphi_{i}^{j} + \varphi_{i+1}^{j-1} - \varphi_{i}^{j-1}\right) + \left(\varphi_{i}^{j} - \varphi_{i-1}^{j} + \varphi_{i}^{j-1} - \varphi_{i-1}^{j-1}\right)}{4\,\Delta y\,\Delta x} = 0.
\end{aligned}
\tag{5.36}
$$

We furthermore define the following averaging operators from the edge-based staggered meshes to the cell barycenter $(x_i,\ y_j)$ and viceversa

$$
\begin{aligned}
(u_1^n)_{i,j} =& \frac{1}{2}\left[(u_1^n)_{i-1/2,j} + (u_1^n)_{i+1/2,j}\right], \\
(u_2^n)_{i,j} =& \frac{1}{2}\left[(u_2^n)_{i,j-1/2} + (u_2^n)_{i,j+1/2}\right], \\
(u_1^n)_{i+1/2,j} =& \frac{1}{2}\left[(u_1^n)_{i,j} + (u_1^n)_{i+1,j}\right], \\
(u_2^n)_{i,j+1/2} =& \frac{1}{2}\left[(u_2^n)_{i,j} + (u_2^n)_{i,j+1}\right].
\end{aligned}
\tag{5.37}
$$

Interpolation from the cell centers to the the cell-edge locations is required for the construction of a compact stencil symmetric positive definite system for the pressure wave equation, while interpolation to the cell centers can be avoided by using face-interpolated pressure values for a direct update of the momentum on the barycenter grid.

Finally we also introduce an interpolation operator to compute cell center approximations of the corner quantities like $\mathbf{b}$ and $\mathbf{a}_{sk}$ which we write

$$
\mathbf{b}_{i,j}^n = \frac{1}{4}\left(\mathbf{b}_{i-1/2,j-1/2}^n + \mathbf{b}_{i+1/2,j-1/2}^n + \mathbf{b}_{i-1/2,j+1/2}^n + \mathbf{b}_{i-1/2,j+1/2}^n\right) \tag{5.38}
$$

which represents a simple (linear) arithmetic averaging operator and introduces minimal numerical dissipation. Whenever flow convection is particularly strong, we found beneficial to apply a partial upwinding to the interpolation operator for the interface field $\mathbf{b}$, so to add additional numerical stabilisiation to the scheme.

$$\mathbf{b}_{i,j}^{n} = w_1 \left(\mathbf{b}^n\right)_{i-1/2}^{j-1/2} + w_2 \left(\mathbf{b}^n\right)_{i+1/2}^{j-1/2} + w_3 \left(\mathbf{b}^n\right)_{i-1/2}^{j+1/2} + w_4 \left(\mathbf{b}^n\right)_{i-1/2}^{j+1/2}. \tag{5.39}$$

The coefficients w_k are obtained by first constructing a set of preliminary weighs by two-dimensional upwinding.

$$\begin{aligned} w_1^* &= \varepsilon + u_1^+ + u_2^+, \qquad & w_2^* &= \varepsilon + u_1^- + u_2^+, \\ w_3^* &= \varepsilon + u_1^+ + u_2^-, \qquad & w_4^* &= \varepsilon + u_1^- + u_2^-, \end{aligned} \tag{5.40}$$

with $\varepsilon = 10^{-6}$, $\mathbf{u}^+ = (u_1^+,\ u_2^+)^\intercal = \max\left(0,\ \mathbf{u}\right)/(\|\mathbf{u}\|+\varepsilon)$, and $\mathbf{u}^- = (u_1^-,\ u_2^-)^\intercal = \max\left(0,\ -\mathbf{u}\right)/(\|\mathbf{u}\|+\varepsilon)$. Then the preliminary weights (5.40) are normalized in such a way that the upwind bias will be reduced for flows with weak convection. The final weights thus are computed as

$$w_k = \frac{w_k^*}{\sum_{k=1}^{4} w_k^*}\,\lambda + \frac{1-\lambda}{4}, \ \text{with } \lambda = \min\left(1,\ 2\ \|\mathbf{u}\|\ \sqrt{\Delta x\,\Delta y}\right). \tag{5.41}$$

5.4 Explicit discretization of involution constrained fields

The key ingredient of the numerical method proposed in this Chapter is the curl-compatible discretization of the terms $\nabla \mathbf{G}_v(\mathbf{Q})$ and $\mathbf{B}_v(\mathbf{Q})\nabla\mathbf{Q}$ present in (5.3). We propose the following compatible discretization for the interface field equation:

$$\begin{aligned} \mathbf{b}_{i+1/2,j+1/2}^{n+1} = \mathbf{b}_{i+1/2,j+1/2}^{n} &- \Delta t\,\nabla_h^{i+1/2,j+1/2}\left(\varphi^n\right) + \\ &- \Delta t\,\Big[\left(\nabla_h \times \mathbf{b}\right) \times \mathbf{u}\Big]_{i+1/2,j+1/2} \end{aligned} \tag{5.42}$$

with the corner-averaged curl term is given by

$$\Big[\left(\nabla_h \times \mathbf{b}\right) \times \mathbf{u}\Big]_{i+1/2,j+1/2} = \frac{1}{4}\sum_{r=0}^{1}\sum_{s=0}^{1}\left(\nabla_h^{i+r,j+s} \times \mathbf{b}^n\right) \times \mathbf{u}_{i+r,j+s}^{n} \tag{5.43}$$

It is easy to check that for an initially curl-free vector field $\mathbf{b}^n$ which satisfies $\nabla_h \times \mathbf{b}^n = 0$ also $\nabla_h \times \mathbf{b}^{n+1} = 0$ holds. In order to see this, one needs to apply the discrete curl operator $\nabla_h \times$ to Eqn. (5.42). One realizes that the second row of (5.42), which contains the discrete curl of $\mathbf{b}^n$ vanishes immediately, due to $\nabla_h \times \mathbf{b}^n = 0$. The curl of the first term on the right hand side in the first row of Eqn. (5.42) is zero because of $\nabla_h \times \mathbf{b}^n = 0$ and the curl of the second term is zero because of $\nabla_h \times (\nabla_h \varphi^n) = 0$, with the auxiliary scalar field $\varphi^n = \mathbf{b}^n \cdot \mathbf{u}^n$, whose degrees of freedom are computed as $\varphi^n_{i,j} = \mathbf{b}^n_{i,j} \cdot \mathbf{u}^n_{i,j}$ after interpolating the velocity vector and the gradient field $\mathbf{b}$ into the barycenters of the control volumes $\Omega_{i,j}$. The key ingredient of our compatible discretization for the $\mathbf{b}$ equation is indeed the use of a discrete gradient operator that is compatible with the discrete curl operator, see Eq. (5.36).

5.5 Compatible numerical viscosity

The previous discretizations were all *central* and thus without artificial numerical viscosity. In order to add a *compatible numerical viscosity* operator, we need to recall the definition of the vector Laplacian at the continuous level, which reads:

$$\nabla^2 \mathbf{b} = \nabla \left(\nabla \cdot \mathbf{b} \right) - \nabla \times \left(\nabla \times \mathbf{b} \right) \tag{5.44}$$

In order to define a discrete analogue of (5.44) we define another discrete divergence operator, which naturally follows from the definition of the discrete gradient (5.34)

$$\begin{aligned} \nabla_h^{i+1/2,j+1/2} \cdot \mathbf{b}^n = {} & \frac{(b_1^n)^{j+1}_{i+1} - (b_1^n)^{j+1}_{i} + (b_1^n)^{j}_{i+1} - (b_1^n)^{j}_{i}}{2\,\Delta x} + \\ & + \frac{(b_2)^{j+1}_{i+1} - (b_2^n)^{j}_{i+1} + (b_2^n)^{j+1}_{i} - (b_2^n)^{j}_{i}}{2\,\Delta y}. \end{aligned} \tag{5.45}$$

and yields the degrees of freedom of the divergence of $\mathbf{b}^n$ at the cell corner locations, starting from the cell center interpolated values of the vector field $\mathbf{b}^n$. By shifting indices by a half step in both directions, the same operator can be used to obtain cell center values for $\nabla_h \cdot \mathbf{b}^n$ starting from the corner values of $\mathbf{b}^n$. In this case, the operator is

$$\begin{aligned} \nabla_h^{i,j} \cdot \mathbf{b}^n = {} & \frac{(b_1^n)^{j+1/2}_{i+1/2} - (b_1^n)^{j+1/2}_{i-1/2} + (b_1^n)^{j-1/2}_{i+1/2} - (b_1^n)^{j-1/2}_{i-1/2}}{2\,\Delta x} + \\ & + \frac{(b_2)^{j+1/2}_{i+1/2} - (b_2^n)^{j-1/2}_{i+1/2} + (b_2^n)^{j+1/2}_{i-1/2} - (b_2^n)^{j-1/2}_{i-1/2}}{2\,\Delta y}. \end{aligned} \tag{5.46}$$

The discrete vector Laplacian then simply reads

$$\begin{aligned} (\nabla_h^2)^{i+1/2,j+1/2} \mathbf{b}^n = {} & \nabla_h^{i+1/2,j+1/2} \cdot \left(\nabla_h^{i,j} \mathbf{b}^n \right) = \\ & = \nabla_h^{i+1/2,j+1/2} \left(\nabla_h^{i,j} \cdot \mathbf{b}^n \right) - \nabla_h^{i+1/2,j+1/2} \times \left(\nabla_h^{i,j} \times \mathbf{b}^n \right), \end{aligned} \tag{5.47}$$

i.e. it is composed of a grad-div contribution minus a curl-curl term. Taking (5.47) into account, a compatible discretization of $\mathbf{b}$ *with* numerical viscosity then reads

$$\begin{aligned}\mathbf{b}^{n+1}_{i+1/2,j+1/2} = \mathbf{b}^{n}_{i+1/2,j+1/2} &- \Delta t\, \nabla_h^{i+1/2,j+1/2} \left(\varphi + h\, c_a\, \nabla_h^{i,j} \cdot \mathbf{b}^n\right) + \\ &- \Delta t \left[(\nabla_h \times \mathbf{b}) \times \mathbf{u} \right]_{i+1/2,j+1/2} + \\ &- \Delta t\, h\, c_a\, \nabla_h^{i+1/2,j+1/2} \times \left(\nabla_h^{i,j} \times \mathbf{b}^n\right)\end{aligned} \tag{5.48}$$

where $h = \max(\Delta x, \Delta y)$, is a characteristic mesh spacing and c_a is a characteristic velocity related to the artificial viscosity that one would like to add to the scheme. In practice we take $c_a = k_L\, \lambda$, with $\lambda = \max_\Omega \left(\|\mathbf{u}\|\right)$ for the evolution of the distortion field $\mathbf{A}$ and $\lambda = \max_\Omega \left(\|\mathbf{u}\| + \sigma \|\mathbf{b}\| / \rho\right)$ for the evolution of the interface field $\mathbf{b}$. Unless otherwise specified, we take $k_L = 0.1$. For the sake of clarity, the additional numerical viscosity terms have been highlighted in red. It is obvious that also (5.48) satisfies the curl-free property $\nabla_h \times \mathbf{b}^{n+1} = \mathbf{0}$ if $\nabla_h \times \mathbf{b}^n = \mathbf{0}$. In order to reduce the numerical dissipation, it is possible to employ a piecewise linear reconstruction and insert the barycenter extrapolated values into the discrete divergence operator under the discrete gradient. In two space dimensions, the curl-curl term in (5.48) simplifies to

$$\nabla_h^{i+1/2,j+1/2} \times \left(\nabla_h^{i,j} \times \mathbf{b}^n\right) = \begin{pmatrix} -\dfrac{\omega^n_{i+1,j+1} - \omega^n_{i+1,j} + \omega^n_{i,j+1} - \omega^n_{i,j}}{2\,\Delta y} \\[2ex] \dfrac{\omega^n_{i+1,j+1} - \omega^n_{i,j+1} + \omega^n_{i+1,j} - \omega^n_{i,j}}{2\,\Delta x} \\[2ex] 0 \end{pmatrix}. \tag{5.49}$$

by denoting with $\omega^n_{i,j} = \left(\nabla_h^{i,j} \times \mathbf{b}^n\right) \cdot \hat{\mathbf{e}}_z$ the third component of the discrete curl of $\mathbf{b}^n$.

5.6 Implicit solution of the pressure equation

Up to now, the contribution of the pressure to the momentum and to the total energy conservation laws has been excluded, i.e. the terms contained in the pressure fluxes $\mathbf{F}_p$. The discrete momentum equations including the pressure terms read

$$\begin{aligned}(\rho\, u_1^{n+1})^j_{i+1/2} &= (\rho\, u_1^*)^j_{i+1/2} + \Delta t\, (f_1^*)^j_{i+1/2} - \frac{\Delta t}{\Delta x} \left(p^{n+1}_{i+1,j} - p^{n+1}_{i,j}\right) \\ (\rho\, u_2^{n+1})^{j+1/2}_i &= (\rho\, u_2^*)^{j+1/2}_i + \Delta t\, (f_2^*)^{j+1/2}_i - \frac{\Delta t}{\Delta y} \left(p^{n+1}_{i,j+1} - p^{n+1}_{i,j}\right)\end{aligned} \tag{5.50}$$

where pressure is taken *implicitly*, while nonlinear convective terms have already been discretized *explicitly* via the operators $(\rho\, u_1^{*})_{i+1/2}^{j}$ and $(\rho\, u_2^{*})_{i}^{j+1/2}$ given in (5.19) and after averaging of the obtained quantities back to the edge-based staggered dual grid. The contribution to momentum due to the gravity source and the vertex fluxes (due to capillarity and viscosity), is computed, using the four point discrete divergence of the fluxes (5.46), as

$$\begin{aligned}(\mathbf{f}^{*})_i^j = {}& \frac{(\boldsymbol{\Omega}_{1k}^{n})_{i+1/2}^{j+1/2} - (\boldsymbol{\Omega}_{1k}^{n})_{i-1/2}^{j+1/2} + (\boldsymbol{\Omega}_{1k}^{n})_{i+1/2}^{j-1/2} - (\boldsymbol{\Omega}_{1k}^{n})_{i-1/2}^{j-1/2}}{2\,\Delta x} + \\ & + \frac{(\boldsymbol{\Omega}_{2k})_{i+1/2}^{j+1/2} - (\boldsymbol{\Omega}_{2k}^{n})_{i+1/2}^{j-1/2} + (\boldsymbol{\Omega}_{2k}^{n})_{i-1/2}^{j+1/2} - (\boldsymbol{\Omega}_{2k}^{n})_{i-1/2}^{j-1/2}}{2\,\Delta y} + \rho_{i,j}^{n+1}\,\mathbf{g}.\end{aligned} \tag{5.51}$$

where $\boldsymbol{\Omega}_{1k}$ and $\boldsymbol{\Omega}_{2k}$ indicate the first and the second row of the tensor $\boldsymbol{\Omega} = -\boldsymbol{\Sigma}_{\mathrm{t}}(\mathbf{b}^{n+1}) - \boldsymbol{\Sigma}_{\mathrm{s}}(\mathbf{A}^{n+1},\ \rho^{n+1})$ collecting the effects of the stress tensors associated with corner quantities $\mathbf{b}$ and $\mathbf{A}$, i.e. capillarity and viscous forces respectively. Both components of the flux divergence are then interpolated onto the corresponding cell edges and yielding

$$(f_1^{*})_{i+1/2}^{j} = \frac{(\mathbf{f}^{*})_i^j + (\mathbf{f}^{*})_{i+1}^j}{2}\cdot\hat{\mathbf{e}}_1, \quad (f_2^{*})_i^{j+1/2} = \frac{(\mathbf{f}^{*})_i^j + (\mathbf{f}^{*})_i^{j+1}}{2}\cdot\hat{\mathbf{e}}_2 \tag{5.52}$$

The first component of $\mathbf{f}^{*}$, $(f_1^{*})_{i+1/2}^{j}$, will contribute to the momentum balance in the x-direction, and for this reason it is interpolated only at the u_1-velocity locations, while, the second component $(f_2^{*})_i^{j+1/2}$ is part of the momentum balance in the y-direction and is interpolated at the u_2-velocity locations.

A preliminary form of the discrete total energy equation reads

$$\begin{aligned}&\rho\, e(p_{i,j}^{n+1}) + (\rho\, e_{\mathrm{t}}^{n+1})_{i,j} + (\rho\, e_{\mathrm{s}}^{n+1})_{i,j} + (\rho\, \tilde{e}_{\mathrm{k}}^{n+1})_{i,j} = \rho\, E_{i,j}^{*} + \\ &\quad -\frac{\Delta t}{\Delta x}\left[\tilde{h}_{i+1/2,j}^{n+1}\,(\rho\, u_1^{n+1})_{i+1/2}^{j} - \tilde{h}_{i-1/2,j}^{n+1}\,(\rho\, u_1^{n+1})_{i-1/2}^{j}\right] + \\ &\quad -\frac{\Delta t}{\Delta y}\left[\tilde{h}_{i,j+1/2}^{n+1}\,(\rho\, u_2^{n+1})_{i}^{j+1/2} - \tilde{h}_{i,j-1/2}^{n+1}\,(\rho\, u_2^{n+1})_{i}^{j-1/2}\right] + (\rho\, \tilde{w}_{\mathrm{g}}^{n+1})_{i,j}\end{aligned} \tag{5.53}$$

with the term $(\rho\, \tilde{w}_{\mathrm{g}}^{n+1})_{i,j} = \rho\, \mathbf{u}_{i,j}^{n+1}\cdot\mathbf{g}$ accounting for the work due to gravity forces.

Inserting the discrete momentum equations (5.50) into the discrete energy equation (5.53) and making tilde symbols explicit via a simple Picard iteration (using the lower index r in the following), as suggested in [104, 109], leads to the following discrete wave equation for the unknown pressure:

$$\begin{aligned}\rho\, e\left(p_{i,j}^{n+1}\right) - \frac{\Delta t^2}{\Delta x^2}\left(\tilde{h}_{i+1/2,j}^{n+1}\right)_r \left(p_{i+1,j}^{n+1} - p_{i,j}^{n+1}\right)_{r+1} &+ \\ + \frac{\Delta t^2}{\Delta x^2}\left(\tilde{h}_{i-1/2,j}^{n+1}\right)_r \left(p_{i,j}^{n+1} - p_{i-1,j}^{n+1}\right)_{r+1} &+ \\ - \frac{\Delta t^2}{\Delta y^2}\left(\tilde{h}_{i,j+1/2}^{n+1}\right)_r \left(p_{i,j+1}^{n+1} - p_{i,j}^{n+1}\right)_{r+1} &+ \\ + \frac{\Delta t^2}{\Delta y^2}\left(\tilde{h}_{i,j-1/2}^{n+1}\right)_r \left(p_{i,j}^{n+1} - p_{i,j-1}^{n+1}\right)_{r+1} &= (d_{i,j})_r \end{aligned} \tag{5.54}$$

with the known right hand side

$$\begin{aligned}(d_{i,j})_r = \rho\, E_{i,j}^* - (\rho\, e_{\mathrm{t}}^{n+1})_{i,j} - (\rho\, e_{\mathrm{s}}^{n+1})_{i,j} - \left[\left(\rho\, \tilde{e}_{\mathrm{k}}^{n+1}\right)_{i,j}\right]_r + \left[\left(\rho\, \tilde{w}_{\mathrm{g}}^{n+1}\right)_{i,j}\right]_r &+ \\ - \frac{\Delta t}{\Delta x}\left(\tilde{h}_{i+1/2,j}^{n+1}\right)_r \left[(\rho\, u_1^*)_{i+1/2}^j + \Delta t\ (f_1^*)_{i+1/2}^j\right] &+ \\ + \frac{\Delta t}{\Delta x}\left(\tilde{h}_{i-1/2,j}^{n+1}\right)_r \left[(\rho\, u_2^*)_{i-1/2}^j + \Delta t\ (f_1^*)_{i-1/2}^j\right] &+ \\ - \frac{\Delta t}{\Delta y}\left(\tilde{h}_{i,j+1/2}^{n+1}\right)_r \left[(\rho\, u_2^*)_i^{j+1/2} + \Delta t\ (f_2^*)_i^{j+1/2}\right] &+ \\ + \frac{\Delta t}{\Delta y}\left(\tilde{h}_{i,j-1/2}^{n+1}\right)_r \left[(\rho\, u_2^*)_i^{j-1/2} + \Delta t\ (f_2^*)_i^{j-1/2}\right] & \end{aligned} \tag{5.55}$$

The density at the new time $\rho_{i,j}^{n+1} = \rho_{i,j}^*$ is already known from (5.19), and so are the energy contribution $(\rho\, e_{\mathrm{s}}^{n+1})_{i,j}$ of the distortion field $\mathbf{A}^{n+1}$ and the interface energy $(\rho\, e_{\mathrm{t}}^{n+1})_{i,j}$ of the field $\mathbf{b}^{n+1}$, after averaging onto the main grid the staggered field components of $\mathbf{b}$ and $\mathbf{A}$ that have been evolved in the vertices via the compatible discretization (5.42). Concerning the kinetic energy contribution, it is updated explicitly via a fixed-point iteration like the enthalpy $\tilde{h}^{n+1}$

$$\begin{aligned}\left[\left(\rho\, \tilde{e}_{\mathrm{k}}^{n+1}\right)_{i,j}\right]_r = \frac{1}{2}\,\rho_{i,j}^{n+1}\left\{\frac{\left[\left(u_1^{n+1}\right)_{i-1/2}^j\right]_r + \left[\left(u_1^{n+1}\right)_{i+1/2}^j\right]_r}{2}\right\}^2 &+ \\ + \frac{1}{2}\,\rho_{i,j}^{n+1}\left\{\frac{\left[\left(u_2^{n+1}\right)_i^{j-1/2}\right]_r + \left[\left(u_2^{n+1}\right)_i^{j+1/2}\right]_r}{2}\right\}^2 &, \end{aligned} \tag{5.56}$$

and the same update strategy is applied for the work due to gravity forces

$$\begin{aligned}\left[(\rho\,\tilde{w}_{\mathrm{g}}^{n+1})_{i,j}\right]_r = \frac{1}{2}\,&\left[\left(\rho\,u_1^{n+1}\right)_{i-1/2}^{j} + \left(\rho\,u_1^{n+1}\right)_{i+1/2}^{j}\right]_r\,\mathbf{g}\cdot\hat{\mathbf{e}}_x + \\ +\frac{1}{2}\,&\left[\left(\rho\,u_2^{n+1}\right)_{i}^{j-1/2} + \left(\rho\,u_2^{n+1}\right)_{i}^{j+1/2}\right]_r\,\mathbf{g}\cdot\hat{\mathbf{e}}_y\end{aligned} \tag{5.57}$$

For a general equation of state, the final system for the pressure (5.54) forms a *mildly nonlinear system* (see [109]) of the form

$$\rho\,\mathbf{e}\left(\mathbf{p}_{r+1}^{n+1}\right) + \mathbf{M}_r\,\mathbf{p}_{r+1}^{n+1} = \mathbf{d}_r^n \tag{5.58}$$

with a linear part contained in $\mathbf{M}$ that is symmetric and at least positive semidefinite. Hence, with the usual assumptions on the nonlinearity detailed in [78], it can be efficiently solved with the nested Newton method of Casulli and Zanolli [77, 78]. For our particular choice of equation of state (stiffened gas), the system is *linear* and thus we can employ an even simpler Jacobi-preconditioned matrix free conjugate gradient method for its solution.

Note that in the incompressible limit $\mathbb{M}\mathrm{a} \to 0$, following the asymptotic analysis performed in [205–207, 238, 241], the pressure tends to a constant and the contribution of the kinetic energy $\rho\,\tilde{e}_{\mathrm{k}}$ can be neglected with respect to the internal energy $\rho\,e$. Therefore, in the incompressible limit the system (5.54) tends to the classic pressure Poisson equation used in incompressible flow solvers. In each Picard iteration, after the solution of the pressure system (5.54), the enthalpies at the interfaces are recomputed and the momentum is updated by

$$\begin{aligned}\left[\left(\rho\,u_1^{n+1}\right)_{i+1/2}^{j}\right]_{r+1} &= (\rho\,u_1^{*})_{i+1/2}^{j} + \Delta t\left(\rho_{i+1/2,j}^{n+1}\,\mathbf{g}\cdot\hat{\mathbf{e}}_x - \frac{p_{i+1,j}^{n+1} - p_{i,j}^{n+1}}{\Delta x}\right)_{r+1}, \\ \left[\left(\rho\,u_2^{n+1}\right)_{i}^{j+1/2}\right]_{r+1} &= (\rho\,u_2^{*})_{i}^{j+1/2} + \Delta t\left(\rho_{i,j+1/2}^{n+1}\,\mathbf{g}\cdot\hat{\mathbf{e}}_y - \frac{p_{i,j+1}^{n+1} - p_{i,j}^{n+1}}{\Delta y}\right)_{r+1},\end{aligned} \tag{5.59}$$

At the end of the Picard iterations, the total energy is updated as

$$\begin{aligned}\rho\,E_{i,j}^{n+1} = \rho\,E_{i,j}^{*} &- \frac{\Delta t}{\Delta x}\left[\tilde{h}_{i+1/2,j}^{n+1}\,(\rho\,u_1^{n+1})_{i+1/2}^{j} - \tilde{h}_{i-1/2,j}^{n+1}\,(\rho\,u_1^{n+1})_{i-1/2}^{j}\right] + \\ &- \frac{\Delta t}{\Delta y}\left[\tilde{h}_{i,j+1/2}^{n+1}\,(\rho\,u_2^{n+1})_{i}^{j+1/2} - \tilde{h}_{i,j-1/2}^{n+1}\,(\rho\,u_2^{n+1})_{i}^{j-1/2}\right] + \\ &+ (\rho\,\tilde{w}_{\mathrm{g}}^{n+1})_{i,j}.\end{aligned} \tag{5.60}$$

While for the final main-grid update of the momentum variables we compute a set of interpolated cell-face values for the pressure field $p_{i+1/2,j}^{n+1} = (p_{i,j}^{n+1} + p_{i+1,j}^{n+1})/2$ and $p_{i,j+1/2}^{n+1} = (p_{i,j}^{n+1} + p_{i,j+1}^{n+1})/2$, and then update the cell-center momentum *directly* with

$$\begin{aligned} \rho\,\mathbf{u}_{i,j}^{n+1} = \rho\,\mathbf{u}_{i,j}^{*} - \frac{\Delta t}{\Delta x}\,\left(p_{i+1/2,j}^{n+1} - p_{i-1/2,j}^{n+1}\right) + \\ - \frac{\Delta t}{\Delta y}\,\left(p_{i,j+1/2}^{n+1} - p_{i,j-1/2}^{n+1}\right) + \Delta t\,\rho_{i,j}^{n+1}\,\mathbf{g}. \end{aligned} \tag{5.61}$$

This approach is chosen as opposed to averaging the momentum from the cell-face grid to the cell center grid, so to avoid the Lax–Friedrichs-type numerical diffusion associated with incurred in the final momentum is averaged back onto the main grid.

5.7 Proof of the Abgrall condition

We provide here a simple proof that the proposed scheme respects the so-called Abgrall condition [2], that is, it preserves exactly those flows characterised by a constant velocity and constant pressure. In absence of other driving forces, such uniform flows should not be affected by spurious perturbations, regardless of the distribution of density or volume fraction, which do not affect the dynamics in these situations.

The starting point is showing that the velocity field produced by the convective step is kept uniform by the MUSCL–Hancock scheme applied to the convective subsystem. In one space dimension, the mixture density ρ obey the update formula

$$\rho_i^{n+1} = \rho_i^n - \frac{\Delta t}{\Delta x}\,\left(f_{i+1/2}^{\rho} - f_{i-1/2}^{\rho}\right), \tag{5.62}$$

with the Rusanov flux yielding explicitly

$$f_{i+1/2}^{\rho}(\rho_L,\ \rho_R) = \frac{1}{2}\,u_1\,\left(\rho_L + \rho_R\right) - \frac{1}{2}\,|u_1|\,\left(\rho_R - \rho_L\right) \tag{5.63}$$

which is easily shown by direct sum of the equations for the phase densities $\alpha_1\,\rho_1$ and $\alpha_2\,\rho_2$. Since it is a fundamental assumption that the velocity field is constant at time level t^n, in this proof we can denote $u_1 = (u_1)_i = (\rho\,u_1)_i^n/\rho_i^n$ for any cell i, The update formula for the mixture momentum $\rho\,u_1$ similarly reads

$$(\rho\,u_1)_i^{n+1} = (\rho\,u_1)_i^n - \frac{\Delta t}{\Delta x}\,\left(f_{i+1/2}^{\rho\,u_1} - f_{i-1/2}^{\rho\,u_1}\right), \tag{5.64}$$

and exploiting the constant velocity assumption, the flux is

$$f^{\rho\, u_1}_{i+1/2} = \frac{1}{2}\, u_1\, u_1\, (\rho_L + \rho_R) - \frac{1}{2}\, |u_1|\, u_1\, (\rho_R - \rho_L) = u_1\, f^{\rho}_{i+1/2}, \tag{5.65}$$

which means that (5.64) can be simplified to

$$(\rho\, u_1)^*_i = (\rho\, u_1)^n_i - u_1\, \frac{\Delta t}{\Delta x}\, \left(f^{\rho}_{i+1/2} - f^{\rho}_{i-1/2}\right), \tag{5.66}$$

from which, setting $(u_1)^*_i = (u_1)^n_i = u_1$ allows to factor out Eq. (5.62), showing that the constant velocity field is preserved by the scheme. The same formal proportionality argument can be applied also to the cell-local predictor of the MUSCL–Hancock scheme, showing that the velocity field generated by the predictor step is unaltered in the same way.

It remains to be shown that a constant pressure field $p = p^n_i = p^{n+1}_i$ for any index i is a solution of the discrete wave equation resulting from the energy balance

$$\begin{aligned} \rho\, e^{n+1}_i + (\rho\, e_{\mathrm{k}})^{n+1}_i = {} & \rho\, e^n_i + (\rho\, e_{\mathrm{k}})^*_i + \\ & - \frac{\Delta t}{\Delta x}\, \left[\tilde{h}^{n+1}_{i+1/2}\, (\rho\, u_1)^*_{i+1/2} - \tilde{h}^{n+1}_{i-1/2}\, (\rho\, u_1)^*_{i-1/2}\right], \end{aligned} \tag{5.67}$$

together with the equivalences $(\rho\, e_{\mathrm{k}})^{n+1}_i = (\rho\, e_{\mathrm{k}})^*_i$ and $(\rho\, u_1)^{n+1}_{i+1/2} = (\rho\, u_1)^*_{i+1/2}$ resulting from the constant-pressure assumption which means that momentum $\rho\, u_1$ and kinetic energy $\rho\, e_{\mathrm{k}}$ at the new time level coincide with those resulting from the convective subsystem. Collecting the constant velocity u_1 and plugging in a generic linear equation of state of the form $\rho\, e = k_0 + k_1\, p$, which is the form of the stiffened gas EOS applied to our model when α_1 is constant throughout the domain, gives

$$\begin{aligned} k_0 + k_1\, p^{n+1}_i + (\rho\, e_{\mathrm{k}})^{n+1}_i = {} & k_0 + k_1\, p^n_i + (\rho\, e_{\mathrm{k}})^*_i + \\ & - \frac{\Delta t}{\Delta x}\, u_1\, \left(\rho^*_{i+1/2}\, \tilde{h}^{n+1}_{i+1/2} - \rho^*_{i-1/2}\, \tilde{h}^{n+1}_{i-1/2}\right), \end{aligned} \tag{5.68}$$

which with the constant pressure assumption $p^{n+1}_i = p_i = p$ yields a condition

$$\rho^*_{i+1/2}\, \tilde{h}^{n+1}_{i+1/2} - \rho^*_{i-1/2}\, \tilde{h}^{n+1}_{i-1/2} = 0, \tag{5.69}$$

highlighting that the enthalpy estimates $\tilde{h}^{n+1}_{i+1/2}$ must be chosen as

$$\tilde{h}^{n+1}_{i+1/2} = \frac{\rho\, \tilde{h}^{n+1}_{i+1/2}}{\rho^*_{i+1/2}} = \frac{\rho\, e(p_{i+1/2}) + p_{i+1/2}}{\rho^*_{i+1/2}}, \tag{5.70}$$

meaning that the density used for the computation of enthalpies must necessarily be the one produced by the convective step $\rho^*_{i+1/2}$. Any interpolation scheme for computing $p_{i+1/2}$ will clearly work in a constant pressure field, and we use a simple arithmetic average $p_{i+1/2} = (p_i + p_{i+1})/2$, and the same average is employed for computing $\rho^*_{i+1/2}$, but in this case it is important that the interpolation operator be the same applied for the computation of the momentum $(\rho\, u_1)^*_{i+1/2}$ from the cell-center quantities. Also, note that in order to be able to set $(\rho\, e_\mathrm{k})^{n+1}_i = (\rho\, e_\mathrm{k})^*_i$, the kinetic energy computed from averaging the (constant) velocities from the cell center to the edges and vice-versa, must be compatible with that obtained by the MUSCL–Hancock scheme itself, which is verified thanks again to the fact that a compatible numerical dissipation was chosen for density, momentum, and kinetic energy. Specifically, it is immediately apparent that, analogously to the momentum flux, we can write the kinetic energy flux as $f^{\rho\, e_\mathrm{k}}_{i+1/2} = e_\mathrm{k}\, f^{\rho}_{i+1/2}$, thus the specific kinetic energy $e_\mathrm{k} = u_1^2/2$ is kept constant after the convective step.

Given the conditions above, one can immediately see that a constant pressure field, with $\rho\, e^{n+1}_\mathrm{k} = \rho\, e^*_\mathrm{k}$ is solution to the discrete wave equation

$$\rho\, e^{n+1}_i + \rho\, e^{n+1}_\mathrm{k} - \frac{\Delta t^2}{\Delta x^2}\left[\tilde{h}^{n+1}_{i+1/2}\left(p^{n+1}_{i+1} - p^{n+1}_i\right) - \tilde{h}^{n+1}_{i-1/2}\left(p^{n+1}_i - p^{n+1}_{i-1}\right]\right) =$$
$$= \rho\, e^n_i + \rho\, e^*_\mathrm{k} - \frac{\Delta t}{\Delta x}\left[\tilde{h}^{n+1}_{i+1/2}\,(\rho\, u_1)^*_{i+1/2} - \tilde{h}^{n+1}_{i-1/2}\,(\rho\, u_1)^*_{i-1/2}\right], \tag{5.71}$$

A further condition on the scheme must be imposed whenever the volume fraction α_1 is not constant in space. In this case, the stiffened gas equation of state applied to each phase provides a more complex closure law of the type $\rho\, e = \alpha_1\, \rho_1\, e_1 + \alpha_2\, \rho_2\, e_2$ or $\rho\, e = \alpha_1\, k_{01} + \alpha_1\, k_{11}\, p + \alpha_2\, k_{02} + \alpha_2\, k_{12}\, p$. Applied to the discrete wave equation, the mixture equation of state gives

$$(\alpha_1)^{n+1}_i = (\alpha_1)^n_i - \frac{\Delta t}{\Delta x}\left[u_1\,(\alpha_1)^*_{i+1/2} - u_1\,(\alpha_1)^*_{i-1/2}\right], \tag{5.72}$$

that is, at least when the velocity field u_1 is a constant, the scheme for the update of α_1 must coincide with one in flux form, for some appropriate choice of $(\alpha_1)^*_{i+1/2}$.

For a constant velocity field, the nonconservative products not associated with pure convection of the volume fraction vanish and, the update scheme reads

$$(\alpha_1)^{n+1}_i = (\alpha_1)^n_i - \frac{\Delta t}{\Delta x}\left(D^{\alpha_1}_{i+1/2} + D^{\alpha_1}_{i-1/2}\right) \tag{5.73}$$

with the path-conservative fluctuations as well as the numerical dissipation from the Rusanov flux collected in

$$\begin{aligned} D^{\alpha_1}_{i+1/2} &= u_1\,\frac{1}{2}\,\Big[(\alpha_1)_R - (\alpha_1)_L\Big]_{i+1/2} - |u_1|\,\frac{1}{2}\,\Big[(\alpha_1)_R - (\alpha_1)_L\Big]_{i+1/2}, \\ D^{\alpha_1}_{i-1/2} &= u_1\,\frac{1}{2}\,\Big[(\alpha_1)_R - (\alpha_1)_L\Big]_{i-1/2} + |u_1|\,\frac{1}{2}\,\Big[(\alpha_1)_R - (\alpha_1)_L\Big]_{i-1/2}. \end{aligned} \tag{5.74}$$

This automatically gives raise to an upwind discretisation that suggests the interpolated values of $\alpha^*_{i+1/2}$ should be computed with the same upwinding rule

$$(\alpha_1)^*_{i+1/2}\Big((\alpha_1)_L,\ (\alpha_1)_R\Big) = \frac{1+\operatorname{sgn}(u_1)}{2}\,(\alpha_1)_L + \frac{1-\operatorname{sgn}(u_1)}{2}\,(\alpha_1)_R, \tag{5.75}$$

for any left and right states $(\alpha_1)_L$ and $(\alpha_1)_R$ obtained from the predictor step of the MUSCL–Hancock scheme. Then it can be verified that for any distribution of volume fraction α_1 and density ρ the discrete wave equation will indeed preserve constant-pressure, constant-velocity solutions exactly.

6 Numerical results

6.1 Semi-analytic integration techniques for finite rate relaxation processes in Baer–Nunziato models

In this Section, we present some computational experiments validating the novel exponential integrator for relaxation processes (pressure and velocity relaxation) in Baer–Nunziato models described in Section 4.2. The main achievements, published in [86], are:

- The proposed semi-analytical integrator experimentally achieves second order convergence (Figure 6.3) and, more importantly, the approach can provide accurate solutions to challenging problems with minimal computational effort, see especially Figure 6.1, in which an initial extremely fast explosion in one of the phase pressures and the subsequent relaxation to steady state are accurately resolved within four sub-timesteps.
- The solver can be used in the infinitely stiff relaxation regime $\lambda \to \infty$, $\nu \to \infty$ to compute solutions of the single pressure, single velocity Kapila [201] model as a limit of the more general Baer–Nunziato system [10, 253, 284]. See for example Figures 6.4 and 6.5.
- While efficient and robust in the stiff relaxation limit, the solver can compute solutions for any finite relaxation rate, without any selector switches. An example is shown in Figure 6.6.

First, we provide validation of the proposed semi-analytical integration method by computing some solutions to the ODE system (4.9)–(4.13) and comparing the results with a reference solution obtained from a sixth order, fully implicit, Runge–Kutta–Gauss–Legendre method [63] (labelled RKGL3) employing adaptive timestepping (test problems A1 and A2, Figures 6.1 and 6.2). Furthermore, test problem A1 is employed also for carrying out a convergence study of the scheme (Figure 6.3), showing that second order convergence is easily achieved. The initial data for the ODE tests are, for test A1,

$$u_1^0 = -5\,\mathrm{m\,s^{-1}}, \quad u_2^0 = 5\,\mathrm{m\,s^{-1}}, \quad p_1^0 = 0.1\,\mathrm{Pa}, \quad p_2^0 = 20\,\mathrm{Pa}, \quad \alpha_1^0 = 0.9, \tag{6.1}$$

while for test A2,

$$u_1^0 = 0\,\mathrm{m\,s^{-1}}, \quad u_2^0 = 0\,\mathrm{m\,s^{-1}}, \quad p_1^0 = 2.0 \times 10^8\,\mathrm{Pa}, \quad p_2^0 = 1\,\mathrm{Pa}, \quad \alpha_1^0 = 0.4. \tag{6.2}$$

The parametric data are, for test A1,

$$\begin{aligned} &\alpha_1\,\rho_1 = 1.0\,\mathrm{kg\,m^{-3}}, \quad \alpha_2\,\rho_2 = 4.0\,\mathrm{kg\,m^{-3}}, \quad \gamma_1 = 6, \quad \gamma_2 = 1.4, \\ &\Pi_1 = 0\,\mathrm{Pa}, \quad \Pi_2 = 0\,\mathrm{Pa}, \quad \lambda = 10^9\,\mathrm{kg\,m^{-1}\,s^{-1}}, \quad \nu = 10\,\mathrm{Pa^{-1}\,s^{-1}}. \end{aligned} \tag{6.3}$$

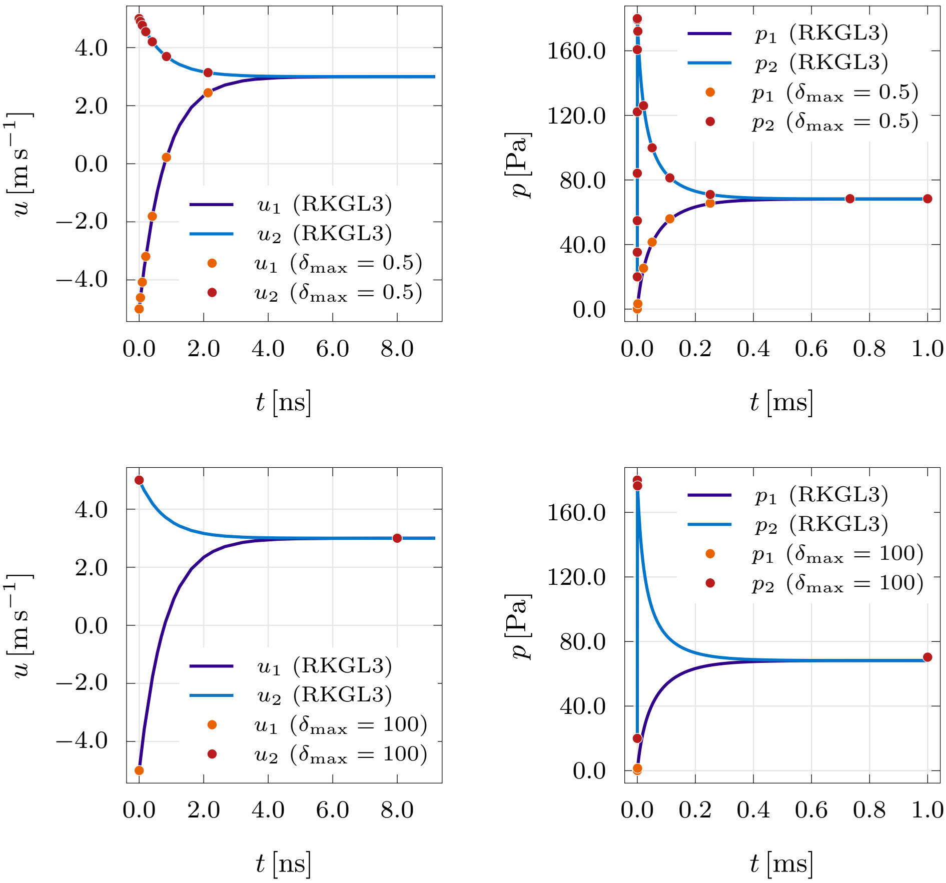


Figure 6.1. Time evolution of velocities and pressures for test problem A1. In the top frames the linearisation tolerance parameter is set as $\delta_{\max} = 0.5$, employing 15 timesteps to reach the final time of 1.0 ms, while in the bottom frames we impose an extremely loose tolerance $\delta_{\max} = 100$, still showing good agreement with the reference solution but using only 4 timesteps for the full run.

and for test A2,

$$\begin{aligned} &\alpha_1\,\rho_1 = 780.0\,\mathrm{kg\,m^{-3}}, \quad \alpha_2\,\rho_2 = 0.22\,\mathrm{kg\,m^{-3}}, \quad \gamma_1 = 6, \quad \gamma_2 = 1.4, \\ &\Pi_1 = 100\,\mathrm{Pa}, \quad \Pi_2 = 0\,\mathrm{Pa}, \quad \lambda = 10^9\,\mathrm{kg\,m^{-1}\,s^{-1}}, \quad \nu = 10\,\mathrm{Pa^{-1}\,s^{-1}}. \end{aligned} \tag{6.4}$$

Then, we show an application of the method in the solution of the mixture-energy-consistent formulation of the six-equation reduced Baer–Nunziato model

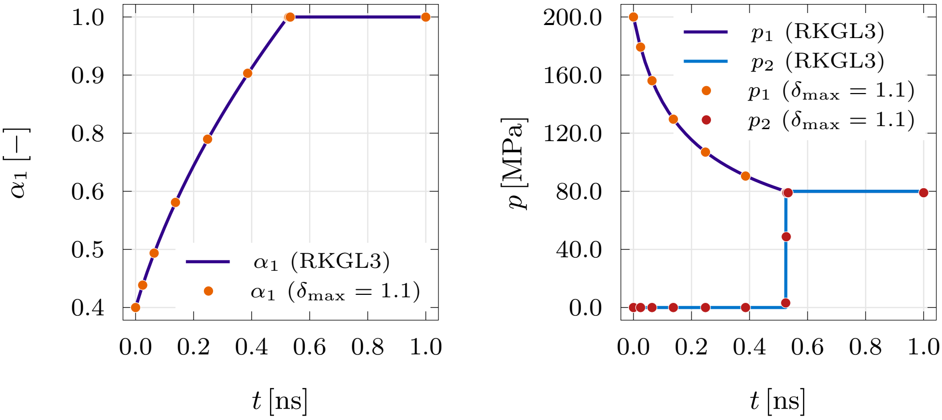


Figure 6.2. Time evolution of volume fraction and pressure for test problem A2. The solution is well captured in 11 timesteps, using a linearisation tolerance $\delta_{\max} = 1.1$.

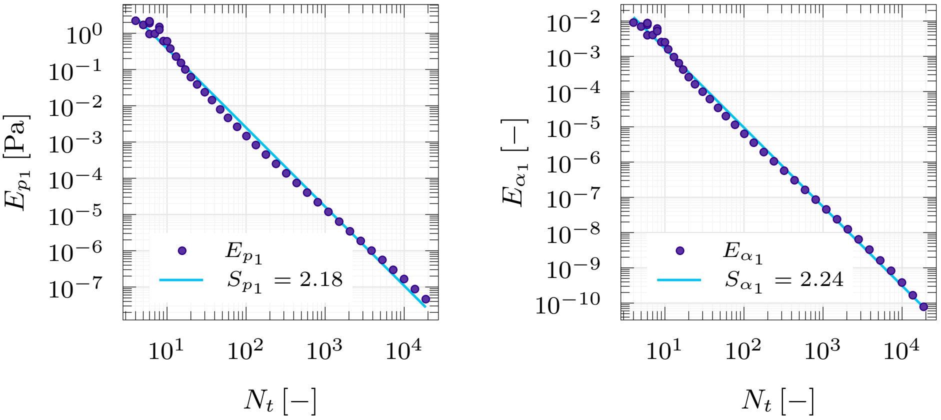


Figure 6.3. Convergence results relative to 40 runs of test problem A1. On the bilogarithmic plane, the slopes of the regression lines are $S_{p_1} = 2.18$ and $S_{\alpha_1} = 2.24$ for the variables p_1 and α_1 respectively, indicating second order convergence.

forwarded in [253]. For these simulations the interface pressure is computed as

$$p_I = \frac{Z_2\, p_1 + Z_1\, p_2}{Z_1 + Z_2}, \quad \text{with} \quad Z_1 = \rho_1\, a_1 \text{ and } Z_2 = \rho_2\, a_2. \tag{6.5}$$

The first two shock-tube problems (from [253, 285]), show that the method is able to deal with very stiff ($\nu = 10^{20}\,\mathrm{Pa}^{-1}\,\mathrm{s}^{-1}$) sources, and in particular in

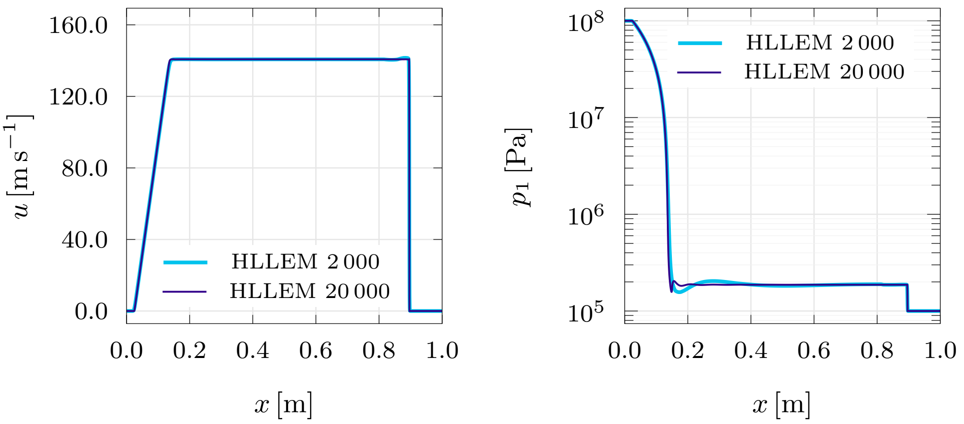


Figure 6.4. Solution of test Problem RP1 on two uniform meshes of 2 000 cells and 20 000 cells respectively, showing convergence with respect to mesh refinement.

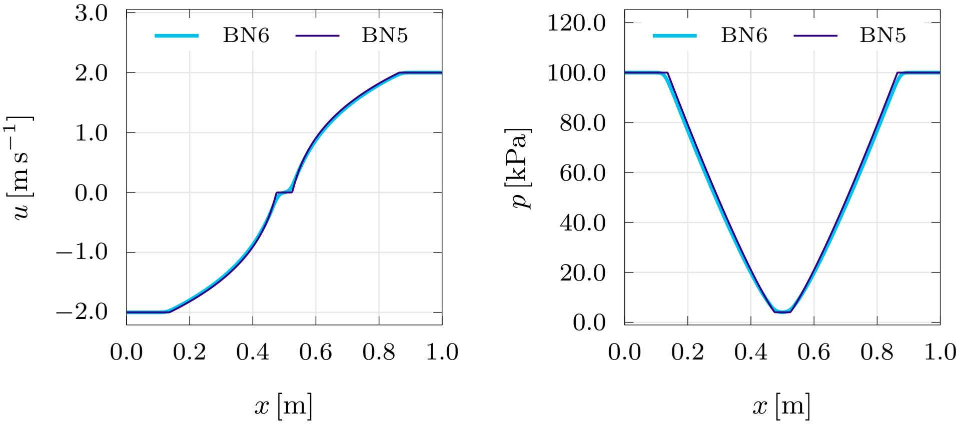


Figure 6.5. Solution of test Problem RP2 computed from the six-equation Baer–Nunziato model (BN6) with stiff relaxation, compared with the five-equation Kapila model (BN5), showing convergence to the limit reduced model.

Figure 6.4 (RP1, a liquid-vapour dodecane shock tube featuring a strong right-moving shockwave) we show mesh convergence of the solution by comparing two runs, both employing the HLLEM Riemann solver proposed in [103], on two different meshes consisting of 2 000 uniform control volumes and 20 000 control volumes respectively, with a computational domain delimited by $x \in [0\,\mathrm{m}, 1\,\mathrm{m}]$. In Figure 6.5 (RP2, two diverging rarefaction waves in liquid water) we then show that, with very stiff relaxation ($\nu = 10^{20}\,\mathrm{Pa}^{-1}\,\mathrm{s}^{-1}$), the solution matches the one computed by solving directly the five-equation instantaneous equilibrium model [201], again using a mesh consisting of 2 000 uniform cells for the six-equation model and a mesh of 20 000 uniform cells for the reference solution, and in particular, rarefaction waves propagate at the same speed for both models. All tests are run using a second order path-conservative MUSCL-Hancock scheme with $k_{\mathrm{CFL}} = 0.95$. The first Riemann Problem (RP1) is set up with uniform liquid and vapour densities $\rho_1^{\mathrm{L}} = \rho_1^{\mathrm{R}} = 500\,\mathrm{kg\,m^{-1}\,s^{-1}}$ and $\rho_2^{\mathrm{L}} = \rho_2^{\mathrm{L}} = 2.0\,\mathrm{kg\,m^{-3}}$, uniform velocity $u^{\mathrm{L}} = u^{\mathrm{R}} = 0\,\mathrm{m\,s^{-1}}$, a jump in pressure given by $p_1^{\mathrm{L}} = p_2^{\mathrm{L}} = 100\,\mathrm{MPa}$, $p_1^{\mathrm{R}} = p_2^{\mathrm{R}} = 100\,\mathrm{kPa}$, almost pure liquid on the left side of the initial discontinuity ($\alpha_1^{\mathrm{L}} = 1 - 10^{-8}$), and almost pure vapor on the right side ($\alpha_1^{\mathrm{R}} = 10^{-8}$). The discontinuity is initially found at $x = 0.75\,\mathrm{m}$, and the end time is $t_{\mathrm{end}} = 473\,\mu\mathrm{s}$. The parameters of the stiffened gas EOS are $\gamma_1 = 2.35$, $\gamma_2 = 1.025$, $\Pi_1 = 400\,\mathrm{MPa}$, $\Pi_2 = 0$.

The second Riemann Problem (RP2) is initialised with constant liquid and vapour densities $\rho_1^{\mathrm{L}} = \rho_1^{\mathrm{R}} = 1150\,\mathrm{kg\,m^{-1}\,s^{-1}}$, $\rho_2^{\mathrm{L}} = \rho_2^{\mathrm{L}} = 0.63\,\mathrm{kg\,m^{-3}}$, constant pressure $p_1^{\mathrm{L}} = p_2^{\mathrm{L}} = p_1^{\mathrm{R}} = p_2^{\mathrm{R}} = 100\,\mathrm{kPa}$, constant liquid volume fraction $\alpha_1^{\mathrm{L}} = \alpha_1^{\mathrm{R}} = 0.99$, and a jump in velocity (initially located at $x = 0.5\,\mathrm{m}$) such

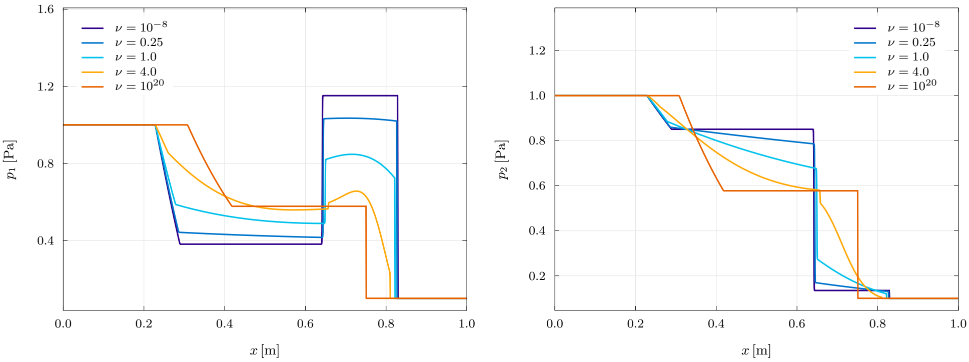


Figure 6.6. Behaviour of the pressure variables p_1 and p_2 in RP3 with several values of ν. It is clear that, in the stiff regime ($\nu = 10^{20}\,\mathrm{Pa}^{-1}\,\mathrm{s}^{-1}$), p_1 and p_2 converge to the same value, while they evolve in a completely distinct fashion if relaxation is set to act on longer timescales.

that $u^{\mathrm{L}} = -2.0\,\mathrm{m\,s^{-1}}$ and $u^{\mathrm{R}} = 2.0\,\mathrm{m\,s^{-1}}$. The final time is $t_{\mathrm{end}} = 3.2\,\mathrm{ms}$ and for this test the parameters of the equation of state $\gamma_1 = 2.35$, $\gamma_2 = 1.43$, $\Pi_1 = 1\,\mathrm{GPa}$, $\Pi_2 = 0$.

Finally, in Figure 6.6 we show the behaviour of the solution of a third Riemann problem (RP3) with several different values of the pressure relaxation parameter ν (ranging from $10^{-8}\,\mathrm{Pa^{-1}\,s^{-1}}$ to $10^{20}\,\mathrm{Pa^{-1}\,s^{-1}}$), highlighting the vast range of solution structures that can be obtained not only with stiff relaxation (the pressure profiles p_1 and p_2 coincide) or in total absence of it, but also with finite values of the relaxation time scale. For RP3, the initial data on the left are

$$\begin{aligned} &\rho_1^{\mathrm{L}} = 1.0\,\mathrm{kg\,m^{-1}\,s^{-1}}, \quad \rho_2^{\mathrm{L}} = 0.2\,\mathrm{kg\,m^{-3}}, \quad u^{\mathrm{L}} = 0.0\,\mathrm{m\,s^{-1}},\\ &p_1^{\mathrm{L}} = 1.0\,\mathrm{Pa}, \quad p_2^{\mathrm{L}} = 1.0\,\mathrm{Pa}, \quad \alpha_1^{\mathrm{L}} = 0.55, \end{aligned} \tag{6.6}$$

while on the right one has

$$\begin{aligned} &\rho_1^{\mathrm{R}} = 0.125\,\mathrm{kg\,m^{-3}}, \quad \rho_2^{\mathrm{R}} = 2.0\,\mathrm{kg\,m^{-3}}, \quad u^{\mathrm{R}} = 0.0\,\mathrm{m\,s^{-1}},\\ &p_1^{\mathrm{R}} = 0.1\,\mathrm{Pa}, \quad p_2^{\mathrm{R}} = 0.1\,\mathrm{Pa}, \quad \alpha_1^{\mathrm{R}} = 0.45. \end{aligned} \tag{6.7}$$

The initial jump is located at $x = 0.6\,\mathrm{m}$, the domain is $x \in [0\,\mathrm{m},\ 1\,\mathrm{m}]$ and the final time is $t_{\mathrm{end}} = 0.15\,\mathrm{s}$. The parameters of the stiffened gas EOS are $\gamma_1 = 2.0$, $\gamma_2 = 1.4$, $\Pi_1 = 2.0\,\mathrm{Pa}$, $\Pi_2 = 0.0\,\mathrm{Pa}$.

6.2 High order ADER schemes for two-phase flow with surface tension

In this Section, we present the results obtained by applying the ADER methods to all three variants of the Schmidmayer *et al* [286] model: the original weakly hyperbolic formulation (2.11), the hyperbolic non-conservative symmetrisable Godunov–Powell form (2.25), and the hyperbolic GLM curl-cleaning formulation (2.43). As we have already mentioned, other variants of the model were also tested (*e.g.* Godunov–Powell + GLM, or GLM with extra terms in the energy equation (2.44)) but these variants show very similar results, at least for the considered test cases, to the first three formulations and therefore, are not presented here. We should clarify that, while all variants of the diffuse interface model here considered satisfy the stationary constraint $\nabla \times \mathbf{b}$ at the continuum level, if it holds for the initial condition, in all computations we do *not* enforce it at the discrete level by modifying the numerical scheme to account for this. We are instead investigating how each model responds to the small numerical errors that are inevitably present in the discrete representation of the initial condition, or that are generated as a result of integration errors. These results are published in [87].

6.2.1 Numerical convergence results

As a first benchmark, we conduct a numerical convergence study on a smooth test problem, for which we have derived the exact solution in Section 2.2.6. The test is very similar to the well known isentropic vortex advection problem [299] for the Euler equations of gasdynamics: a steady state solution is initialised at time $t = 0$ in a uniform flow field $\mathbf{u} = (u_0,\ v_0,\ 0)^\mathsf{T}$ and evolved with periodic boundary conditions on a rectangular domain of edge lengths L_x and L_y. Due to the Galilean invariance of the governing equations, the exact solution is obtained by transporting the initial condition with the uniform flow speed.

Problem setup. The initial condition for the liquid volume fraction α_1 is given according to the chosen colour function profile, but bounding it between the two values $\alpha_{\min} = 0.01$ and $\alpha_{\max} = 0.99$, so that we have

$$\alpha_1(r) = \alpha_{\min} + \frac{\alpha_{\max} - \alpha_{\min}}{2} \operatorname{erfc}\left(\frac{r'_* - 1}{k_\varepsilon}\right). \tag{6.8}$$

Since also the density fields should be transported with uniform velocity regardless of their value, we decided not to impose a constant value for ρ_1

and ρ_2, but rather specify a more interesting periodic two-dimensional wave configuration in the form

$$\rho_1(x,\ y) = \overline{\rho_1} + \delta\,\overline{\rho_1}\,\sin\left[\omega\,(2\,x + y)\right]\,\cos\left[\omega\,(x - 2\,y)\right], \tag{6.9}$$

$$\rho_2(x,\ y) = \overline{\rho_2} + \delta\,\overline{\rho_2}\,\sin\left[\omega\,(x - 2\,y)\right]\,\cos\left[\omega\,(2\,x + y)\right]. \tag{6.10}$$

The numerical values for the test are $u_0 = v_0 = 1\,\mathrm{m\,s^{-1}}$, $R = 1\,\mathrm{m}$, $k_\varepsilon = 0.3$, $\sigma = 1\,\mathrm{N\,m^{-1}}$, $p_{\mathrm{atm}} = 1\,\mathrm{Pa}$, $\overline{\rho_1} = 1000\,\mathrm{kg\,m^{-3}}$, $\overline{\rho_2} = 1\,\mathrm{kg\,m^{-3}}$, $\delta = 0.1$, $\omega = \pi/3\,\mathrm{m^{-1}}$, $\Pi_1 = 20\,\mathrm{Pa}$, $\Pi_2 = 0\,\mathrm{Pa}$, $\gamma_1 = 4$, $\gamma_2 = 1.4$.

The computational domain is the square $\Omega = [-3\,\mathrm{m},\ 3\,\mathrm{m}] \times [-3\,\mathrm{m},\ 3\,\mathrm{m}]$ so that at $t = 12\,\mathrm{s}$ we expect the water column to have completed two full advection cycles. We evolve the system from time $t = 0$ to time $t = 12\,\mathrm{s}$

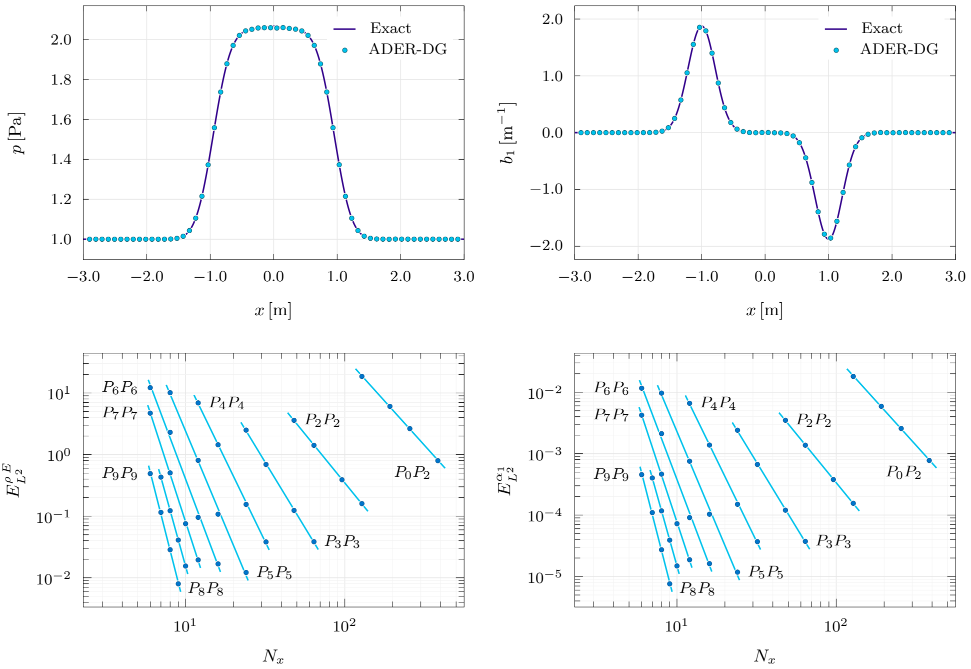


Figure 6.7. Numerical convergence results. In the top row, one-dimensional cuts (60 equidistant samples along the x axis) of the pressure p and the x component of the interface field b_1 are presented at time $t = 12\,\mathrm{s}$ (at the end of the second advection cycle), computed with a $\mathbb{P}_9\mathbb{P}_9$ ADER-DG scheme on a uniform grid of 6^2 elements. In the bottom row we show logarithmic least-squares regression lines of the L^2 error norms of total energy density $\rho\,E$ and of the liquid volume fraction α_1 for the ADER-DG $\mathbb{P}_N\mathbb{P}_N$ schemes of orders 3 to 10 and for the third order $\mathbb{P}_0\mathbb{P}_2$ ADER-WENO Finite Volume scheme.

Table 6.1. Numerical convergence results regarding the ADER-DG $\mathbb{P}_N\mathbb{P}_N$ schemes of nominal orders of accuracy 3 to 10 and the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ Finite Volume scheme for all conserved variables. The values reported in the Table are computed from a logarithmic least-square fit of the L^2 error norms as shown in Figures 6.7.

		$\alpha_1\,\rho_1$	$\alpha_2\,\rho_2$	$\rho\,u$	$\rho\,v$	$\rho\,E$	α_1	b_1	b_2
$\mathbb{P}_0\mathbb{P}_2$	$\mathcal{O}_{L^1}$	2.9	3.1	2.9	2.9	2.9	2.9	2.8	2.8
	$\mathcal{O}_{L^2}$	2.9	3.0	2.9	2.9	2.9	2.9	2.8	2.8
	$\mathcal{O}_{L^\infty}$	2.8	3.0	2.8	2.8	2.8	2.8	2.7	2.7
$\mathbb{P}_2\mathbb{P}_2$	$\mathcal{O}_{L^1}$	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2
	$\mathcal{O}_{L^2}$	3.2	3.2	3.2	3.2	3.2	3.2	3.2	3.2
	$\mathcal{O}_{L^\infty}$	3.1	3.1	3.1	3.1	3.1	3.1	3.1	3.1
$\mathbb{P}_3\mathbb{P}_3$	$\mathcal{O}_{L^1}$	4.3	4.3	4.3	4.3	4.3	4.3	4.3	4.3
	$\mathcal{O}_{L^2}$	4.3	4.3	4.3	4.3	4.3	4.2	4.3	4.3
	$\mathcal{O}_{L^\infty}$	4.2	4.1	4.2	4.2	4.2	4.2	4.3	4.3
$\mathbb{P}_4\mathbb{P}_4$	$\mathcal{O}_{L^1}$	5.5	6.1	5.5	5.5	5.5	5.5	5.5	5.5
	$\mathcal{O}_{L^2}$	5.3	5.6	5.3	5.3	5.3	5.3	5.3	5.3
	$\mathcal{O}_{L^\infty}$	5.1	5.3	5.1	5.1	5.0	5.1	5.0	5.0
$\mathbb{P}_5\mathbb{P}_5$	$\mathcal{O}_{L^1}$	6.5	7.3	6.5	6.5	6.5	6.4	6.4	6.4
	$\mathcal{O}_{L^2}$	6.2	6.8	6.2	6.2	6.2	6.2	6.1	6.1
	$\mathcal{O}_{L^\infty}$	5.8	6.6	5.8	5.8	5.8	5.7	5.8	5.8
$\mathbb{P}_6\mathbb{P}_6$	$\mathcal{O}_{L^1}$	7.2	8.6	7.2	7.2	7.2	7.2	7.1	7.1
	$\mathcal{O}_{L^2}$	6.9	8.0	6.9	6.9	6.9	6.9	6.8	6.8
	$\mathcal{O}_{L^\infty}$	6.6	8.1	6.6	6.6	6.6	6.5	6.4	6.3
$\mathbb{P}_7\mathbb{P}_7$	$\mathcal{O}_{L^1}$	8.3	10.0	8.3	8.3	8.3	8.2	8.1	8.1
	$\mathcal{O}_{L^2}$	8.0	9.4	8.0	8.0	8.0	7.9	7.7	7.7
	$\mathcal{O}_{L^\infty}$	7.8	9.4	7.8	7.8	7.8	7.4	7.1	7.1
$\mathbb{P}_8\mathbb{P}_8$	$\mathcal{O}_{L^1}$	9.7	11.1	9.7	9.7	9.7	9.6	9.6	9.6
	$\mathcal{O}_{L^2}$	9.3	10.2	9.3	9.3	9.3	9.2	9.2	9.2
	$\mathcal{O}_{L^\infty}$	8.7	9.1	8.7	8.7	8.7	8.8	8.4	8.4
$\mathbb{P}_9\mathbb{P}_9$	$\mathcal{O}_{L^1}$	10.7	11.7	10.7	10.7	10.7	10.6	10.6	10.6
	$\mathcal{O}_{L^2}$	10.2	11.0	10.2	10.2	10.2	10.1	10.1	10.1
	$\mathcal{O}_{L^\infty}$	9.9	11.2	9.9	9.9	9.9	10.0	9.7	9.7

Table 6.2. Numerical convergence results regarding the ADER-DG $\mathbb{P}_N\mathbb{P}_N$ schemes of nominal orders of accuracy 3 to 6 and the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ Finite Volume scheme for the liquid volume fraction α_1. With N_x we indicate the number of cells in one row of the Cartesian computational grid.

	N_x	E_{L^1}	E_{L^2}	E_{L^∞}	$\mathcal{O}_{L^1}$	$\mathcal{O}_{L^2}$	$\mathcal{O}_{L^\infty}$
$\mathbb{P}_0\mathbb{P}_2$	128	4.32×10^{-2}	1.81×10^{-2}	1.42×10^{-2}	–	–	–
	192	1.41×10^{-2}	5.90×10^{-3}	4.76×10^{-3}	2.76	2.76	2.70
	256	6.10×10^{-3}	2.57×10^{-3}	2.08×10^{-3}	2.92	2.89	2.87
	384	1.83×10^{-3}	7.75×10^{-4}	6.29×10^{-4}	2.97	2.96	2.95
$\mathbb{P}_2\mathbb{P}_2$	48	7.91×10^{-3}	3.49×10^{-3}	3.24×10^{-3}	–	–	–
	64	3.07×10^{-3}	1.37×10^{-3}	1.32×10^{-3}	3.29	3.25	3.13
	96	8.47×10^{-4}	3.79×10^{-4}	3.77×10^{-4}	3.17	3.17	3.09
	128	3.47×10^{-4}	1.55×10^{-4}	1.56×10^{-4}	3.10	3.10	3.07
$\mathbb{P}_3\mathbb{P}_3$	24	5.50×10^{-3}	2.40×10^{-3}	2.35×10^{-3}	–	–	–
	32	1.46×10^{-3}	6.67×10^{-4}	7.04×10^{-4}	4.61	4.46	4.19
	48	2.62×10^{-4}	1.20×10^{-4}	1.27×10^{-4}	4.24	4.23	4.22
	64	8.18×10^{-5}	3.73×10^{-5}	3.87×10^{-5}	4.05	4.06	4.14
$\mathbb{P}_4\mathbb{P}_4$	12	1.76×10^{-2}	6.60×10^{-3}	5.52×10^{-3}	–	–	–
	16	3.24×10^{-3}	1.38×10^{-3}	1.48×10^{-3}	5.89	5.44	4.59
	24	3.29×10^{-4}	1.50×10^{-4}	1.71×10^{-4}	5.64	5.48	5.32
	32	8.15×10^{-5}	3.69×10^{-5}	4.03×10^{-5}	4.86	4.86	5.02
$\mathbb{P}_5\mathbb{P}_5$	8	2.87×10^{-2}	9.66×10^{-3}	6.66×10^{-3}	–	–	–
	12	1.88×10^{-3}	7.53×10^{-4}	7.87×10^{-4}	6.72	6.29	5.27
	16	2.35×10^{-4}	1.03×10^{-4}	1.32×10^{-4}	7.23	6.91	6.20
	24	2.59×10^{-5}	1.18×10^{-5}	1.36×10^{-5}	5.43	5.36	5.62

for all $\mathbb{P}_N\mathbb{P}_N$, $N = 2,\ 3,\ \ldots,\ 9$ schemes with local space-time DG predictor step performed in the primitive variable variant, using the HLL flux. Since in this work we employ extensively the ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ method with reconstruction and predictor in primitive variables, in the role of *a posteriori* subcell limiter scheme, we also report convergence results for this method, when applied to the main grid directly, without additional *a posteriori* limiting.

The employed mathematical model is the nonconservative hyperbolic Godunov–Powell formulation. The results confirm that the error norms of the conserved variables decrease at a rate that is in agreement with the nominal order of accuracy of the scheme, and are summarised in Tables 6.1, 6.2 and 6.3. In Tables 6.2 and 6.3, we report the error norms and convergence rates for the

Table 6.3. Numerical convergence results regarding the ADER-DG $\mathbb{P}_N\mathbb{P}_N$ schemes of nominal orders of accuracy 7 to 10 for the liquid volume fraction α_1. With N_x we indicate the number of cells in one row of the Cartesian computational grid.

	N_x	E_{L^1}	E_{L^2}	E_{L^∞}	$\mathcal{O}_{L^1}$	$\mathcal{O}_{L^2}$	$\mathcal{O}_{L^\infty}$
$\mathbb{P}_6\mathbb{P}_6$	6	3.62×10^{-2}	1.16×10^{-2}	8.98×10^{-3}	–	–	–
	8	5.97×10^{-3}	2.12×10^{-3}	1.74×10^{-3}	6.27	5.92	5.71
	12	2.26×10^{-4}	9.10×10^{-5}	1.07×10^{-4}	8.07	7.76	6.88
	16	3.65×10^{-5}	1.62×10^{-5}	1.62×10^{-5}	6.34	6.00	6.55
$\mathbb{P}_7\mathbb{P}_7$	6	1.24×10^{-2}	4.23×10^{-3}	3.48×10^{-3}	–	–	–
	8	1.17×10^{-3}	4.64×10^{-4}	5.41×10^{-4}	8.19	7.68	6.47
	10	1.69×10^{-4}	7.24×10^{-5}	1.01×10^{-4}	8.69	8.33	7.54
	12	4.22×10^{-5}	1.87×10^{-5}	2.07×10^{-5}	7.60	7.41	8.68
$\mathbb{P}_8\mathbb{P}_8$	7	9.88×10^{-4}	4.01×10^{-4}	4.32×10^{-4}	–	–	–
	8	2.69×10^{-4}	1.17×10^{-4}	1.42×10^{-4}	9.73	9.23	8.35
	9	8.71×10^{-5}	3.91×10^{-5}	5.80×10^{-5}	9.59	9.31	7.60
	10	3.20×10^{-5}	1.49×10^{-5}	1.79×10^{-5}	9.50	9.17	11.16
$\mathbb{P}_9\mathbb{P}_9$	6	1.19×10^{-3}	4.55×10^{-4}	5.21×10^{-4}	–	–	–
	7	2.53×10^{-4}	1.10×10^{-4}	1.49×10^{-4}	10.02	9.24	8.13
	8	5.97×10^{-5}	2.72×10^{-5}	3.86×10^{-5}	10.83	10.43	10.11
	9	1.62×10^{-5}	7.59×10^{-6}	9.05×10^{-6}	11.07	10.84	12.31

liquid volume fraction α_1, for numerical schemes of order up to 10. In Table 6.1, numerical details concerning the regression lines of the L_2 norms of the error for all variables are given. Since the interface field **b** is evolved as a vector of independent state variables, as opposed standard schemes which differentiate the colour function and thus lose one order of accuracy for the discrete gradient, in our scheme the nominal high order convergence rate is achieved for the gradient field **b** as well. The regression lines for the mixture energy density $\rho\,E$ and for the liquid volume fraction α_1 are plotted in Figure 6.7, where also one-dimensional cuts through the numerical solution are shown along the x axis, comparing the exact solution derived in Section 2.2.6 with 60 uniformly spaced samples from a computation employing the $\mathbb{P}_9\mathbb{P}_9$ ADER-DG scheme on a very coarse uniform Cartesian grid composed of only 6^2 total elements.

6.2.2 Interaction between a shock wave and a water column

With this test case we want to show that the ADER-DG schemes with *a posteriori* subcell Finite Volume limiter are capable of computing solutions of nonconservative hyperbolic systems not only in smooth regions, but can also robustly deal with shock waves while preserving the sharp profile that characterises these flow features. Specifically, we want to reproduce the results of the experiment of Igra and Takayama [193], as it was done in [286].

The simulation setup is as follows: a cylindrical water column of radius $R = 3.2\,\mathrm{mm}$ is initialised at the origin of the computational domain following the exact solution given in Section 2.2.6. The interface thickness parameter is $k_\varepsilon = 1/20$ and the surface tension coefficient is that of water, i.e. $\sigma = 0.072\,\mathrm{N\,m^{-1}}$. Outside the water column, the pressure is set to $p = p_{\mathrm{atm}} = 10^5\,\mathrm{Pa}$ and the liquid volume fraction is $\alpha_1 = 10^{-5}$, while inside the droplet we have $\alpha_1 = 1 - 10^{-5}$. The density for water and air are taken as $\rho_1 = 998.2\,\mathrm{kg\,m^{-3}}$ and $\rho_2 = 1.18\,\mathrm{kg\,m^{-3}}$, respectively. The parameters for the equation of state are the usual ideal gas parameters for air $\Pi_2 = 0\,\mathrm{Pa}$ and $\gamma_2 = 1.4$, while for the water we wanted to reproduce the correct speed of sound in the pure liquid phase so we set $\Pi_1 = 4.7\times10^8\,\mathrm{Pa}$ and $\gamma_1 = 4.7$. Since a perfectly pure phase is

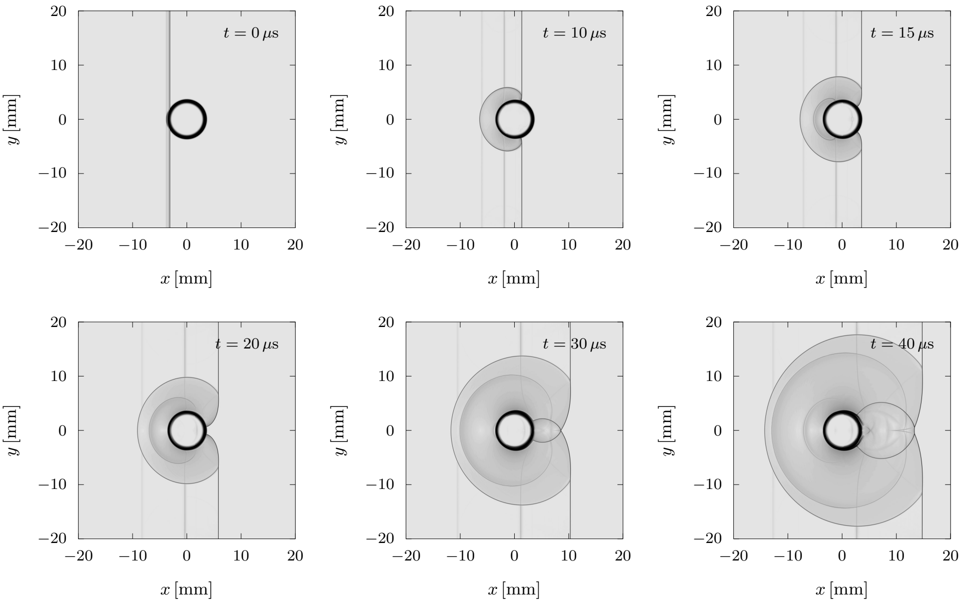


Figure 6.8. Numerical schlieren images of the early stages of the shock–water column interaction problem computed with a $\mathbb{P}_3\mathbb{P}_3$ ADER-DG scheme and a TVD subcell limiter on a mesh of spacing $\Delta x = \Delta y = 0.0625\,\mathrm{mm}$.

never present in our test, the speed of sound in water is slightly smaller than the correct one, but still significantly larger than the speed of sound in air, where the correct speed of sound is very well reproduced for $\alpha_1 = 10^{-5}$. A shock moving at speed $S_s = 1.3\,a_{\text{air}}$ in the x direction (a_{air} being the speed of sound in air, that is, we have a Mach 1.3 shock), with post-shock state computed following the jump relations found in [283], is localised, at time $t = -1.0\,\mu$s, when the simulation is started, at a distance $\delta = S_s\,\mathrm{m\,s^{-1}} \cdot 1.0\,\mu$s

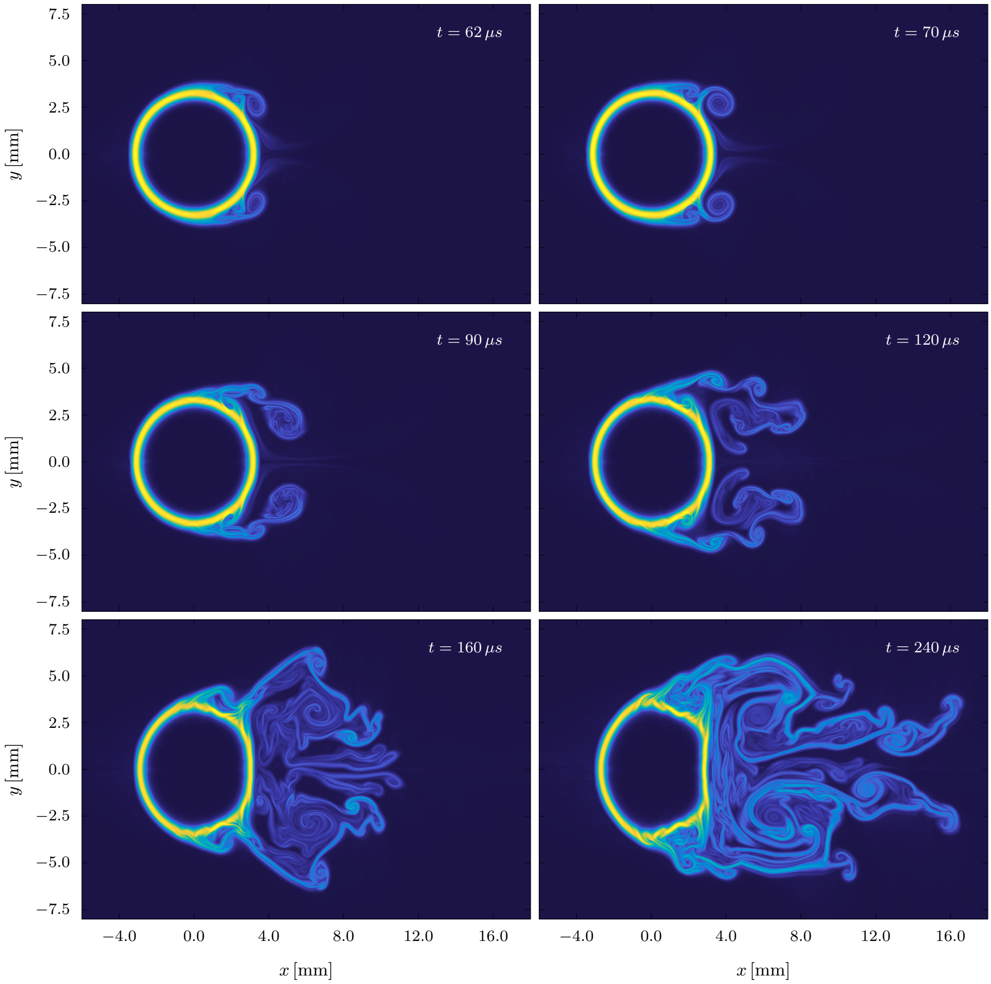


Figure 6.9. Visualisation of the interface transport by means of the filled contour plot of $\sigma\ \|\mathbf{b}\|^{1/4}$ for the shock–water column interaction problem. The solution has been obtained with a $\mathbb{P}_3\mathbb{P}_3$ ADER-DG scheme and a TVD subcell limiter on a uniform mesh of spacing $\Delta x = \Delta y = 0.0625\,$mm.

(rounded to the nearest element edge) from the *nominal* edge of the droplet (see Figure 6.8 for a snapshot at time $t = 0\,\mu$s). A result of this sharp initialization of the shock profile can be clearly observed in the numerical schlieren images of Figure 6.8, in that two acoustic waves due to the startup error can be seen travelling upstream in the post-shock region.

In order to produce the results discussed in this section, we ran, for convenience, two distinct simulations with different domain sizes, but with the same mesh spacing. One simulation deals with the early phase of the simulation, that is the impact between the shock and the water column and the computational domain is the square $[-20\,\text{mm},\ 20\,\text{mm}] \times [-20\,\text{mm},\ 20\,\text{mm}]$, while for the simulation on longer timescales we adopt a rectangular domain $[-10\,\text{mm},\ 30\,\text{mm}] \times [-10\,\text{mm},\ 10\,\text{mm}]$. For the solution we employ a fourth order ADER-DG $\mathbb{P}_3\mathbb{P}_3$ scheme with primitive variable predictor step, supplemented with a robust second order TVD limiter with primitive variable reconstruction. The element size is the same for both simulations, since we use a grid of 640×640 cells in the former case, and of 320×640 cells in the latter. The numerical fluxes are computed with the HLL approximate Riemann solver.

In order to visualise the flow field, we plot the commonly used numerical schlieren pictures for the early stages of the simulation to highlight the shock-waves and aid comparison with the literature [193, 286], while, for the later stages of the simulation, we employ the key variable of the model, that would be the interface field $\mathbf{b}$, to construct images that are very rich in detail and show quite effectively the complex turbulent structures which develop in this test problem, in a manner that is reminiscent of numerical schlieren pictures, since these are also nonlinearly scaled plots of the magnitude of a gradient.

In Figure 6.8. one can see the first time instants of the numerical experiment: discontinuities are very sharp, travel with the correct speed and in general show good agreement with both the experimental data of [193] and the simulations of [286]. It is then notable that at time $t = 0\,\mu$s some interaction can be observed between the shock and the smoothing region of the water column, which extends symmetrically towards the centre of the water column and towards the environment past the *nominal* edge.

In Figures 6.9, at time $t = 62\,\text{ms}$ we can see the first vortical structures developing around the water column and, at time $t = 120\,\mu$s, Kelvin–Helmoltz-type [179, 314] instabilities are clearly distinguishable, while at time $t = 240\,\mu$s one can also observe the presence of Richtmyer–Meshkov-type instabilities [235, 274].

With this visualization method, the process of formation of filaments in the edge of the water column, which then are drawn into the vortical flows in the wake of the obstacle, is quite apparent.

6.2.3 Droplet transport in two and three space dimensions

In this section, we conduct a systematic study of the stability and accuracy of the two new strongly hyperbolic systems of governing equations that have been proposed in this work, which are both different from the original weakly hyperbolic model introduced in [286]. First, we set up a two-dimensional droplet in equilibrium, as prescribed by the exact solution given in Section 2.2.6, in a uniform velocity field with periodic boundary conditions, and track the time evolution of the domain-averaged curl constraint violations. The problem is analogous as the one used for the convergence study and is chosen because an exact solution for the problem is available, which allows to assess the correctness of the results unequivocally. Differently from what has been done in the convergence study, the sinusoidal density field given in Eq. (6.9), is replaced with two constant density values with ratio $\rho_1/\rho_2 = 1000$. In two space dimensions, the same test is repeated for the original weakly hyperbolic model of [286], for the new hyperbolic formulation using the Godunov–Powell-type nonconservative products (denoted by GPNCP in the plots), and for another three runs with the new augmented hyperbolic GLM curl-cleaning system, with increasing values of the cleaning speed c_h, namely choosing $c_h \in \{10,\ 20,\ 40\}\,\mathrm{m\,s^{-1}}$. For each one of these five choices, we let the computations run up to a final time $t_{\text{end}} = 20.0\,\mathrm{ms}$, which corresponds to 20 full advection cycles, first on a coarse mesh of 16^2 cells, and then on a finer grid counting 32^2 elements, with the ADER-DG $\mathbb{P}_5\mathbb{P}_5$ scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ *a posteriori* subcell limiter. The purpose of these runs is to verify how the different formulations react to mesh refinement and how they compare for a given resolution.

Then we carry out another set of five runs, studying the advection of a three-dimensional droplet by means of the ADER-DG $\mathbb{P}_3\mathbb{P}_3$ scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ *a posteriori* subcell limiter, on a coarse mesh of 16^3 elements, to extend the previous two-dimensional results to the full three-dimensional case.

The droplet has radius $R = 3\,\mathrm{mm}$ and is centred at the origin of a square domain $\Omega = [-6\,\mathrm{mm},\ 6\,\mathrm{mm}] \times [-6\,\mathrm{mm},\ 6\,\mathrm{mm}]$, the liquid and gas density are respectively set to $\rho_1 = 1000\,\mathrm{kg\,m^{-1}}$ and $\rho_2 = 1\,\mathrm{kg\,m^{-1}}$ throughout the domain. The volume fraction follows Eq. (6.8), with $\alpha_{\min} = 0.01$ and $\alpha_{\max} = 0.99$, and the interface field is given by (2.59), with the dimensionless interface thickness parameter being $k_\varepsilon = 1/6$ for the two-dimensional tests and $k_\varepsilon = 1/3$ for the three-dimensional problem, additionally setting $b_3 = 0$ for the two-dimensional runs. The pressure is initialised following the exact solution (2.63), with atmospheric pressure $p_{\text{atm}} = 100\,\mathrm{kPa}$, and the uniform velocity field components are $u_0 = 12\,\mathrm{m\,s^{-1}}$, $v_0 = 12\,\mathrm{m\,s^{-1}}$, and $w_0 = 12\,\mathrm{m\,s^{-1}}$ in three space dimensions or $w_0 = 0\,\mathrm{m\,s^{-1}}$ in two dimensions. The parameters for the equation of state are $\Pi_1 = 1\,\mathrm{MPa}$, $\Pi_2 = 0$, $\gamma_1 = 4$, $\gamma_2 = 1.4$, and the surface tension coefficient is set to $\sigma = 60\,\mathrm{N\,m^{-1}}$.

The results are depicted in Figures 6.10, 6.11, and 6.12. In Figure 6.10, we plot the time evolution of the normalised L_1 and L_2 norms of the curl

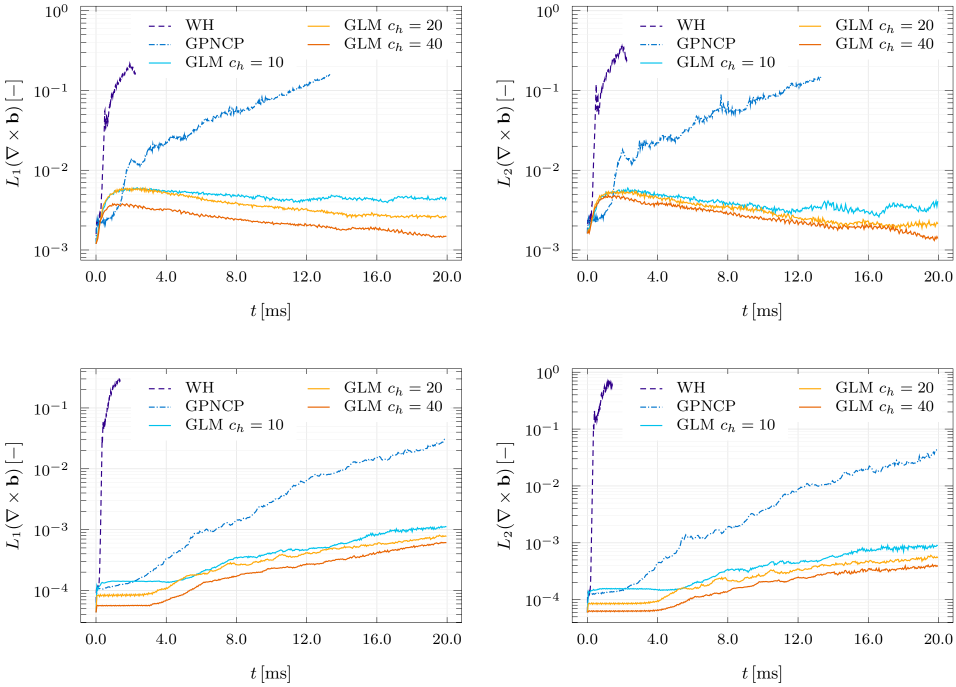


Figure 6.10. Time evolution of the L_1 and L_2 norms of the curl constraint violations for the two-dimensional droplet advection problem. In the top row, the results from a $\mathbb{P}_5\mathbb{P}_5$ ADER-DG scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on a uniform coarse grid composed of 16^2 elements; in the bottom row, the results from the same method, but on a finer mesh composed of 32^2 cells. In both cases, it is verified that curl errors are significantly reduced with the new GLM curl cleaning, with respect to those generated with the nonconservative Godunov–Powell-type formulation of the equations (GPNCP). In the GLM formulation, the constraint violations decrease with increasing cleaning speed c_h, as expected. Furthermore, on the coarser grid, the computation with the Godunov–Powell formulation fails after about 13 advection cycles (13 ms). In no case stable results can be obtained from the original weakly hyperbolic model.

constraint violations, defined as

$$\begin{aligned} L_1(\nabla\times\mathbf{b}) &= \left\{\int_\Omega \sqrt{\operatorname{tr}\left[(\nabla\mathbf{b})^\mathsf{T}\,\nabla\mathbf{b}\right]}\,\bigg|_{t=0}\right\}^{-1}\int_\Omega \sqrt{(\nabla\times\mathbf{b})^\mathsf{T}\,\nabla\times\mathbf{b}}, \\ L_2(\nabla\times\mathbf{b}) &= \left\{\int_\Omega \left|\operatorname{tr}\left[(\nabla\mathbf{b})^\mathsf{T}\,\nabla\mathbf{b}\right]\right|\,\bigg|_{t=0}\right\}^{-1}\int_\Omega \left|(\nabla\times\mathbf{b})^\mathsf{T}\,\nabla\times\mathbf{b}\right|. \end{aligned} \tag{6.11}$$

We observe that in all cases the same trend is apparent: the curl error given by the weakly hyperbolic model quickly grows until the computation terminates with unphysical values at rather early times, while the new strongly hyperbolic

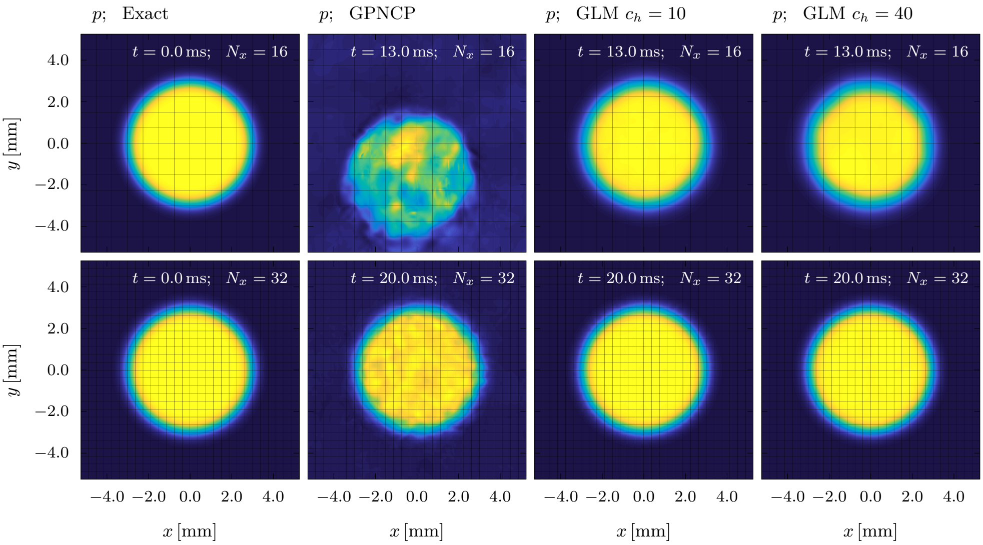


Figure 6.11. Numerical results for the two-dimensional droplet advection test problem. In the first row, we compare the results of the nonconservative Godunov–Powell model and of the augmented GLM curl cleaning system with two different values of cleaning speed c_h with the exact solution. The snapshots of the pressure field are taken at time $t = 13.0\,\text{ms}$, which corresponds to thirteen advection cycles. The scheme used is ADER-DG $\mathbb{P}_5\mathbb{P}_5$ with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter and the mesh is composed of 16^2 square control volumes. In the second row, the same comparison is carried out again, but on a finer mesh of 32^2 elements at time $t = 20.0\,\text{ms}$, or after 20 advection cycles. The results from the nonconservative model show a significant deviation from the exact solution of the problem, while the GLM curl cleaning approach yields very stable and accurate results: on the coarser mesh, some numerical diffusion is visible by comparing with the exact solution, but on the finer mesh numerical diffusion can be considered negligible.

variants of the governing equations are stable, at least with increasing mesh refinement. It seems that not much can be done to improve the stability of the computation when dealing with the the weakly hyperbolic model, which in the run with finer mesh blows up even earlier than with the coarse grid, most likely due to the smaller numerical dissipation of the scheme. This can be attributed to the fact that the $\nabla \times \mathbf{b} = 0$ constraint is not enforced in any way and thus curl involution errors are allowed to grow indefinitely. Evidence corroborating this statement can be found in Figure 6.10, where one can see that early blowup of the computations is associated with very large violations of the curl involutive constraint. On the other hand, both of the strongly hyperbolic variants here proposed automatically limit the growth of curl involution errors and in particular, in the long term, it is always true that the curl errors are lower with GLM curl cleaning than they are with the nonconservative Godunov–Powell-type model. One can also see that the higher the cleaning speed c_h is, the smaller the constraint violations are. Moreover, on the fine mesh, the nonconservative Godunov–Powell system, while still generating much larger errors than the augmented GLM model (clearly visible also in the pressure field shown in Figure 6.11), could be solved for the full 20 advection cycles, as opposed to only 13 on the coarse mesh.

Concerning the effects of numerical dissipation, we can see that the curl errors for the GLM curl cleaning simulations on the coarse grid decrease in time with the aid of numerical diffusion, which reduces the overall steepness of the interface field. This effect can be easily quantified by inspecting Figure 6.11 where it is apparent that with the coarse mesh the pressure field after thirteen

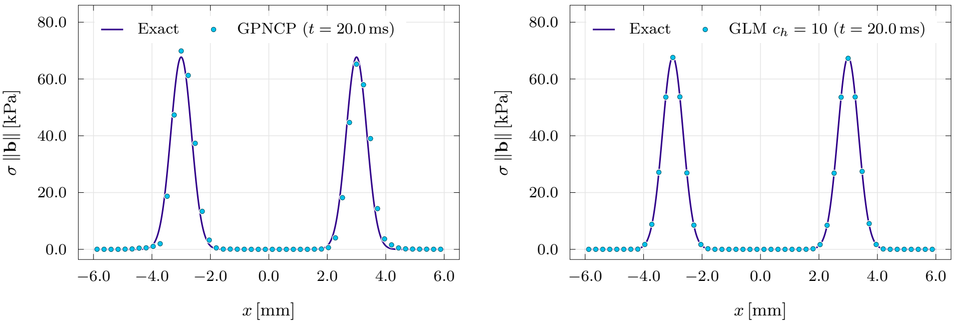


Figure 6.12. One-dimensional cuts (50 uniform samples along the x axis) of the interface energy $\sigma\,\|\mathbf{b}\|$ for the two-dimensional droplet advection test problem. The cuts are taken at time $t = 20.0\,$ms (after 20 advection cycles) The scheme used is ADER-DG $\mathbb{P}_5\mathbb{P}_5$ with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter and the mesh is composed of 32^2 square control volumes.

advection cycles is more diffused than in the initial condition, while this effect is minimised by mesh refinement, as one can clearly see in Figure 6.12, where the profile of the interface field on the GLM simulations is still in perfect agreement with the exact solution after 20 full advection cycles. Regarding this simulation with the finer grid, the curl error timeseries no longer shows the effects of numerical dissipation and in the first stages of the computation (up to about three to four advection cycles) one can see that the curl errors are maintained at a very precise constant value, suggesting that a sort of

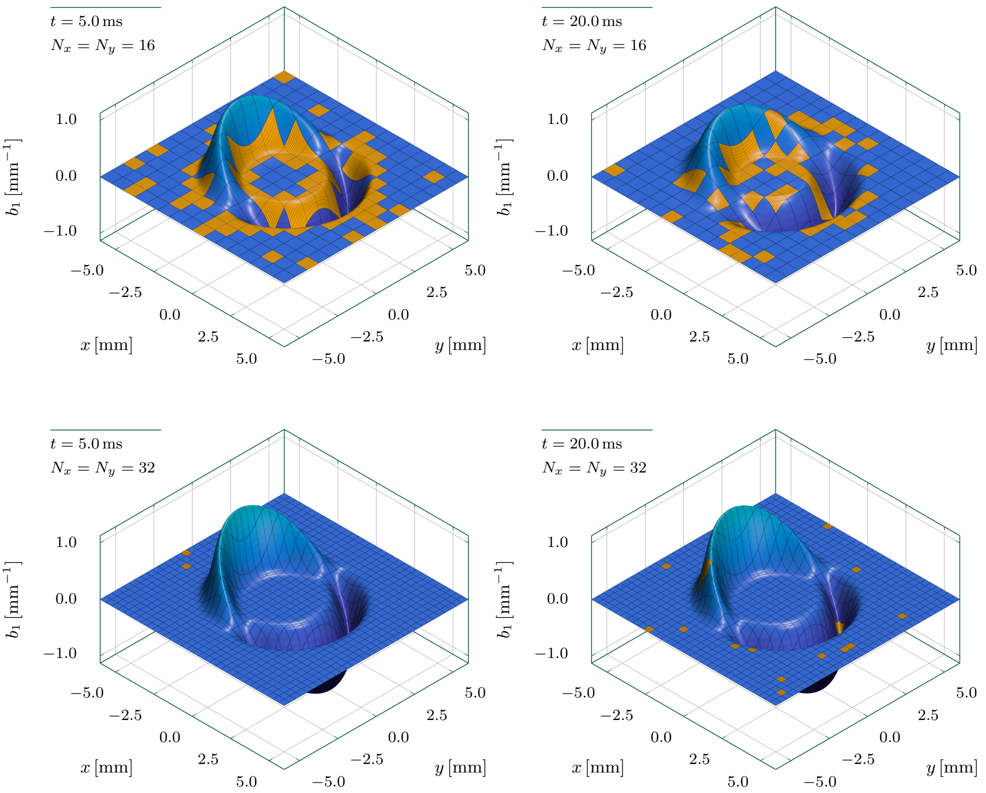


Figure 6.13. Surface plots highlighting subcell limiter activations (in orange colour), relatively to the simulation of a two-dimensional droplet transported in a uniform velocity field. The results are relative to the solution of the GLM curl cleaning model with $c_h = 40\,\mathrm{m\,s^{-1}}$ obtained using a $\mathbb{P}_5\mathbb{P}_5$ ADER-DG scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on two uniform grids composed of 16^2 or 32^2 elements. The plots show the first component of the interface field $\mathbf{b}$, at times $t = 5.0\,\mathrm{ms}$ (five advection cycles, 85 927 timesteps on the finer grid), and $t = 20.0\,\mathrm{ms}$ (twenty advection cycles, 345 207 timesteps on the finer grid).

balance is established between the sources of the curl errors in the numerical scheme and their transport via the Maxwell-type curl cleaning waves of the augmented GLM system. Also, one can note that, for the run with cleaning speed $c_h = 40\,\mathrm{m\,s^{-1}}$, in this early phase, the curl error is kept very close to its non-zero initial value, which is given by the necessity of projecting the inteface field on the piecewise-polynomial Discontinuous Galerkin data representation.

Additional results concerning the effects of mesh resolution on the quality of the solution (namely numerical diffusion) and on activations of the subcell limiting procedure are given in Figure 6.13, where we depict the first component of the interface field **b**, highlighting the cells in which the Finite Volume subcell limiting procedure has been activated. It is clear that the lower resolution simulation ($N_x = N_y = 16$) shows significantly stronger attenuation of the peaks in the interface field, and requires more limiter activations. Opposite to that, using a higher cell count ($N_x = N_y = 32$), the data is recognised by the numerical scheme as well resolved and smooth and, for this reason, limiter activations are very sparse.

In the three-dimensional tests, the effects of numerical diffusion are not seen because the interface profile was chosen to be smoother than the one used for the two-dimensional simulations from the beginning. Otherwise, the

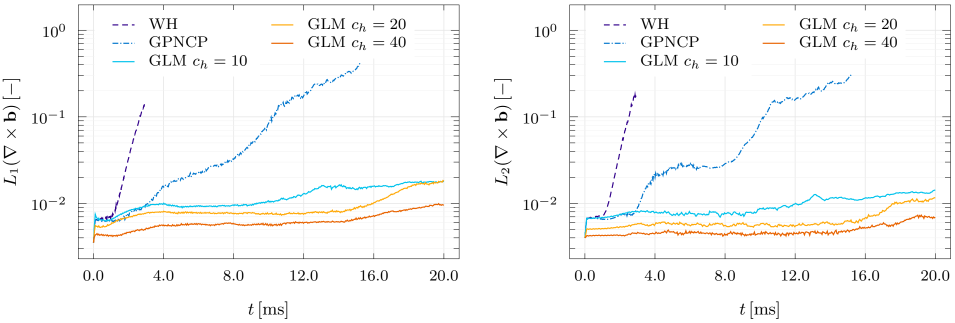


Figure 6.14. Time evolution of the L_1 and L_2 norms of the curl constraint violation for the three-dimensional droplet advection problem. The results obtained for the two-dimensional experiments are confirmed. It is quite apparent that, employing the GLM curl cleaning, the constraint violation grows much slower than with the Godunov–Powell-type formulation of the system (GPNCP). In particular, in this latter case the computation fails after about 15 advection cycles (15.0 ms), while the augmented GLM curl cleaning system shows much better stability properties. The simulations have been carried out using a $\mathbb{P}_3\mathbb{P}_3$ ADER-DG scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on a uniform grid composed of 16^3 elements.

same observations given for the two-dimensional case are valid, namely one can construct a hierarchy of the simulations based on the entity of the curl-constraint violations, that sees the weakly hyperbolic model break down very early, the Godunov–Powell-type symmetrisable model being more stable, but

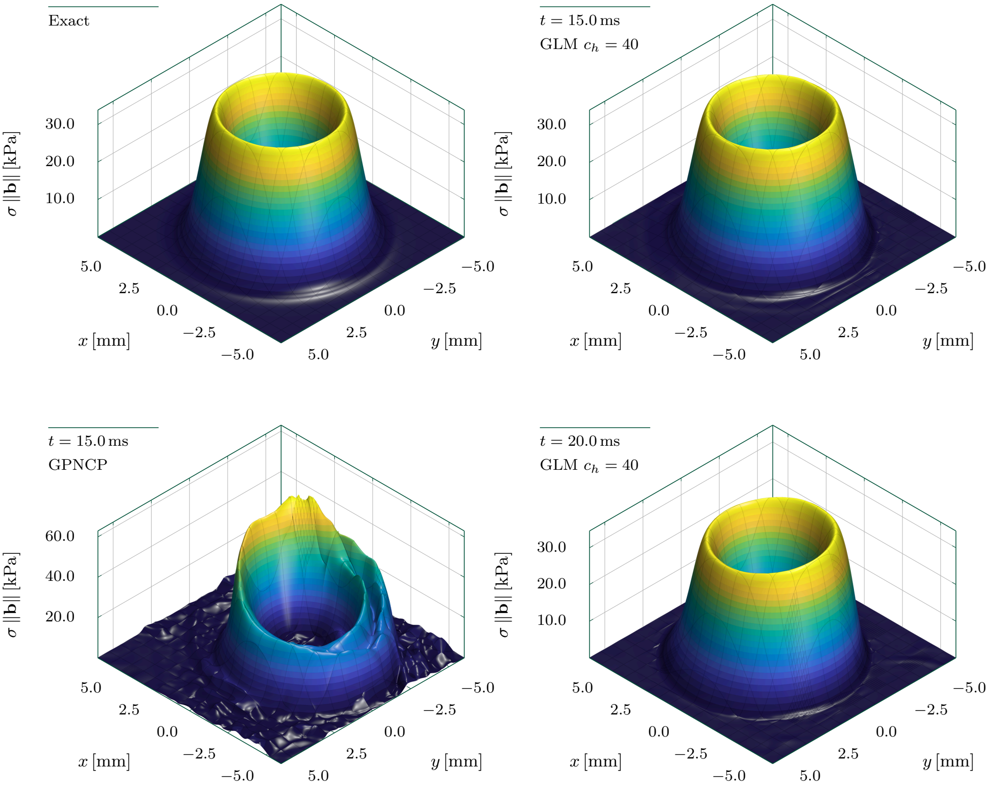


Figure 6.15. Two-dimensional slices, at $z = 0$, of the solution for the surface energy $\sigma\,\|\mathbf{b}\|$, for the three-dimensional droplet advection test. The top left panel shows the exact solution of the problem. The top right plot is the numerical solution at time $t = 15.0\,\mathrm{ms}$, that is, after fifteen advection cycles, obtained using the GLM curl cleaning formulation of the model with cleaning speed $c_h = 40$, showing good agreement with the exact solution. The bottom left plot represents the solution at $t = 15.0\,\mathrm{ms}$ obtained with the Godunov–Powell nonconservative formulation of the model (GPNCP); strong artefacts are visible. The bottom right plot shows the GLM curl cleaning solution after five additional advection cycles, with comparatively minor deformation of the interface. The results are obtained with a fourth order ADER-DG $\mathbb{P}_3\mathbb{P}_3$ and ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter, on a very coarse mesh of 16^3 elements.

more sensitive in the long term than the GLM cleaning simulations, which in turn have lower errors for higher cleaning speeds. The timeseries of the constraint violations are plotted in Figure 6.14, where the error is kept essentially equal to the initial value with the GLM curl cleaning, while it grows rather quickly for the Godunov–Powell formulation, for which the computation stops after completing 15 advection cycles.

In Figure 6.15, we show a set of two-dimensional slices of the solution for the interface energy and we observe that, as for the analogous two-dimensional test,

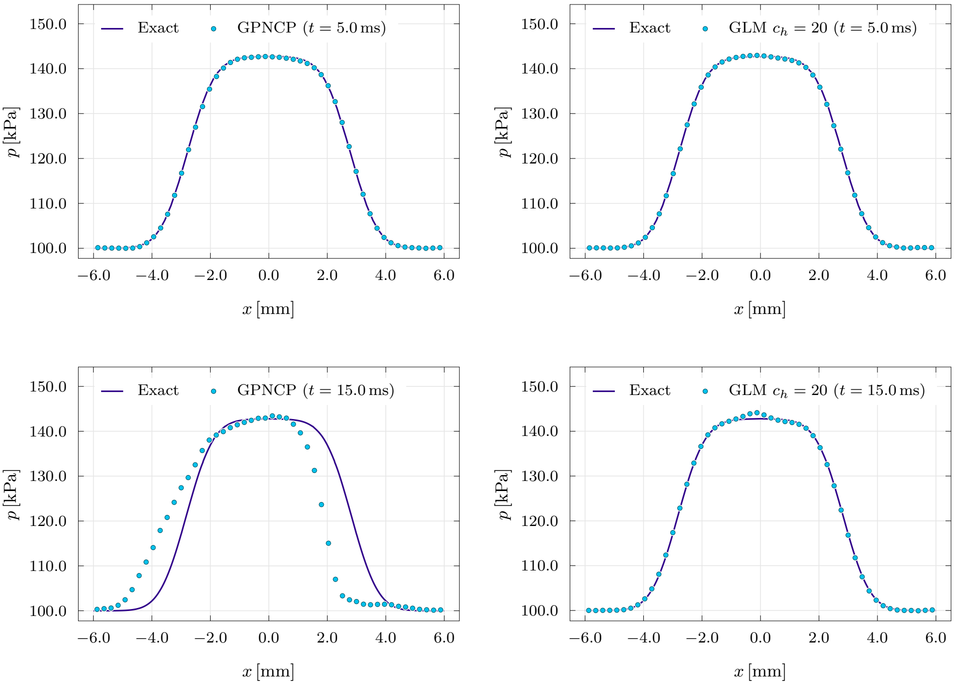


Figure 6.16. One-dimensional cuts (50 uniform samples along the x axis), at $z = y = 0$, of the solution for the pressure field p, for the three-dimensional droplet advection problem. On the left, the solution obtained with the Godunov–Powell nonconservative formulation of the model (GPNCP). On the right, the solution from the GLM curl cleaning formulation, with cleaning speed $c_h = 20$. One can note that at time $t = 15.0\,\mathrm{ms}$, after fifteen full advection cycles, the nonconservative formulation significantly deviates from the exact solution derived in Section 2.2.6, while the error is very well contained by the GLM curl cleaning. At time $t = 5.0\,\mathrm{ms}$, the models yield comparable results. The solutions are obtained with a fourth order ADER-DG $\mathbb{P}_3\mathbb{P}_3$ and ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter, on a very coarse mesh of 16^3 elements.

the hyperbolic Godunov–Powell model shows severe degradation of the interface field after fifteen advection cycles and the droplet is even shifted out of centre, as was the two-dimensional droplet in the second panel of Figure 6.11. At the same time instant, the GLM curl cleaning formulation seems to adequately match the exact solution, despite using a rather coarse mesh, and shows no spurious shift of the centre of mass of the droplet, as seen also in the one-dimensional cuts of Figure 6.16.

6.2.4 Oscillation of an elliptical water column

We continue our systematic comparison of the different formulations of the hyperbolic surface tension model under investigation with a test involving the oscillation of an elliptical water column, which, due to the elongated initial shape, is not in mechanical equilibrium and tends to deform towards restoring a circular shape. The phenomenon is of periodic nature since when the droplet has indeed reached a circular shape, it also stores an amount of kinetic energy such that it starts to elongate again perpendicularly with respect to the previous major axis, up to a maximum deformation, then deforming back to a circular shape and finally to the initial configuration.

6.2.4.1 Problem setup

For the description of the geometry of a smoothed elliptical water column having a nominal interface defined by the parametric equation

$$\mathbf{r}_{\mathrm{b}} = (x_{\mathrm{b}},\ y_{\mathrm{b}}) = (R_x\,\cos\psi,\ R_y\,\sin\psi) \tag{6.12}$$

we introduce the following coordinates: for each point $(x,\ y)$ in the Cartesian plane the local eccentric anomaly ψ is defined implicitly by the formulas

$$\cos^2\psi(x,\ y) = \frac{R_y^2\,x^2}{R_y^2\,x^2 + R_x^2\,y^2}, \qquad \sin^2\psi(x,\ y) = \frac{R_x^2\,y^2}{R_y^2\,x^2 + R_x^2\,y^2}; \tag{6.13}$$

for each point $(x,\ y)$, we can then define the nominal radius of the ellipse in the direction of the local eccentric anomaly

$$R_\psi(\psi(x,\ y)) = \sqrt{R_x^2\,\cos^2\psi + R_y^2\,\sin^2\psi}, \tag{6.14}$$

which would be the length of the segment running from the centre of the water column (located at the origin of the reference system) to the intersection between the ellipse boundary and the line connecting said generic point with the origin. Then we denote as usual with $r(x,\ y) = \sqrt{x^2 + y^2}$ the distance of a generic point from the centre of the water column. Then, given its dimensionless form r_* with respect to R_ψ, the colour function c, its gradient $\mathbf{b} = [b_1,\ b_2,\ 0]$ and the liquid volume fraction α_1 are given as

$$c(x,\ y) = \frac{1}{2}\,\mathrm{erfc}\left(\frac{r - R_\psi}{\varepsilon}\right), \tag{6.15}$$

$$\alpha_1(x,\ y) = \alpha_{\mathrm{min}} + (\alpha_{\mathrm{max}} - \alpha_{\mathrm{min}})\, c(x,\ y), \tag{6.16}$$

$$b_1(x,\ y) = \frac{x}{\sqrt{\pi}\,\varepsilon\, r}\ \exp\left[-\left(\frac{r - R_\psi}{\varepsilon}\right)^2\right]\left[\left(1 - \frac{R_y^2}{R_x^2}\right)\frac{R_\psi}{r}\ \sin^2\psi - 1\right], \tag{6.17}$$

$$b_2(x,\ y) = \frac{y}{\sqrt{\pi}\,\varepsilon\, r}\ \exp\left[-\left(\frac{r - R_\psi}{\varepsilon}\right)^2\right]\left[\left(1 - \frac{R_x^2}{R_y^2}\right)\frac{R_\psi}{r}\ \cos^2\psi - 1\right], \tag{6.18}$$

while the pressure field is initialised as a local application of the solution for a cylindrical water column in the form

$$p\,(r_*) = p_{\mathrm{atm}} + (d - 1)\,\frac{\sigma}{R}\int_{r_*}^{\infty}\frac{1}{\sqrt{\pi}\,k_\varepsilon\, r_*'}\ \exp\left[-\left(\frac{r_*' - 1}{k_\varepsilon}\right)^2\right]\,\mathrm{d}r_*', \tag{6.19}$$

with

$$r_*(x,\ y) = \frac{r(x,\ y)}{R_\psi(\psi(x,\ y))}, \quad k_\varepsilon = \frac{\varepsilon}{R_\kappa}, \tag{6.20}$$

where by local we mean that an average curvature radius R_κ, defined at each point $(x,\ y)$ inside, on the nominal boundary, or outside of the droplet, is computed by averaging the curvature along the nominal boundary of the ellipse, with a weight function inversely proportional to the square of the distance from the interface $[R_\psi(\psi) - r(x,\ y)]^2$, so that we have

$$R_\kappa(x,\ y) = \left\{\frac{\displaystyle\int_0^{2\,\pi}\frac{1}{[R_\psi(\psi) - r(x,\ y)]^2}\,\frac{R_x\,R_y}{R_x^2\,\sin^2\psi + R_y^2\,\cos^2\psi}\,\mathrm{d}\psi}{\displaystyle\int_0^{2\,\pi}\frac{1}{[R_\psi(\psi) - r(x,\ y)]^2}\,\sqrt{R_x^2\,\sin^2\psi + R_y^2\,\cos^2\psi}\,\mathrm{d}\psi}\right\}^{-1}. \tag{6.21}$$

This averaging procedure yields a local curvature radius such that the initial pressure configuration is similar to the one occurring at oscillation extrema, *i.e.* at the end of every half-period, when the kinetic energy of the droplet is zero, as it is set initially. Even if based only on geometrical considerations, this initial condition is sufficient for individuating very clearly only the main oscillation mode of the droplet, allowing to obtain a clean estimate of the oscillation period. A comparison between the geometrically-derived initial pressure field and the configuration after three oscillation periods is shown in

Figure 6.17, together with the complex flow features that are generated in the earliest instants of the simulation.

The density fields are set to the uniform values ρ_1^0 and ρ_2^0 throughout the computational domain, as is the velocity field for which we set $\mathbf{u} = (0,\ 0,\ 0)^\mathsf{T}$. The numerical values employed for this test problem are: $\rho_1^0 = 1000\,\mathrm{kg\,m^{-3}}$, $\rho_2^0 = 1\,\mathrm{kg\,m^{-3}}$, $p_{\mathrm{atm}} = 100\,\mathrm{kPa}$, $R_x = 3\,\mathrm{mm}$, $R_y = 2\,\mathrm{mm}$, $\alpha_{\min} = 0.01$, $\alpha_{\max} = 0.99$, $\sigma = 60\,\mathrm{N\,m^{-1}}$. The parameters for the stiffened gas equation of state are: $\Pi_1 = 1\,\mathrm{MPa}$, $\Pi_2 = 0$, $\gamma_1 = 4$, $\gamma_2 = 1.4$. The domain is the square $\Omega = [-6\,\mathrm{mm},\ 6\,\mathrm{mm}] \times [-6\,\mathrm{mm},\ 6\,\mathrm{mm}]$ and additionally, the initial condition is rotated counter-clockwise by 30 degrees, in order to avoid mesh alignment. In a first batch of tests, we set $\varepsilon = 0.5\,\mathrm{mm}$ and discretise the computational domain with 64^2 square cells, then solving with an ADER-DG $\mathbb{P}_5\mathbb{P}_5$ scheme, supplemented with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ *a posteriori* Finite Volume limiter on the subgrid cells and HLL flux. These simulations are intended to test the capability of the proposed models in a dynamical setting where the interface deforms significantly under the effect of strong surface tension, and verify that the GLM curl cleaning approach can deal with the violations of involution constraints that such deformations generate.

In a second run, in order to study the sensitivity of results and in particular of the oscillation period to the thickness of the diffuse interface region, we set $\varepsilon = 0.25\,\mathrm{mm}$, impose no initial rotation of the droplet, thus aligning the two axis of the ellipse with the reference frame, and compute the solution of the problem on a uniform grid of 50^2 elements, with an eight order ADER-DG $\mathbb{P}_7\mathbb{P}_7$ scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ Finite Volume subcell limiter, and Rusanov flux.

6.2.4.2 Discussion of the results

In this test, the Discontinuous Galerkin scheme is supplemented by a third order ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ Finite Volume subcell limiter, which is an important ingredient for obtaining accurate results in this test and for preserving the very complex smooth structures that arise in the flow (see Figure 6.17); in fact, even with the the extremely large value we adopted for σ, the timescales associated with the theoretical oscillation period T_p^a, given for small amplitude oscillations by the formula [146, 307]

$$T_\mathrm{p}^\mathrm{a} = 2\,\pi \left[\sqrt{\frac{6\,\sigma}{(\rho_1+\rho_2)\,R^3}}\right]^{-1}, \quad \text{with} \quad R = \frac{R_x+R_y}{2}, \tag{6.22}$$

are much larger than the timestep restriction for the numerical method in use and thus the task can be regarded as a long-time integration problem.

Specifically, to test the robustness of the different formulations of the governing equations we evolved an oscillating elliptical droplet up to a final time $t_{\text{end}} = 20.0\,\text{ms}$, which correspond to 605 914 timesteps for the GLM curl cleaning simulation with $c_h = 10$, while the simulation with $c_h = 120$ required 649 578 timesteps, and the Godunov–Powell simulation reaches the final time in 637 368 timesteps. Note that the number of timesteps employed, and thus the associated numerical dissipation, essentially independent of c_h in these runs, due to the fact that the maximum wavespeed for the problem is the Wood sound speed which is slightly smaller than the largest chosen value of $c_h = 120\,\text{m}\,\text{s}^{-1}$. During the run, the total kinetic energy E_{k} (simply computed by integrating $\rho\, e_{\text{k}} = \rho\, \left\| \mathbf{u} \right\|^2 / 2$ over the full domain) is tracked and subsequently employed to measure the oscillation period of the droplet. Together with the kinetic energy, also the L_1 and L_2 norms of the curl errors and of the gradient of the interface field $\nabla \mathbf{b}$ are computed and stored. The norms of $\nabla \mathbf{b}$ can be taken as an indicator of

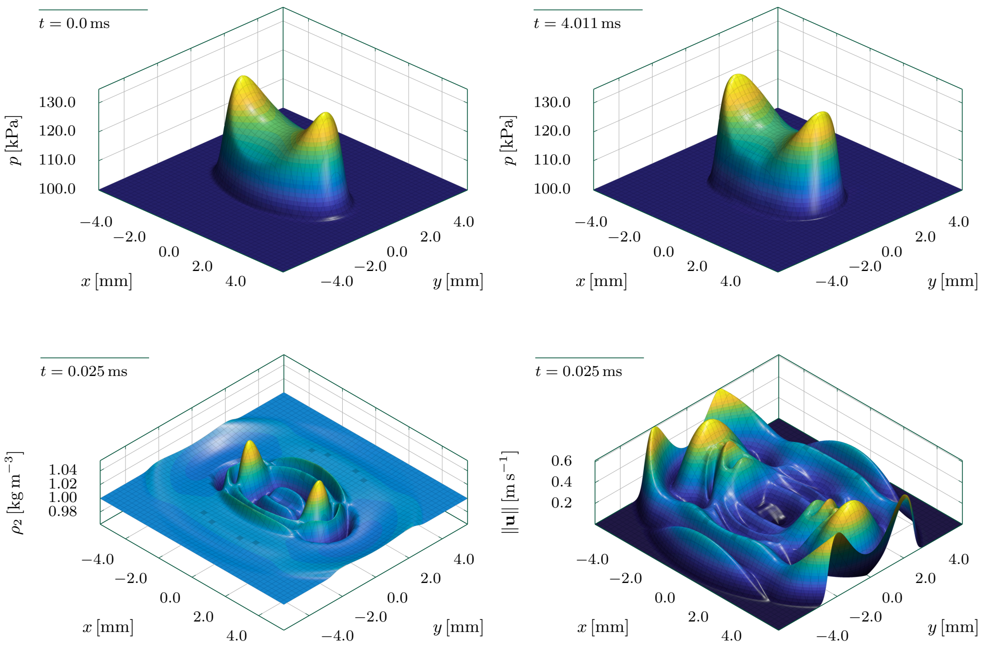


Figure 6.17. Initial stages of the elliptical droplet oscillation problem. In the top row, oscillation periods ($t = 4.011\,\text{ms}$). In the bottom row, density of the gas phase (on the left) and flow speed distribution (on the right) at time $t = 0.025\,\text{ms}$. The Godunov–Powell nonconservative formulation of the model was solved with a $\mathbb{P}_7\mathbb{P}_7$ ADER-DG scheme and ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on a uniform grid of 50^2 elements.

the roughness of the solution, which, with reference to Figure 6.18, can in turn indicate that the solution is developing spurious artefacts instead of maintaining its interface field smooth; alternatively, since the interface field is sensitive to numerical diffusion, seeing that $\nabla \mathbf{b}$ does not quickly decay, indicates that the

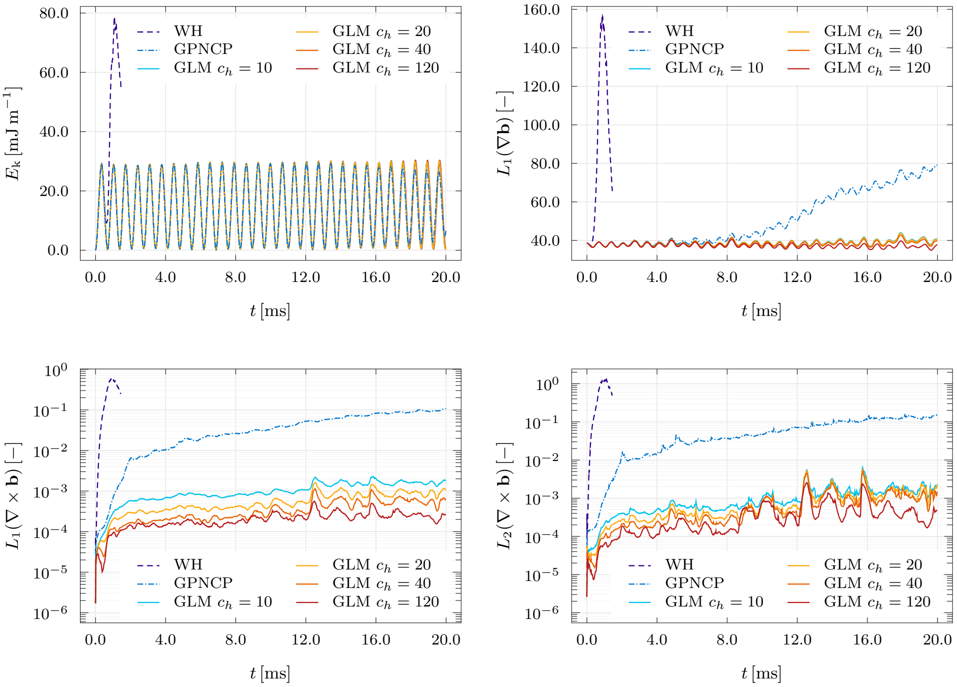


Figure 6.18. Time evolution of global dynamics and of curl errors for an oscillating elliptical water column. In the top row, left to right: the total kinetic energy E_k and the L_1 norm of $\nabla \mathbf{b}$ over time. In the bottom row, the timeseries for the L_1 and L_2 norms of the curl constraint violation error. One can see that all the GLM curl cleaning simulation yield a constant oscillation period and kinetic energy is preserved quite well over more than 6×10^5 timestep, while the Godunov–Powell nonconservative formulation of the governing equations (GPNCP) shows signs of deterioration of the kinetic energy oscillation envelope. Moreover, curl errors decrease for increasing cleaning speed c_h, and in general GLM curl cleaning is effective in containing them, compared to the Godunov–Powell nonconservative formulation, and especially compared to the original weakly hyperbolic (WH) system, for which the computation fails rather early. The simulations employed a $\mathbb{P}_5\mathbb{P}_5$ ADER-DG scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on a uniform grid composed of 64^2 elements.

scheme is not introducing excessive artificial dissipation into the system. The timeseries for these integral quantities are shown in Figure 6.18, and again it appears that the GLM curl cleaning approach yields the best results: the Godunov–Powell simulation show some signs of deterioration of the kinetic energy oscillation envelope and an increase in the average magnitude of the gradients of the interface field. On the other side, the timeseries of kinetic energy obtained with GLM curl cleaning does not show any signs of decay in the solution quality, and this is reflected in the fact that violations of curl involutions are significantly lower with respect to the Godunov–Powell run. Moreover, after fifteen oscillation periods, we cannot observe any decay in kinetic energy due to numerical diffusion, which one would expect from lower order explicit methods for compressible flow. Finally, it can be confirmed that the weakly hyperbolic formulation of the equations is not well suited for solving time-dependent problems with high order Godunov-type schemes, without accounting for curl involutions, as our computations, employing a sixth order ADER-DG method on 64^2 cells, blew up before a single full oscillation period could be simulated.

In Figure 6.19, we see that the simulation reproduces the expected global dynamic behaviour in that the droplet can be observed achieving a circular shape at a quarter and at three quarters of the oscillation period, when the maxima of kinetic energy are reached, while the minima of kinetic energy, defining the half-period and end-period times, correspond to the maximum elongation of the droplet in orthogonal directions. In these early stages of the simulation, we can observe very little difference between the results from the GLM curl cleaning simulations with low cleaning speed ($c_h = 10\,\mathrm{m\,s^{-1}}$) or high cleaning speed ($c_h = 120\,\mathrm{m\,s^{-1}}$), as well as with respect to the solutions of the Godunov–Powell nonconservative variant of the model. On the contrary, since curl involutions are severely violated, it is clear in the first column of Figure 6.19, that the original weakly hyperbolic model does instead develop spurious filaments in the interface field starting from the first oscillation period, leading to a very early breakdown of the computation.

In Figure 6.20, we compare the solutions, at two different time instants, obtained from the Godunov–Powell-type formulation of the model (2.25) and from the augmented GLM curl cleaning system (2.43), with several different values of cleaning speed c_h. At both time instants, we can see only small differences among the simulations using GLM curl cleaning, while it is clear in the first column of Figure 6.19 that with the nonconservative Godunov–Powell-type model some secondary subdivisions of the interface field are starting to develop at time $t = 10.68\,\mathrm{ms}$, three quarters through the eighth oscillation cycle. The effects on the pressure field are not yet visible at this stage, but one can

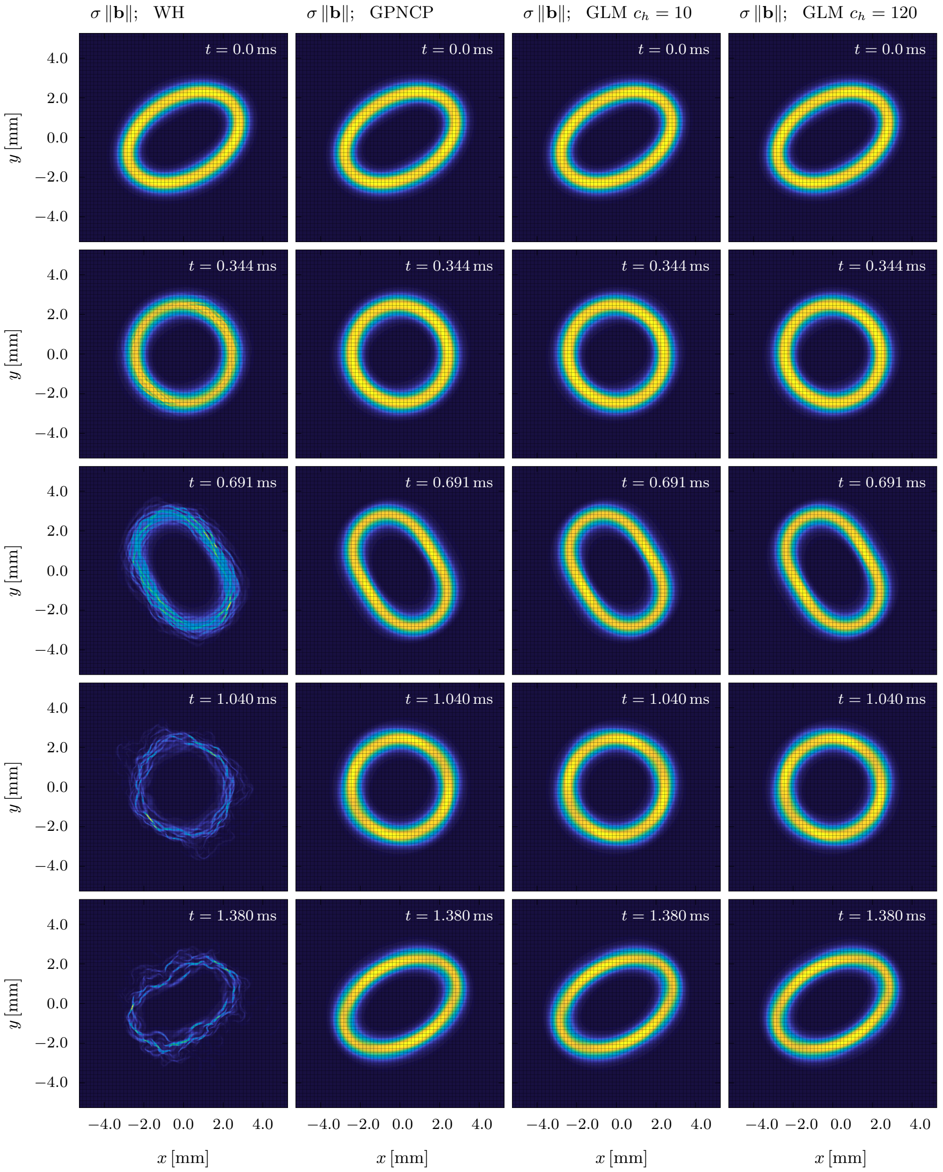


Figure 6.19. Early stages of the oscillation of a two-dimensional elliptical droplet. The first column shows the quick deterioration of the interface field that is observed when solving the weakly hyperbolic formulation of the model (WH), the second shows that restoring hyperbolicity with the Godunov–Powell-type nonconservative products (GPNCP) prevents such ill behaviour, and the same is true for the GLM curl cleaning approach with different cleaning speeds c_h. The simulations employed a $\mathbb{P}_5\mathbb{P}_5$ ADER-DG scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter.

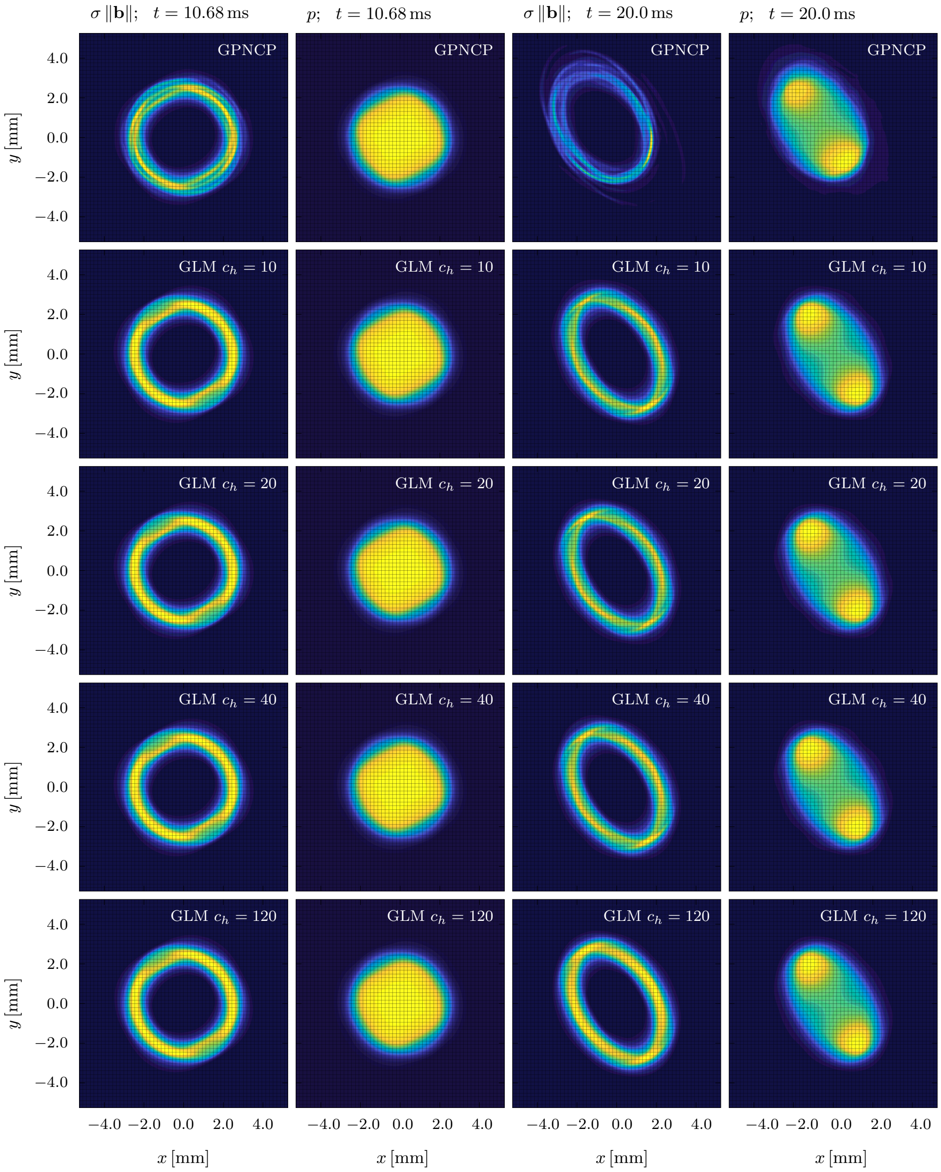


Figure 6.20. Late stages of the oscillation of a two-dimensional elliptical droplet. In the first two columns we plot the surface energy $\sigma\ \|\mathbf{b}\|$ and the pressure p three quarters through the eighth oscillation, and in the last two at about half of the fifteenth oscillation. The first row shows the results of the Godunov–Powell model, with signs of deterioration of the solution, while in the other rows one can see the consistent results of GLM curl cleaning with several cleaning speeds c_h. The simulations employed a $\mathbb{P}_5\mathbb{P}_5$ ADER-DG scheme with ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter.

see in the last two columns of Figure 6.20 that at the end time $t = 20\,\mathrm{ms}$, about half of the fifteenth oscillation cycle, these artefacts have further developed and have caused not only a visible distortion of the pressure field, but also determined a bulk shift of the full droplet, which is no longer centred at the origin of the computational domain, as already seen in the three-dimensional droplet in Figures 6.15 and Figure 6.16.

Our numerical estimate of the oscillation period is obtained by solving the nonconservative Godunov–Powell-type formulation of the model with an ADER-DG $\mathbb{P}_7\mathbb{P}_7$ scheme and ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on a mesh of 50^2 cells; the interface thickness is set to $\varepsilon = 0.25\,\mathrm{mm}$. The deviation of the numerical estimate $T_\mathrm{p} = 1.337\,\mathrm{ms}$ with respect to the analytical prediction of $T_\mathrm{p}^\mathrm{a} = 1.3097\,\mathrm{ms}$ is of 2.1%. While not exact, the result can be considered satisfactory, since the difference can be attributed in part to the fact that the reference formula (6.22) was obtained under the assumption of small amplitude oscillations, and is thus valid only for almost circular droplets. Another potential source of deviation from the predictions of linear theory is the diffuse interface representation of the droplet, which distributes mass differently with respect to the ideal sharp interface jump. In this regard, we computed the oscillation period also from another set of simulations, namely those comparing the long-term behaviour of the Godunov–Powell and of the GLM curl cleaning formulations of the model, employing an ADER-DG $\mathbb{P}_5\mathbb{P}_5$ scheme and ADER-WENO $\mathbb{P}_0\mathbb{P}_2$ subcell limiter on a mesh of 64^2 elements, while the interface thickness is doubled with respect to the previous run from $\varepsilon = 0.25\,\mathrm{mm}$ to $\varepsilon = 0.5\,\mathrm{mm}$. The estimated period for this more diffuse droplet is $T_\mathrm{p}^\mathrm{GPNCP} = 1.375\,\mathrm{ms}$ with the Godunov–Powell-type variant of the model, while we computed $T_\mathrm{p}^\mathrm{GLM} = 1.377\,\mathrm{ms}$ with the GLM curl cleaning formulation. These estimates for the droplet with a thicker interface correspond to a difference of 2.8% to 3.0% with respect to the previous estimate $T_\mathrm{p} = 1.337\,\mathrm{ms}$, and deviate by 5.0% to 5.1% from the small oscillations theory, despite the interface thickness being twice as large as the one used in the previous run.

6.2.5 Tests with variants of the GLM curl cleaning model

Here we present and discuss some of our preliminary results, which justify the exclusion from this work of in-depth analysis for certain variants of the governing equations. In particular we aim to show that:

1. Accounting for the curl constraint $\nabla\times\mathbf{b} = \mathbf{0}$ only in the interface evolution equation (and not in the momentum equation), that is, using the Galilean-invariant equation (2.11f), rather than its non-Galilean-invariant version (2.21), yields similarly unstable results.

2. Conservation errors produced when using the Godunov–Powell-type model (2.25) do not grow extremely quickly but nonetheless they are not automatically guaranteed to be of the order of machine epsilon as they instead are for the conservative GLM curl cleaning augmented system (2.43).

3. Correcting the energy equation of the GLM curl cleaning system (2.43) to the more rigorous nonconservative form (2.44) does not significantly alter the results, but again does not allow to conserve total energy to machine precision.

4. Including the Godunov–Powell symmetrising terms in the GLM augmented system does not alter the results significantly, but again removes formal conservation of momentum and energy.

We repeat the numerical experiments presented is Section 6.2.4, with the same physical parameters and initial data used for the computations shown in Figures 6.18–6.20, but now employing a second order Finite Volume MUSCL–Hancock method and HLL flux (the same used for subcell limiting in Section 6.2.2 and described in Section 3.1.8) on a uniform Cartesian grid of 256^2 elements. This allows to show that the observations given in this Section are valid not only when using high order ADER-DG schemes, but they also hold for standard second order Finite Volume methods.

In a first battery of simulations we compare the conservation errors (in mass, momentum and energy) given by the GLM system (2.43) and by the symmetrised equations (2.25) based on the Godunov–Powell-type approach, labelled GPNCP.

In Figure 6.21, we plot timeseries showing differences in conservation errors, numerical diffusion and constraint violations associated with four variants of the governing equations. The considered variants are the GLM curl cleaning model, the symmetrised Godunov–Powell model, and two variants of the original weakly hyperbolic model, one evolving the interface field with the Galilean-invariant equation (2.11f), here labelled WH (GI), the other using the non-Galilean-invariant version (2.21), labelled WH (CONS), which is the same as using the GLM model with $c_h = 0$. While all variants do not account for the presence of the curl constraint (2.18) at the discrete level, we again see, in the bottom left panel of Figure 6.21, that curl errors are much better contained with GLM curl cleaning than with other methods. Note that the apparently decreasing trend in curl errors of the WH models is associated with an attenuation of the interface field itself, evidenced by the timeseries plot of the L_1 norm of $\nabla \mathbf{b}$ (top right panel of Figure 6.21). The panel shows that

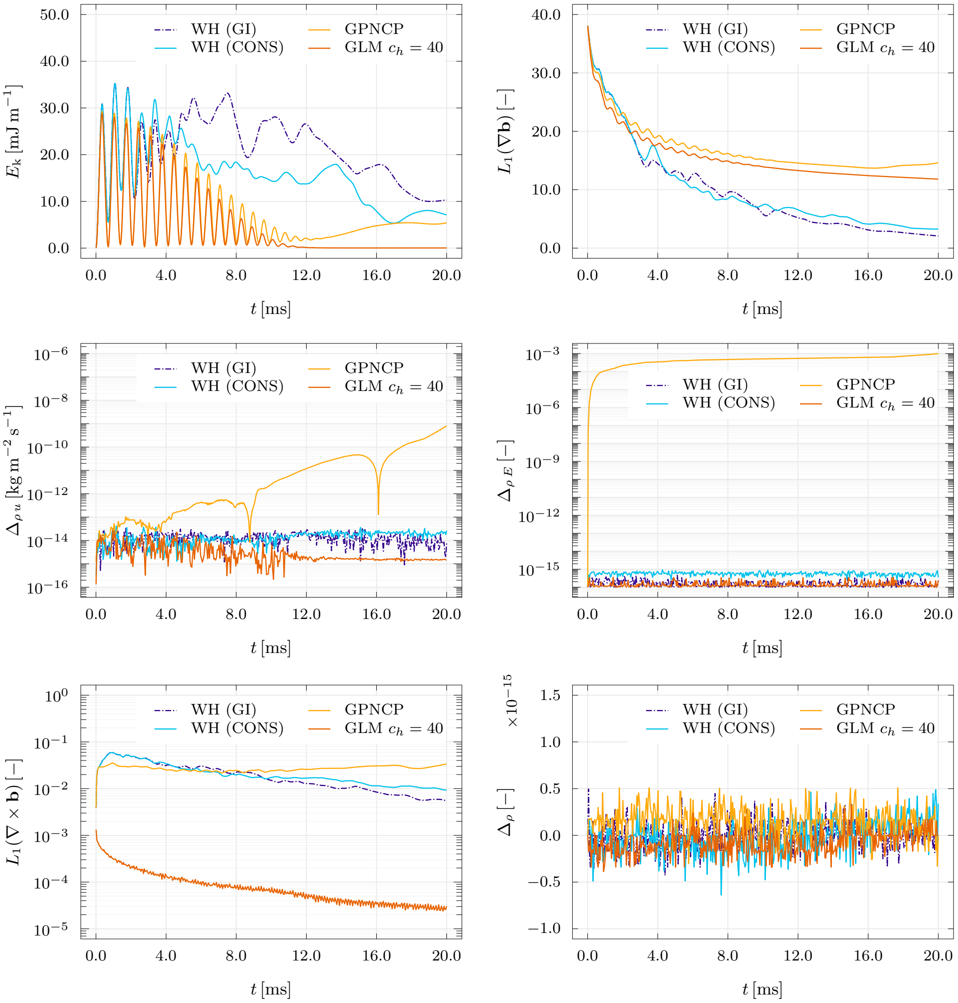


Figure 6.21. Time evolution of integral quantities for a 2D oscillating elliptical droplet. The top panels show the total kinetic energy E_k and the L_1 norm of the gradient of the interface field $\mathbf{b}$; the bottom left panel shows the sum of curl violations $L_1(\nabla \times \mathbf{b})$; in the remaining panels we plot the conservation errors for mass (Δ_ρ), x-momentum ($\Delta_{\rho\, u}$), and energy ($\Delta_{\rho\, E}$). The legend labels indicate: with WH (GI) the weakly hyperbolic model with Galilean-invariant evolution of the interface field; with WH (CONS) the WH model with conservative, but not Galilean-invariant, evolution of the interface field; with GPNCP the Godunov–Powell-type nonconservative model; with GLM $c_h = 40$ the GLM curl cleaning model with cleaning speed $c_h = 40\,\mathrm{m\,s^{-1}}$. The computations have been carried out with a second order MUSCL–Hancock scheme and HLL flux on a uniform mesh of 256^2 cells.

the MUSCL–Hancock second order numerical method, equipped with the very robust but dissipative Minmod slope limiter, introduces a similar amount of numerical diffusion to the GLM and GPNCP models, while the strong spurious oscillations characterising the WH behaviour (top left panel) also trigger much stronger dissipation. Note that, despite the stronger dissipation, the spurious oscillation persist in both WH models, while they are much weaker for the GPNCP model and in particular we observe that the motion simply dies off

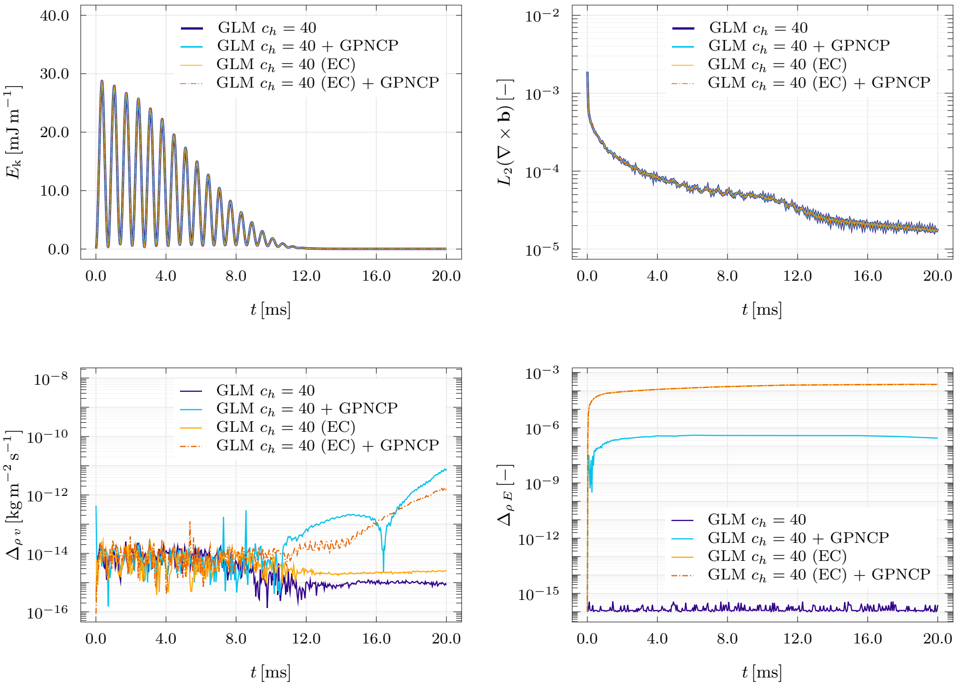


Figure 6.22. Time evolution of integral quantities for a two-dimensional oscillating elliptical droplet. The top panels show the total kinetic energy E_k and the L_2 norm of the curl constraint errors; in the bottom panels we plot the conservation errors for momentum in the y direction ($\Delta_{\rho v}$), and energy ($\Delta_{\rho E}$). With GLM $c_h = 40$ we indicate the results from the GLM curl cleaning model with cleaning speed $c_h = 40\,\mathrm{m\,s^{-1}}$. Additionally, the label GLM + GPNCP refers to the fact that the mathematical model also includes the Godunov–Powell nonconservative products, and with GLM (EC) we denote the fact that the model includes the energy correction term given in Equation (2.44). The computations have been carried out with a second order MUSCL–Hancock path-conservative scheme and HLL numerical flux on a uniform mesh of 256^2 Cartesian cells.

due to numerical diffusion, when using GLM curl cleaning. Note that in the numerical results of Section 6.2.4 (Figure 6.18), because of the very high order of accuracy of the scheme, attenuation of the oscillation amplitude was not observed at all, and thus the considerations presented in this Section up to now are robust to the choice of numerical scheme, at least within the category of explicit Godunov-type methods. With regards to conservation errors, we introduce the quantities

$$\Delta_\rho = \frac{\displaystyle\int_\Omega \rho(\mathbf{x},\ t)}{\displaystyle\int_\Omega \rho(\mathbf{x},\ 0)} - 1, \qquad \Delta_{\rho E} = \left| \frac{\displaystyle\int_\Omega \rho E(\mathbf{x},\ t)}{\displaystyle\int_\Omega \rho E(\mathbf{x},\ 0)} - 1 \right|, \tag{6.23}$$

$$\Delta_{\rho u} = \left| \int_\Omega \rho u(\mathbf{x},\ t) \right|, \qquad \Delta_{\rho v} = \left| \int_\Omega \rho v(\mathbf{x},\ t) \right|,$$

and we track their time evolution during the simulation. In Figure 6.21, we see that both the WH and GLM models conserve the mass, momentum, and energy to machine precision, while the Godunov–Powell-type system does not conserve momentum or energy. The latter observation is expected because the curl constraint is not satisfied exactly and hence, the discrete counterparts of the non-conservative terms in (2.23) and (2.24) are not zeros as they would be at the continuum level. Moreover, we see that errors in momentum (middle left panel of Figure 6.21) are somehow contained even for the GPNCP model: the cusps in the plot of $\Delta_{\rho u}$ correspond to changes of sign of the global momentum conservation error. While six to seven orders of magnitude larger than machine accuracy, momentum errors are still small. On the other hand, energy conservation errors quickly compound to a significant increase in total energy (about one part per thousand). Following this quick increase, growth slows down and stable oscillations of the droplet can be observed, but nonetheless we can say that the non-conservative form of the momentum and energy equations is an issue that should be taken into consideration when using the Godunov–Powell-type symmetrized model.

A second set of simulations is carried out to justify the exclusion of certain variants of the GLM curl cleaning model from being further investigated in this Section. Namely we did not present in detail the model obtained by combining the Godunov–Powell nonconservative products with GLM curl cleaning (GLM + GPNCP), nor the energy-consistent GLM curl cleaning model using the corrected energy balance given in Equation (2.44), here labelled GLM (EC). Also, we did not present in detail the model that combines both nonconservative corrections (GLM (EC) + GPNCP).

The main conclusion that one can draw from the results shown in Figure 6.22 is that the three alternative variants are essentially equivalent to the GLM curl cleaning scheme: the total kinetic energy E_k and curl errors are basically undistinguishable in the four cases considered in this Figure. This is easily justifiable by noting that both the Godunov–Powell nonconservative products and energy correction (2.44) disappear for vanishing $\nabla \times \mathbf{b}$, and in general, they scale precisely as $\nabla \times \mathbf{b}$: this means that, since the GLM part of the model is strongly limiting all violations of the curl constraint, then the nonconservative corrections considered in Figure 6.22 are very small and do not affect the bulk flow parameters. However, formal conservation (up to machine precision) of momentum and energy, is lost. More in detail, we observe that the energy correction (2.44) correctly does not affect momentum conservation, but is more impactful to energy conservation than adding the Godunov–Powell nonconservative products to the GLM curl cleaning formulation: the total deviation $\Delta_{\rho E}$ stabilises around 2×10^{-4} for all variants using the energy correction (2.44), and around 4×10^{-7} for the model combining GLM curl cleaning with Godunov–Powell nonconservative products.

Thus, for the test problems here presented, these variants are equivalent to plain GLM curl cleaning, while at the same time not formally conservative for energy and momentum.

6.3 High order ADER schemes on fixed and moving meshes for solids with and without material failure

In this Section we report computational results obtained with High order ADER methods of Discontinuous Galerkin and Finite Volume type, on fixed Cartesian grids and polygonal unstructured moving meshes. Moreover, we validate our novel exponential-type integrators for the strain relaxation equations of the unified model of continuum mechanics, with applications with diffuse interface crack propagation, material failure, and fatigue. Additionally, we highlight the versatility of the semi-analytical integration technique proposed in this work by showing results obtained from its inclusion in an updated Lagrangian method on unstructured Delaunay and tetrahedral meshes. The results are published in [47, 59, 147, 313].

6.3.1 Circular explosion problem in an elastic solid

In this Section, we simulate a circular explosion problem in an ideal elastic solid. We compare the results obtained with a third order ADER-WENO finite volume scheme on moving unstructured polygonal meshes with changes of connectivity, with those obtained with a fourth order ADER discontinuous Galerkin finite element scheme on a very fine uniform Cartesian mesh composed

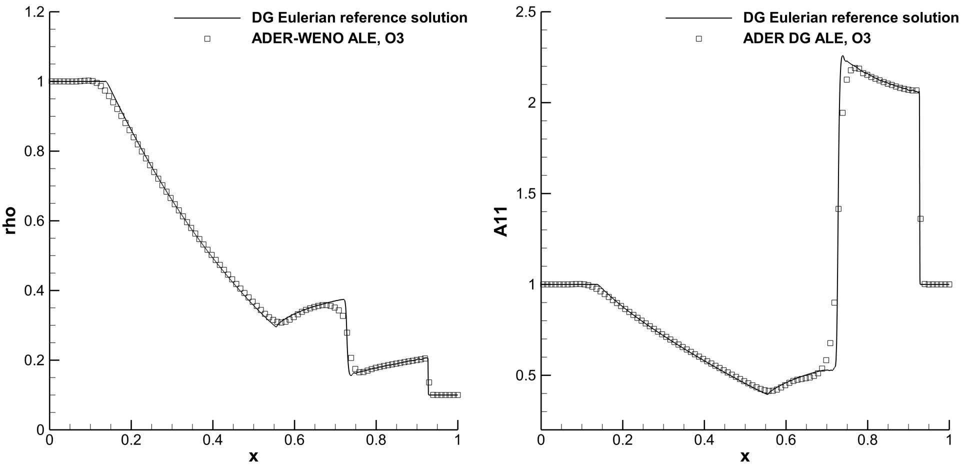


Figure 6.23. Radial cuts along the x-axis for the explosion problem obtained with a third order ADER-WENO ALE finite volume scheme on a moving polygonal grid composed of 82 919 cells and with a fourth order ADER-DG scheme on a Cartesian grid of size $512^2 = 262\,144$ (4.2×10^6 DOFs).

of 512 × 512 elements, which will be taken as the reference solution for this benchmark. The computational domain is $\Omega = [-1,\ 1] \times [-1,\ 1]$ and the final simulation time is $t = 0.25$. We set $\alpha = 1$, $\mathbf{u} = \mathbf{0}$, $\mathbf{A} = \mathbf{I}$ and $\mathbf{J} = \mathbf{0}$ in the entire domain. For $r = \|\mathbf{x}\| \leq 0.5$ the initial density and the initial

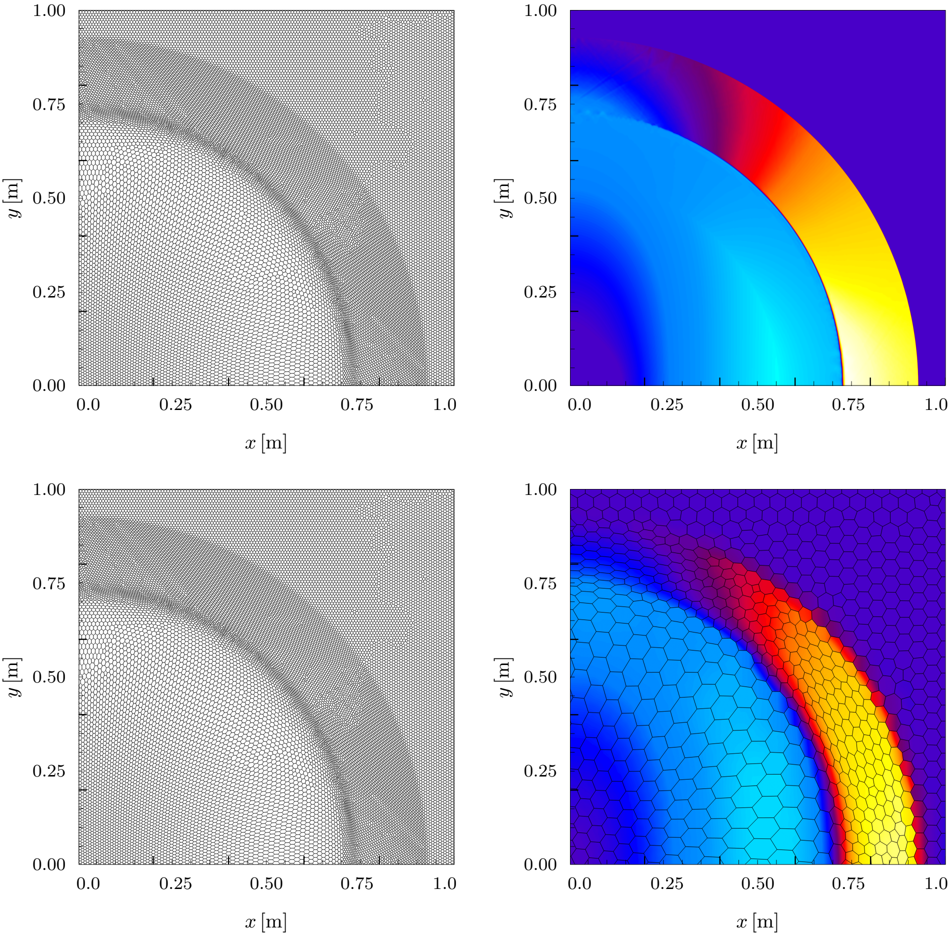


Figure 6.24. Simulation results for the explosion problem obtained with a third order ADER-WENO ALE finite volume scheme on a moving polygonal grid composed of 82 919 cells and with a fourth order ADER-DG scheme on a Cartesian grid of size $512^2 = 262\,144$ (4.2×10^6 DOFs). In the top row, from the left, the solution for A_{11} obtained with the ADER-WENO ALE scheme and with the ADER-DG Eulerian scheme; in the bottom row, the polygonal grid at the final simulation time and the results from the ADER-WENO ALE scheme on a coarser grid of 2 727 elements.

pressure are set to $\rho = 1$ and $p = 1$, while in the rest of the domain we set $\rho = 0.1$ and $p = 10^{-3}$. The parameters of the GPR model are chosen as follows: $c_\mathrm{s} = 0.2$, $\alpha_T = 0$, $\tau \to \infty$ (in order to model an elastic solid). We use the stiffened gas equation of state with $\gamma = 2$ and $\Pi = 0$. For the simulation on the moving polygonal mesh, we employ a mesh with 82 919 control volumes. The computational results obtained with the unstructured ADER-WENO ALE scheme and those obtained with the high order Eulerian ADER-DG scheme are presented and compared with each other in Figures 6.23 and 6.24. We can note a very good agreement between the two results. The high quality of the ADER-WENO finite volume scheme on coarse grids is mainly due to the natural mesh refinement around the shock, which is typical for Lagrangian schemes. Furthermore, Lagrangian schemes are well known to capture material interfaces and contact discontinuities very well, since the mesh is moving with the fluid and thus numerical dissipation at linear degenerate fields moving with the fluid velocity is significantly lower than with classical Eulerian schemes.

6.3.2 Solid rotor test problem

A second solid mechanics benchmark consists in the simulation of a plate on which a rotational impulse is initially impressed, in a circular region centered with respect to the computational domain. This *rotor* will initially move according to the rotational impulse, while emitting elastic waves which ultimately determine the formation of a set of concentric rings with alternating direction of rotation. The test is analogous to the rotor problem shown in [255], but with a weakened material in order to show stronger motion of the polygonal grid.

The results of the third order ADER-WENO finite volume method on a moving polygonal grid with variable connectivity, composed of 150 561 cells, are compared against a reference solution obtained with a fourth order ADER Discontinuos Galerkin scheme on a very fine uniform Cartesian mesh counting 512×512 elements, for a total of over four million spatial degrees of freedom.

The computational domain is the square $\Omega = [-1,\ 1] \times [-1,\ 1]$ and the final simulation time is set to $t = 0.5$. With exception made for the velocity field, all variables are initially constant throughout the domain. Specifically we set $\alpha = 1$, $\rho = 1$, $p = 1$, $\mathbf{A} = \mathbf{I}$, $\mathbf{J} = \mathbf{0}$, while the velocity field is $\mathbf{u} = (-y/R,\ x/R,\ 0)^\mathsf{T}$ if $r = \|\mathbf{x}\| \leq R$, and $\mathbf{u} = \mathbf{0}$ otherwise, that is, outside of the circle of radius $R = 0.2$; this way, the initial tangential velocity at $r = R$ is one. The solid is taken to be elastic ($\tau \to \infty$), heat wave propagation is neglected ($\alpha_T = 0$), and the characteristic speed of shear waves is $c_\mathrm{s} = 0.25$. The constitutive law is chosen to be the stiffened-gas EOS with $\gamma = 1.4$ and $\Pi = 0$. We can see in

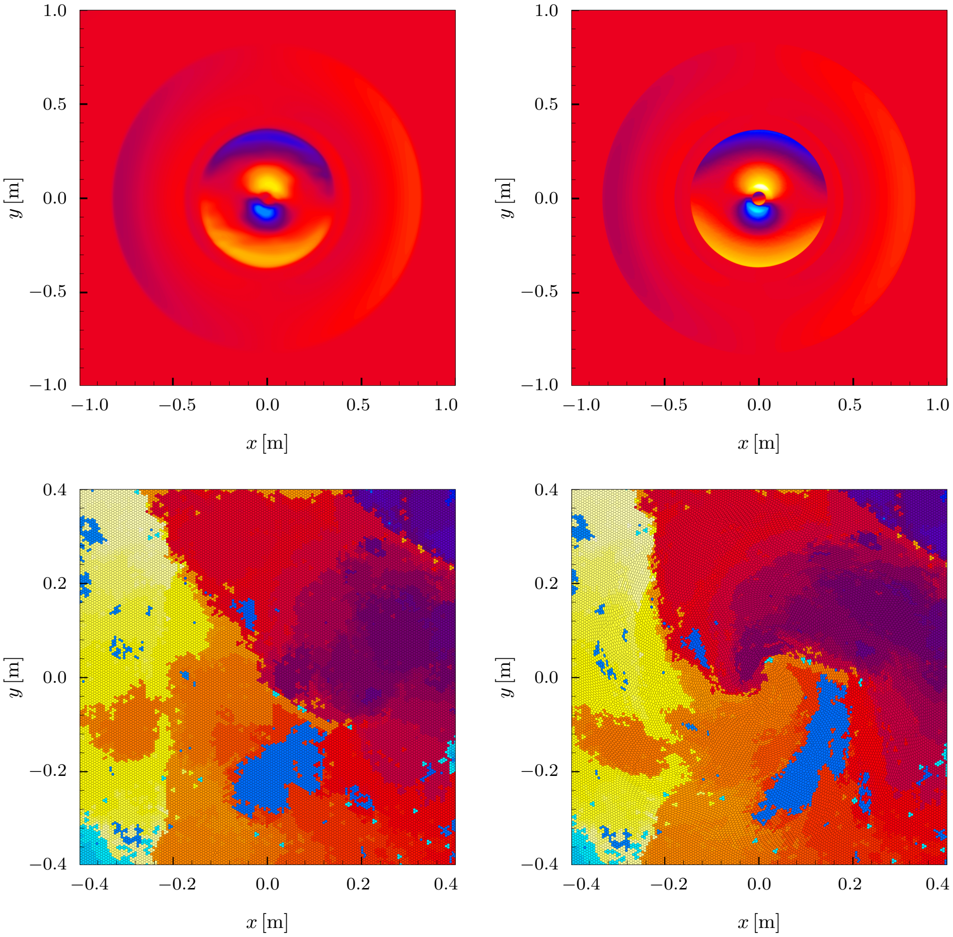


Figure 6.25. Simulation results for the solid rotor problem obtained from a third order ADER-WENO ALE finite volume scheme on a moving polygonal grid composed of 150 561 cells and with a fourth order ADER-DG scheme on a Cartesian grid of size $512^2 = 262\,144$ (4.2×10^6 DOFs). In the top row, the solutions for the u component of the velocity field are shown, on the left those obtained with the unstructured ADER-WENO ALE scheme on moving polygonal meshes and on the right those of the ADER-DG scheme on a fixed Cartesian grid; in the bottom panels the cells are coloured according to their mesh numbering to show the mesh motion between the beginning of the ALE simulation and the final time.

Figure 6.25 that, although some of the finer features are lost (specifically the small central counterclockwise-rotating ring) due to the lower resolution of the finite volume method on a coarser grid, the shear waves travel outwards with the correct velocity and the moving polygonal Finite Volume simulation can be said to be in agreement with the high resolution discontinuous Galerkin results. Also in Figure 6.25, it is shown that the central region of the computational grid has undergone significant motion but thanks to the absence of constraints on the connectivity between elements, the polygonal control volumes have not been stretched excessively as would instead happen for a similar moving unstructured grid, but with fixed connectivity.

6.3.3 Validation of the exponential integrator against LSODA and comparison with the implicit Euler scheme

Here we validate our exponential time integrator against available standard software for the numerical integration of stiff ODE. In particular, we compare against the community standard LSODA/ODEPACK, see [183, 184]. We also compare our exponential time integrator against a simple implicit Euler time integration scheme, showing clearly that implicit Euler time stepping, even with very fine substeps, can lead to significantly inaccurate results.

The benchmark consists in computing the stress-strain diagram for a given choice of spatially homogeneous material parameters: we distort, with constant strain rate $\dot{\boldsymbol{\varepsilon}}$, a homogeneous sample of material, which is initially unstressed ($\mathbf{A} = \mathbf{I}$) and undamaged ($\xi = 0$). The cobasis equation therefore reduces to a simple ODE system. The strain rate tensor (accounting for the left hand side of the PDE system) will have the form

$$\dot{\boldsymbol{\varepsilon}}(t) = \begin{pmatrix} \dot{\varepsilon}_{11}(t) & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \tag{6.24}$$

and the resulting governing ODE system for the dynamics reads

$$\begin{cases} \dfrac{\mathrm{d}\xi}{\mathrm{d}t} = -\vartheta\,\dfrac{\partial E}{\partial \xi}, \\[2ex] \dfrac{\mathrm{d}\mathbf{A}}{\mathrm{d}t} = -\mathbf{A}\,\dot{\boldsymbol{\varepsilon}}(t) - \dfrac{3}{\tau_{\mathrm{m}}}\,(\det\mathbf{A})^{5/3}\,\mathbf{A}\,\operatorname{dev}\mathbf{G}, \end{cases} \tag{6.25}$$

6.3.3.1 Brittle and ductile material behavior

The aim of this test is not only the validation of the new ODE solver with exponential time integration, but also the capability of the GPR model with material failure to describe completely different material behaviour with one and the same set of governing equations, just changing the model parameters appropriately. In particular, the proposed model is able to reproduce brittle material as well as ductile material behaviour.

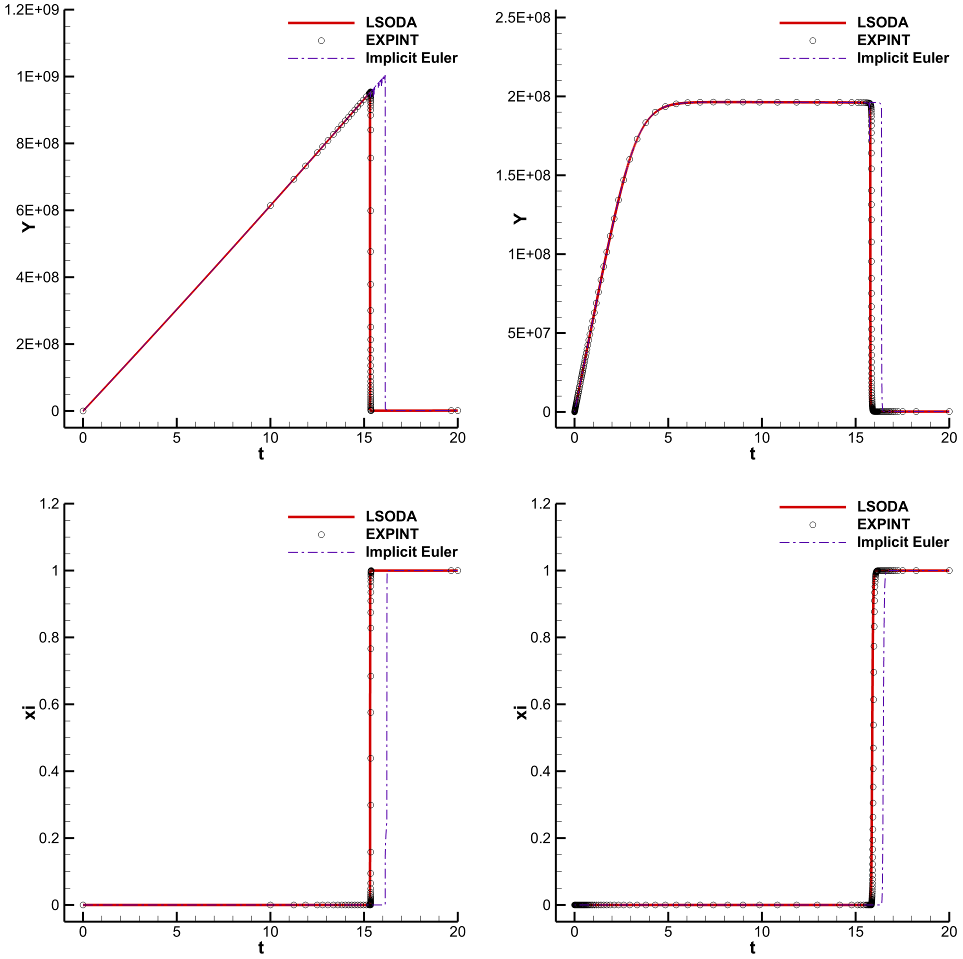


Figure 6.26. Stress-strain diagrams (top row) and time evolution of the damage variable ξ (bottom row) for an example material characterized by brittle failure (left column) and for a material with ductile failure mode (right column). We show that the solution obtained from the proposed exponential integration method matches the reference given by the LSODA library, while integration with the simple implicit Euler method can yield inaccurate solutions even with very small timesteps.

We set the strain rate to $\dot{\varepsilon}_{11}(t) = -0.001\,\mathrm{s}^{-1}$ and compute the solution until a fixed end time $t_{\mathrm{end}} = 20\,\mathrm{s}$, first with the new exponential integrator with adaptive timestepping, and then with the `scipy.integrate.solve_ivp` Python library routine, which contains a wrapper to the popular LSODA/ODEPACK Fortran code [184]. Furthermore, we also show the solution obtained by integrating the governing ODE in 10^4 uniform timesteps with the implicit Euler scheme.

In Figure 6.26 we plot the von Mises stress $Y = \sqrt{3\,\mathrm{tr}\,(\mathrm{dev}\,\boldsymbol{\sigma}\ \mathrm{dev}\,\boldsymbol{\sigma})\,/2}$ against time, together with the time evolution of the damage variable ξ. In a first run (first column of Figure 6.26) the material parameters are chosen so that the material exhibits a characteristically brittle behavior (like ceramics or glass), with linear elastic deformation until the failure point (clearly marked by the jump in the damage variable ξ), while a second run is carried out with ductile material (like for example most metals). In this latter case (second column of Figure 6.26), one can clearly distinguish an initial linear elastic deformation, followed by a nonlinear transition into ideal plastic flow until eventual failure. Our example brittle material is obtained with the following choice of model parameters: $\rho_0 = 3000\,\mathrm{kg\,m}^{-1}$, $\mu_I = 30\,\mathrm{GPa}$, $\mu_D = 30\,\mathrm{MPa}$, $\lambda_I = \lambda_D = 60\,\mathrm{GPa}$, $\tau_{I0} = 3\times10^3\,\mathrm{s}$, $\tau_{D0} = 3\,\mathrm{s}$, $\vartheta_0 = 8$, $a = 32.5$, $Y_0 = 1.4\,\mathrm{GPa}$, $Y_1 = 10\,\mathrm{MPa}$, $\alpha_I = 35$, $\alpha_D = 35$, $\beta_I = 2.2\times10^{-8}\,\mathrm{Pa}^{-1}$, $\beta_D = 2.2\times10^{-7}\,\mathrm{Pa}^{-1}$, while the ductile behavior is given with $\rho_0 = 3000\,\mathrm{kg\,m}^{-1}$, $\mu_I = 30\,\mathrm{GPa}$, $\mu_D = 30\,\mathrm{MPa}$, $\lambda_I = \lambda_D = 60\,\mathrm{GPa}$, $\tau_{I0} = 1\times10^3\,\mathrm{s}$, $\tau_{D0} = 1\,\mathrm{s}$, $\vartheta_0 = 1$, $a = 1$, $Y_0 = 8\,\mathrm{TPa}$, $Y_1 = 4\,\mathrm{MPa}$, $\alpha_I = 0$, $\alpha_D = 30$, $\beta_I = 2\times10^{-8}\,\mathrm{Pa}^{-1}$, $\beta_D = 1\times10^{-4}\,\mathrm{Pa}^{-1}$.

The results from the exponential integrator are in perfect agreement with the LSODA reference solution, while it is apparent that 10^4 timesteps with the implicit Euler method are not sufficient for adequately capturing the sudden onset of material failure.

6.3.3.2 Rate-dependent behaviour and material fatigue

In this test we want to show that our model is a so-called *rate-dependent model*, that is, the maximum stress that can be sustained by the material before the failure point is reached, can vary as a function of the speed of deformation, i.e. the strain rate. It is quite common for example that impacts or explosions allow a given material to achieve a higher equivalent stress than slow (quasi-static) loads, before total breakdown. To show that effects of this type can be reproduced by our model, we repeat the previous test on the ductile material, but by varying the strain rate from $\dot{\varepsilon}_{11}(t) = -0.001\,\mathrm{s}^{-1}$ to $\dot{\varepsilon}_{11}(t) = -0.002\,\mathrm{s}^{-1}$ and $\dot{\varepsilon}_{11}(t) = -0.004\,\mathrm{s}^{-1}$, reaching higher maximum equivalent stress states as strain is applied faster. This is shown in Figure 6.27, where we also report the time evolution of the damage parameter ξ, with logarithmic scaling, in order to highlight that very small deviations of this variable from the perfectly intact state (of the order of 10^{-15} to 10^{-5}) can indicate the difference between the Hookian elastic response and the plastic regime.

Finally, we want to show that the proposed model can also describe *fatigue* effects by allowing the material to retain memory of previous stress states. In fact, thermodynamical consistency prescribes that the source term associated with the damage variable always be positive (or possibly vanishing), this means that any stress applied to the material will ever so slightly damage it and eventually cause a deterioration of its mechanical properties. The extremely nonlinear response to different intensity levels of the stress norm determines that appropriate choices of parameters can describe materials with very late onset of fatigue effects, as well as properly governing the deterioration process following a certain level of exposure to cyclic stress.

For this test we employ an elasto-plastic material characterized by the following choice of parameters: $\rho_0 = 3000\,\mathrm{kg\,m^{-1}}$, $\mu_I = 30\,\mathrm{GPa}$, $\mu_D = 33\,\mathrm{MPa}$, $\lambda_I = \lambda_D = 60\,\mathrm{GPa}$, $\tau_{I0} = 2 \times 10^5\,\mathrm{s}$, $\tau_{D0} = 2 \times 10^3\,\mathrm{s}$, $\vartheta_0 = 1$, $a = 1$, $Y_0 = 8\,\mathrm{TPa}$, $Y_1 = 8\,\mathrm{TPa}$, $\alpha_I = 0$, $\alpha_D = 0$, $\beta_I = 3 \times 10^{-8}\,\mathrm{Pa^{-1}}$, $\beta_D = 0\,\mathrm{Pa^{-1}}$. We setup a series of computations by first subjecting the material to a low intensity cyclic stress (about 20% of the limit for the elastic regime), determined by a variable

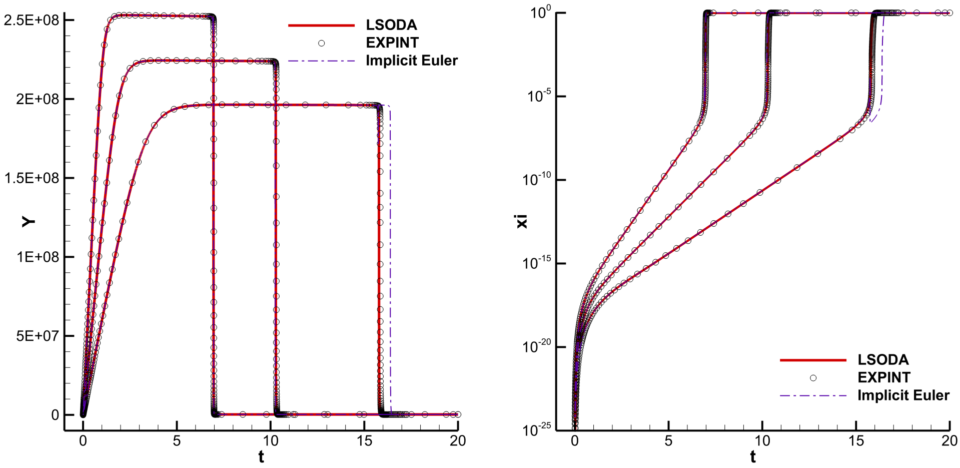


Figure 6.27. Rate-dependent stress-strain diagrams (left panel) and time evolution of the damage variable (right panel) for a ductile material subjected to three different constant strain rates.

strain rate of the form $\dot{\varepsilon}_{11}(t) = -0.001\,\sin(2\,\pi\,t)\,\mathrm{s}^{-1}$, and then measuring the residual strength of the material by means of a quasi-static destructive test

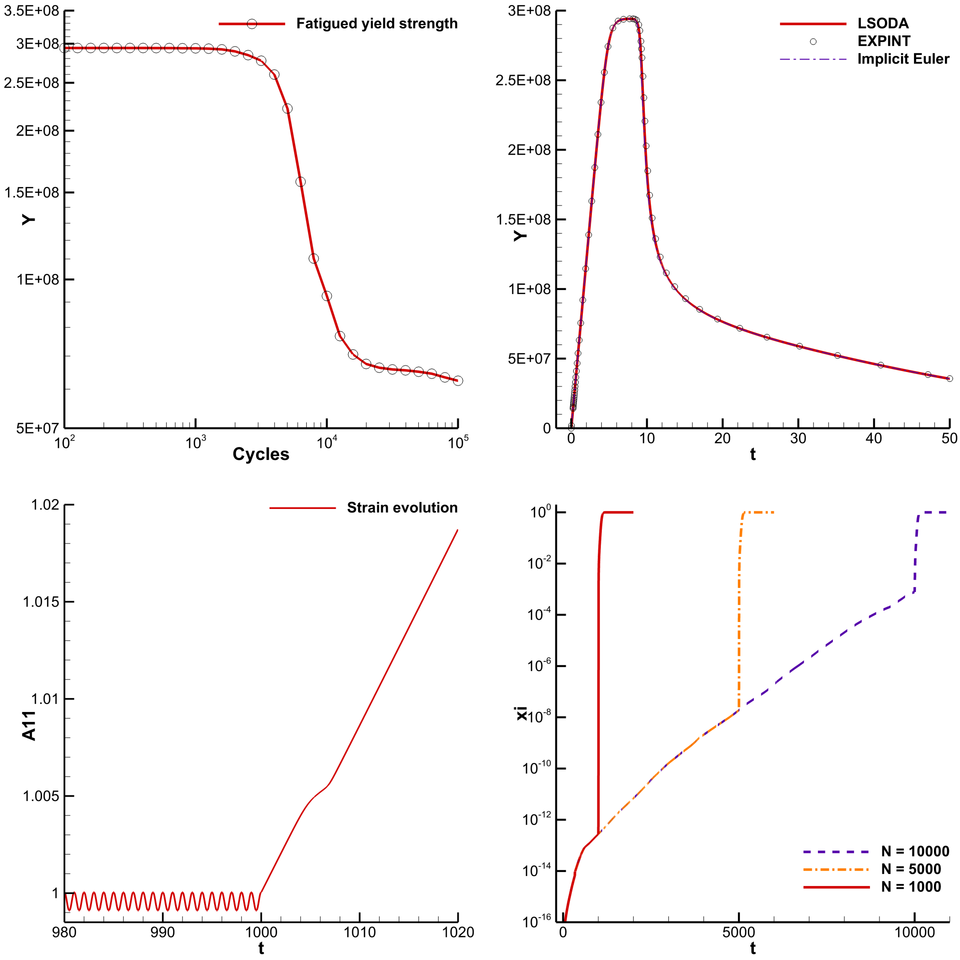


Figure 6.28. Numerical results obtained for the fatigue behaviour. Top left panel: Wöhler diagram, showing the weakening of the material strength as a function of the number of applied load cycles. Top right panel: stress-strain diagram of the test material for a single, quasi-static (slow) load cycle until rupture. Bottom left panel: evolution of the A_{11} component of the distortion field for the last elastic load cycles in the case $N = 1000$ and final traction test until material failure. Bottom right panel: temporal evolution of the damage variable ξ for three different numbers of load cycles N.

as those shown in Figures 6.26 and 6.27. A plot of the material distortion caused by this type of forcing (low intensity periodic deformations followed by a destructive test) is shown in Figure 6.28, together with the numerical stress-strain diagram of an initially intact sample under slow (quasi-static) loading. The main result of this benchmark is the so-called Wöhler diagram: for this purpose, in the top left panel of Figure 6.28 we show a bilogarithmic plot of the material strength as a function of the number of applied cycles. The Wöhler diagram obtained in our simulations shows the same qualitative behaviour as those obtained in experiments for real materials, with the classical division into three different regimes: short, intermediate and long-term durability. Our numerical results illustrate that the test material shows no signs of weakening at a low number of load cycles (up to about 1000). For more load cycles, one can note the typical decay of the material resistance, while for a very large number of cycles the the long-time durability limit is reached, which, however, still shows a small but steady degradation, as it is commonly found in aluminum-alloys. In order to give the reader a more detailed insight about how fatigue is accounted for in our model, we also show the temporal evolution of the damage variable ξ (bottom right panel) for three of the tests (one at $N = 1000$ elastic cycles, one at $N = 5000$ cycles, one at $N = 10000$ cycles), showing that the accumulation of fatigue effects is well captured by the exponential growth of the damage variable. We would like to emphasize that the addition of a thermodynamically compatible evolution equation (2.66f) for a simple scalar ξ to the GPR model [120, 257] is enough to model such a complex behaviour as material fatigue.

6.3.4 Application of the semi-analytical integration scheme to viscous flow with shockwaves in an updated Lagrangian scheme on unstructured meshes

In order to argue the generality of the proposed semi-analytical integration scheme, we report the results of two test cases in which we solve the equations of the GPR model in the Navier–Stokes regime, but within the framework of an *updated Lagrangian* code on unstructured Delaunay grids [47]. The method is here abbreviated LAGR-GPR The method, introduced by Boscheri, Loubère and Maire in [48] for the computation of hyperelastic solids, features the convenient property of evolving in a compatible manner the mass density field ρ and the determinant of the metric tensor $\mathbf{G}$, so that, albeit at the cost of losing the rotational information encoded in the distortion matrix $\mathbf{A}$ and its simple involution structure, the constraint $\det \mathbf{G} = (\rho/\rho_0)^2$ is satisfied exactly by construction. The extension to inviscid and to viscous, heat-conducting fluids simply involved the discretisation of the equations for the thermal impulse **J** and the inclusion of the novel semi-analytical solver for strain relaxation presented in Section 4.5. Second order of accuracy is obtained by means of an IMEX [9, 35–38, 204, 250] time discretization. The results have been published in [47].

6.3.4.1 Sedov blast wave problem

The Sedov problem is concerned with the evolution of a blast wave with cylindrical or spherical symmetry, generated at the lower left corner of the initial computational domain $\Omega = [0,\ 1.2]^d$. Symmetry boundary conditions are imposed on those faces which share the origin, while the remaining sides are treated as slip-walls. The characteristic mesh size is $h = 1/60$ in 3D, whereas we use $h = 1/40$ in 2D. The ideal gas with $\gamma = 1.4$ is initially at rest with an initial pressure of $p = 10^{-6}$ in the entire computational domain, apart in those cells containing the lower left corner as a vertex, where the pressure is prescribed as $p_0 = (\gamma - 1)\,\rho_0\,E_{\text{tot}}\,(\alpha\,V_0)^{-1}$, with $E_{\text{tot}} = 0.244816$ in two space dimensions ($d = 2$), or $E_{\text{tot}} = 0.851072$ in three space dimensions ($d = 3$), and with V_0 denoting the volume of the elements having the lower left corner of the domain as a vertex. The factor α takes into account the cylindrical or spherical

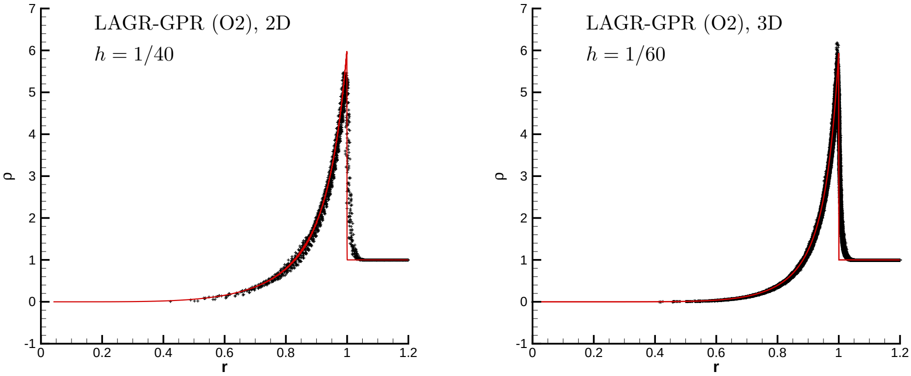


Figure 6.29. Scatterplots for the two-dimensional (left panel) and three-dimensional (right panel) Sedov problem, obtained with an updated Lagrangian scheme on unstructured triangular/tetrahedral meshes and the semi-analytical solver for strain relaxation applied to the equations of the unified model of continuum mechanics in the inviscid stiff relaxation limit. The solid line is an exact reference solution of the Euler equations. Note that the characteristic mesh size for the 2D case is $h = 1/40$, while for the 3D test, the characteristic mesh size is $h = 1/60$.

symmetry and is set to $\alpha = 4$ and $\alpha = 8$, respectively. An analytical solution can be derived from self-similarity arguments [200], making this test widely used in literature [3, 224]. Figure 6.29 shows the results for these two simulations, with a scatter plot of the cell density compared against the exact solution. The shock wave is correctly captured by the conservative LAGR-GPR scheme, and a very good symmetry of the density distribution can be appreciated. The density peak is captured well despite the coarse meshes, thanks to the Lagrangian nature of the scheme, which introduces less numerical viscosity, compared to Eulerian schemes on fixed grids at the same order of accuracy, as well as providing adaptive resolution around shocks where necessary.

6.3.4.2 Viscous shock profile

In order to verify the numerical method against supersonic viscous flows, we propose to solve the problem of an isolated viscous shock wave which is travelling into a viscous heat conducting fluid at rest with a shock Mach number of $\mathbb{M}\mathrm{a}_\mathrm{s} = 2$. The analytical solution can be found in [26], where the compressible Navier–Stokes–Fourier equations are solved for the special case of a stationary shock wave at Prandtl number $\mathbb{P}\mathrm{r} = 3/4$ with constant viscosity. The exact solution [26] for the dimensionless velocity $\bar{u} = u \left(\mathbb{M}\mathrm{a}_\mathrm{s}\, \gamma\, p_0\right)^{-1}$ of this stationary shock wave with shock Mach number $\mathbb{M}\mathrm{a}_\mathrm{s}$ is then given by the

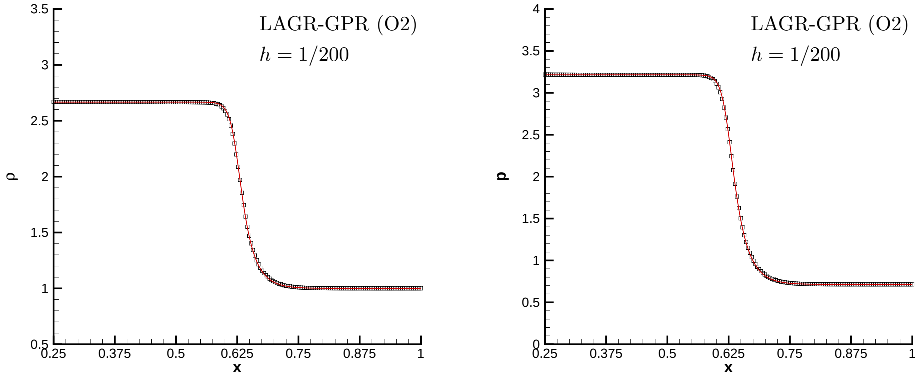


Figure 6.30. One-dimensional cuts of density ρ and pressure p along the x-axis for the viscous shockwave problem at Prandtl number $\mathbb{P}\mathrm{r} = 3/4$, obtained with an updated Lagrangian scheme on unstructured Delaunay meshes, employing the semi-analytical solver for strain relaxation. The solid line is an exact reference solution of the compressible Navier–Stokes equations.

root of

$$\frac{|\bar{u}-1|}{|\bar{u}-\kappa^2|^{\kappa^2}} = \left|\frac{1-\kappa^2}{2}\right|^{(1-\kappa^2)} \exp\left(\frac{3}{4}\,\mathbb{Re}\,\frac{\mathbb{M}\mathrm{a}_\mathrm{s}^2-1}{\gamma\,\mathbb{M}\mathrm{a}_\mathrm{s}^2}\,x\right), \tag{6.26}$$

with

$$\kappa^2 = \frac{1+\mathbb{M}\mathrm{a}_\mathrm{s}^2\,(\gamma-1)/2}{\mathbb{M}\mathrm{a}_\mathrm{s}^2\,(\gamma+1)/2}. \tag{6.27}$$

Equation (6.26) allows the dimensionless velocity $\bar{u}$ to be obtained as a function of x. The form of the viscous profile of the dimensionless pressure $\bar{p} = (p-p_0)\left(\gamma\,p_0\,\mathbb{M}\mathrm{a}_\mathrm{s}^2\right)^{-1}$ is given by the relation

$$\bar{p} = 1-\bar{u}+\left(\frac{1}{2\,\gamma}\right)\left(\frac{\gamma+1}{\gamma-1}\right)\frac{(\bar{u}-1)}{\bar{u}}\,(\bar{u}-\kappa^2). \tag{6.28}$$

Finally, the profile of the dimensionless density $\bar{\rho} = \rho/\rho_0$ is derived from the integrated continuity equation: $\bar{\rho}\,\bar{u} = 1$. To obtain an unsteady shock wave traveling into a medium at rest, it is sufficient to superimpose a constant velocity field $\mathbf{u} = (\mathbb{M}\mathrm{a}_\mathrm{s}\,\gamma\,p_0,\ 0,\ 0)^\intercal$ to the solution of the stationary shock wave found in the previous steps. The initial computational domain is the rectangular channel $\Omega(0) = [0,\ 1] \times [0,\ 0.2]$ which is paved with a triangular mesh of characteristic mesh size $h = 1/200$. On the left side of the domain ($x = 0$) the constant inflow velocity is prescribed, whereas periodic boundaries are set along the y direction and a constant pressure is imposed at $x = 1$. The initial condition involves a shock wave centered at $x = 0.25$ propagating at Mach $\mathbb{M}\mathrm{a}_\mathrm{s} = 2$ from left to right with a Reynolds number of $\mathbb{Re} = 100$. The polytropic index of the ideal gas is $\gamma = 1.4$ and the upstream shock state has density $\rho = 1$, velocity $\mathbf{u} = \mathbf{0}$, and pressure $p = 1/\gamma$.

Figure 6.30 illustrates a comparison against the analytical solution at the final time, where one can note an excellent matching. We compare the exact solution and the numerical solution, extracted as a 1D cut with 200 points along the x-direction at $y = 0.1$, for density, and pressure.

6.3.4.3 Viscous shock tube problem

The test cases presented so far deal with the stiff hydrodynamics limit of the GPR model. Here, we want to test the new LAGR-GPR scheme for the simulation of viscous fluids using the *same* set of equations. To this purpose, we solve in a two-dimensional setting the well-known Sod shock tube problem, which is a classical one-dimensional test problem that involves a rarefaction wave travelling towards the left boundary as well as a right-moving contact discontinuity and a shock wave travelling to the right. However, instead of ideal

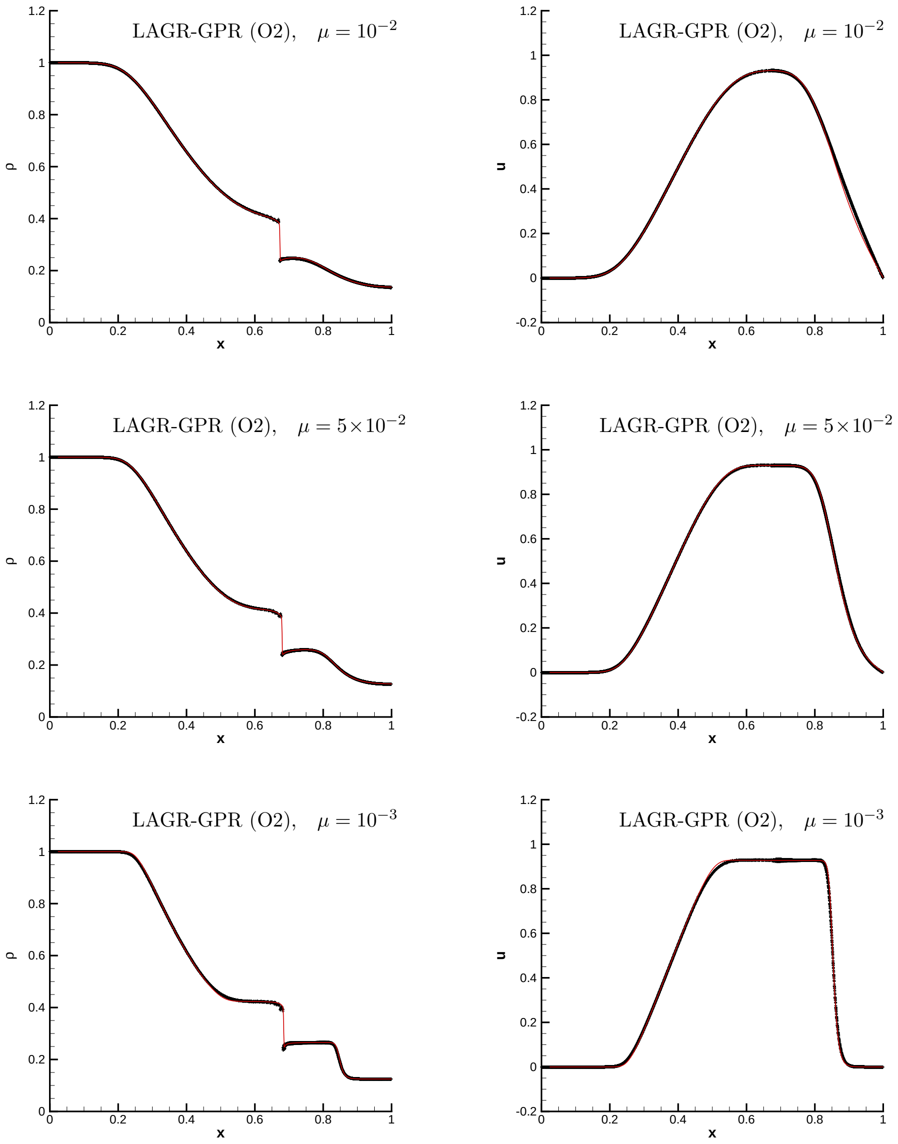


Figure 6.31. Scatterplots for the viscous shock tube problem, obtained with an updated Lagrangian scheme on unstructured Delaunay meshes, employing the semi-analytical solver for strain relaxation. The solid line is a reference solution computed by solving the compressible Navier–Stokes equations on a fine mesh of 10 000 cells with a second order MUSCL–Hancock TVD scheme.

fluids, viscous ones are now considered, with three different viscosity coefficients, namely $\mu = 10^{-3}$, $\mu = 5 \times 10^{-3}$ and $\mu = 10^{-2}$. The initial computational domain is the rectangular box $\Omega(0) = [0,\ 1] \times [0,\ 0.1]$ which is discretized with uniform triangular elements having a characteristic mesh size of $h = 1/200$. Slip-wall boundaries are set everywhere, hence prescribing zero normal velocity in the nodal solver. The final time is $t_f = 0.2$ so that all waves remain bounded in the computational domain. The initial condition consists in a discontinuity located at $x_0 = 0.5$ with left and right states defined by

$$(\rho(x),\ u(x),\ p(x)) = \begin{cases} (1,\ 0,\ 1)\,, & \text{if} \quad x \leq x_0, \\ (0.125,\ 0,\ 0.1)\,, & \text{if} \quad x > x_0. \end{cases} \tag{6.29}$$

The fluid is initially at rest and the ideal gas EOS is adopted with $\gamma = 1.4$, while thermal conduction is neglected. The results of the LAGR-GPR scheme are compared with a reference solution of the Navier–Stokes equations without thermal conduction computed on a very fine one-dimensional mesh of 10 000 cells with a MUSCL–Hancock TVD finite volume method. Figure 6.31 shows a scatter plot for density, horizontal velocity and pressure of the numerical solution compared against the reference solution of the Navier–Stokes model. An excellent agreement is obtained for all different values of the viscosity coefficient, demonstrating that the novel LAGR-GPR scheme, in conjunction with the proposed semi-analytical integrator is capable of accurately simulating viscous fluids.

6.3.5 Fracture generation in a circular block of pre-damaged rock

In order to show the applicability of our new algorithm to realistic experiments that are frequently carried out in civil engineering and in geophysics, we consider the crack propagation in a pre-damaged rock-like disc (a so-called Brazilian test), with the aim of comparing to the experimental results given in [174]. The disc is described with the moving diffuse interface approach, and its boundary is thus represented by means of the solid volume fraction α. An inclined slit is represented as predamaged zone. In order to match the experimental data, we rotated the disc with respect to the clamping apparatus in such a way that the slit has an inclination angle of 35° with respect to the x-axis. We impose a velocity pointing towards the disc on the upper and bottom boundary, with a magnitude given by the Gaussian profile $|v| = 4\ \exp\left[-(25\,x)^2\right]$. We take our domain to be $\Omega = [-1.1,\ 1.1]^2$ and employ a fourth order ADER-DG scheme with *a-posteriori* subcell limiting. The material properties for the disc are $\rho_0 = 1440\,\mathrm{kg\,m^{-3}}$, $\mu_I = 6.20\,\mathrm{GPa}$, $\mu_D = 0.006198\,\mathrm{GPa}$, $\lambda_I = 4.49\,\mathrm{GPa}$, $\lambda_D = 4.49\,\mathrm{GPa}$, $\vartheta_0 = 10.0$, $Y_0 = 0.55\,\mathrm{MPa}$, $Y_1 = 0.001\,\mathrm{MPa}$, $a = 52.5$,

$\alpha_I = 36.25$, $\alpha_D = 36.25$, $\beta_I = 0.0\,\mathrm{Pa}^{-1}$, $\beta_D = 10^{-6}\,\mathrm{Pa}^{-1}$, while in the top and bottom clamps the material is modified by setting $Y_0 = Y_1 = 100\,\mathrm{TPa}$, effectively rendering these regions unbreakable. For the equivalent stress we take a combination of shear stress and pressure defined as

$$Y = A\,Y_\mathrm{s} + B\,Y_\mathrm{p} + C \tag{6.30}$$

where $A = 1.0$, $B = 1.5$, $C = -2.0\,\mathrm{MPa}$, $Y_\mathrm{s} = \sqrt{3\ \mathrm{tr}\left(\mathrm{dev}\,\boldsymbol{\Sigma}\ \mathrm{dev}\,\boldsymbol{\Sigma}\right)/2}$ is the von Mises stress, while $Y_\mathrm{p} = \mathrm{tr}\,\boldsymbol{\Sigma}/3$ accounts for the spherical part of the stress tensor. The choice $A = 1$, $B = C = 0$, gives $Y = Y_\mathrm{s}$, that is, the von Mises stress, while other choices of coefficients are intended to describe a Drucker–Prager-type yield criterion. The resulting evolution of the crack is shown in Fig. 6.32. The main crack propagates starting from the corners of the pre-damaged zone, following the experimental results very closely to the

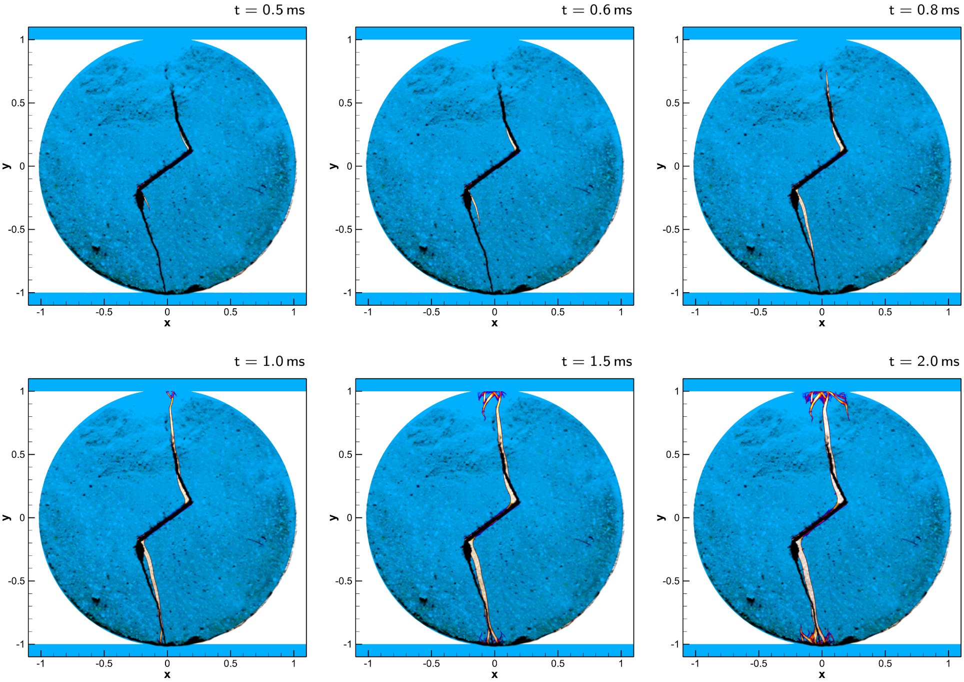


Figure 6.32. Evolution of the rupture front (colour contours of the damage field ξ) compared with experiments at times $t = [0.5,\ 0.6,\ 0.8,\ 1.0,\ 1.5,\ 2.0]$ ms. The computations are carried out with a fourth order ADER-DG scheme with *a-posteriori* subcell limiting on a uniform Cartesian grid of 256×256 elements. Only regions with $\alpha > 0.5$ are shown. The experimental picture is that of the experiment conducted in [174].

top and bottom of the disc. In agreement with the experimental observations, we have also a damaged zone due to shear on the upper and lower part of the disc, while the rupture inside is driven by traction. Last but not least, we add a careful mesh convergence analysis for this nontrivial test problem, in order to check whether the position and shape of the crack converges with mesh refinement. The result of this detailed mesh refinement study carried out on a sequence of uniform Cartesian grids with mesh resolution from 48 × 48 elements to 256 × 256 elements is shown in Figure 6.33. One can observe that the main cracks properly converge with mesh refinement, in particular the crack becomes thinner on finer meshes. Only the secondary cracks generated at the upper and lower boundaries of the disc differ between one mesh and the other.

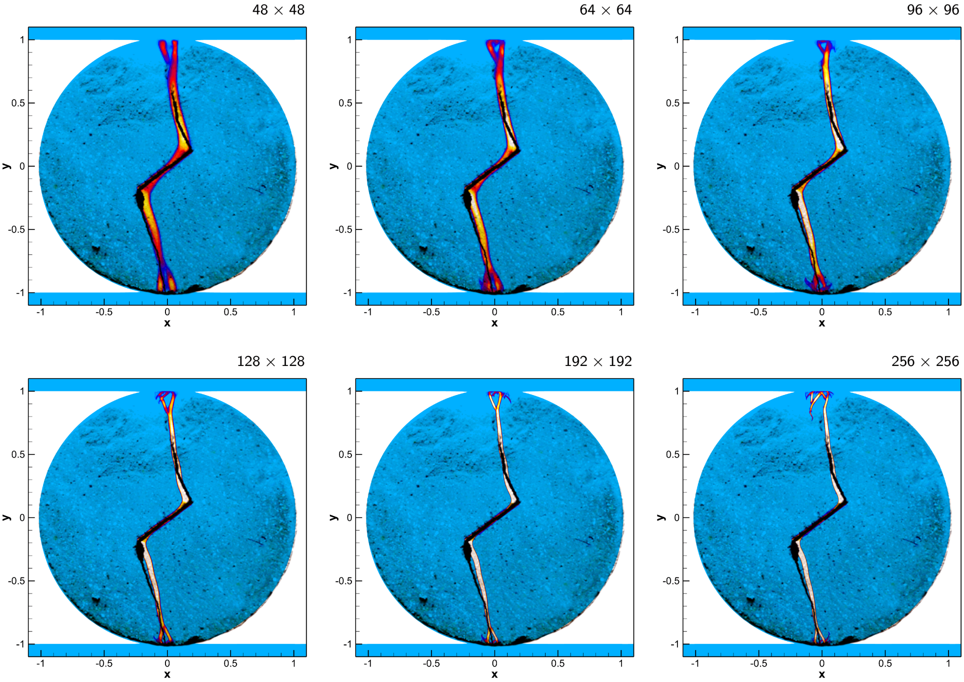


Figure 6.33. Mesh convergence study (colour contours of the damage field ξ) for the simulation of a pre-damaged rock disc. The computations employ a fourth order ADER-DG scheme with MUSCL-Hancock *a posteriori* subcell finite volume limiting and show mesh convergence of the main cracks for six grid spacings ranging from 48×48 to 256×256 Cartesian cells. Only regions with $\alpha > 0.5$ are shown. The experimental picture has been that of the experiment conducted in [174].

6.4 Structure-preserving semi-implicit schemes for viscous two-phase flow with surface tension

In this Section we present the yet to be published [85] computational results obtained by means of our semi-implicit, curl-preserving Finite Volume scheme for viscous two-phase flow with surface tension. The results are comprised of an extensive suite of validation benchmarks, as well as application to real-world problems, most notably, to Rayleigh–Taylor gravity-driven flow instabilities.

6.4.1 Experimental verification of the Abgrall condition

We begin the validation of the proposed numerical method by showing that the implementation of the numerical scheme does indeed satisfy the Abgrall condition [2], i.e. it preserves uniformity of the velocity and pressure fields regardless of the distribution of density or volume fraction. The test is carried out on a domain $\Omega = [-1,\ 1] \times [-1,\ 1]$, where we set up a circular jump in ρ_1, ρ_2, and α_1. Precisely, if $r = \|\mathbf{x}\| < 1/2$ we set $\rho_1 = 1$, $\rho_2 = 1/2$, and $\alpha_1 = 0.3$ and otherwise if $r \geq 1/2$ we have $\rho_1 = 1/2$, $\rho_2 = 1$, and $\alpha_1 = 0.7$. The uniform pressure is $p = 1$ and a constant velocity field $\mathbf{u} = (1,\ 1,\ 0)^\mathsf{T}$ is initially imposed, so that the solution will consist in simple advection (and numerical diffusion) of the initial circular density and volume fraction jump. Surface tension, shear, and gravity effects are not present, hence we set $\sigma = 0$ and $\mathbf{A} = \mathbf{0}$, $\mathbf{b} = \mathbf{0}$, $\mathbf{g} = \mathbf{0}$, $\tau = 10^{-14}$, $c_\mathrm{s} = 0$, and the governing equations reduce to Kapila's model. The material parameters for this test are $\gamma_1 = 4.0$, $\gamma_2 = 1.4$, $\Pi_1 = 2.0$, $\Pi_2 = 0$. Throughout this section, unless explicitly noted,

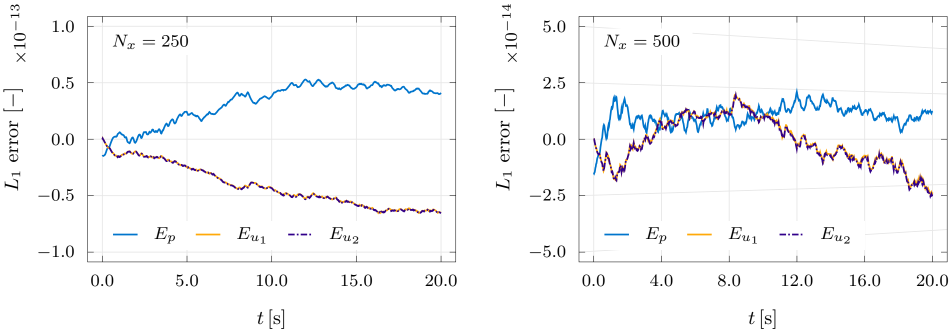


Figure 6.34. Timeseries of errors for the Abgrall condition verification test for the pressure field p, and for the velocity components u_1, u_2. No linear or exponential growth is observed, showing that indeed the implementation of the scheme does satisfy the Abgrall condition.

we adopt SI units for all quantities.

In Figure 6.34 we report the time evolution of errors (in the L_1 norm) for both nonzero components of the velocity field and for pressure with regard to simulations carried out on two different uniform Cartesian grids of 250^2 and 500^2 elements. For both meshes, the errors are of order 10^{-14} to 10^{-13} and present no exponential or linear growth, thus they can perfectly be explained as accumulation of roundoff errors due to floating point arithmetic and small errors due to the iterative solution of the discrete wave equation for the pressure field.

6.4.2 Numerical convergence study for a steady droplet in equilibrium

A numerical convergence study is carried out in order to assess the order of accuracy of the proposed semi-implicit curl-preserving method. The problem setup is described in Section 6.2.1 and consists of the uniform convective transport of a diffuse droplet initialised according to the exact solution derived in Section 2.2.6. Here we carry out an additional second test with a stationary droplet in order to assess the convergence rates both in steady and unsteady problems. The order is theoretically expected to be 2 for steady solutions, since all fluxes are integrated separately with second order accurate discretisations. In the unsteady case, due to the first order splitting of convection, pressure, and capillarity effects, we expect that the scheme be first order accurate. Error norms and convergence rates, with respect to the variables $\alpha_1\,\rho_1$, $\rho\,\mathbf{u}$, $\rho\,E$, α_1, and b_1, are given in Table 6.4 for the steady case and in Table 6.5 for the unsteady one. Both sets of simulations experimentally confirm the expected order of accuracy.

6.4.3 Validation of the viscosity model and algorithms

First problem of Stokes. In the context of this thesis, the test serves the dual purpose of showing that the unified model of continuum mechanics does indeed include the Navier–Stokes equations as a special case, and that the semi-analytical integration scheme can capture the same limit. An important benchmark for viscous flow solvers is the first problem of Stokes: for this test the computational domain is $\Omega = [0,\ 1] \times [0,\ 0.1]$, with periodic boundary conditions in y direction and wall boundaries in the x direction. The initial condition of the problem is given by uniform density $\rho = 1$ and pressure $p = 1/\gamma$. The x-component of the velocity field is initialized as $u_1 = 0$, the distortion field is initially set to $\mathbf{A} = \mathbf{I}$, while the y-velocity component u_2 is $u_2 = -(u_2)_0$

Table 6.4. Numerical convergence results for the curl-preserving semi-implicit scheme applied to a droplet in equilibrium under surface tension forces at rest. With N_x we indicate the number of cells in one row of the Cartesian computational grid.

	N_x	$\mathcal{E}_{L_1}$	$\mathcal{E}_{L_2}$	$\mathcal{E}_{L_\infty}$	$\mathcal{O}_{L_1}$	$\mathcal{O}_{L_2}$	$\mathcal{O}_{L_\infty}$
$\alpha_1\,\rho_1$	128	8.72×10^{-3}	4.01×10^{-3}	5.48×10^{-3}	–	–	–
	192	3.48×10^{-3}	1.68×10^{-3}	2.89×10^{-3}	2.27	2.15	1.58
	256	1.81×10^{-3}	9.17×10^{-4}	2.08×10^{-3}	2.28	2.10	1.16
	384	7.08×10^{-4}	3.87×10^{-4}	1.04×10^{-3}	2.31	2.13	1.72
	512	3.80×10^{-4}	2.16×10^{-4}	6.64×10^{-4}	2.16	2.04	1.54
$\rho\,u$	128	5.94×10^{-4}	3.06×10^{-4}	4.88×10^{-4}	–	–	–
	192	3.72×10^{-4}	2.04×10^{-4}	3.59×10^{-4}	1.16	1.00	0.76
	256	2.40×10^{-4}	1.36×10^{-4}	2.49×10^{-4}	1.52	1.42	1.28
	384	1.12×10^{-4}	6.94×10^{-5}	1.38×10^{-4}	1.72	1.66	1.46
	512	6.88×10^{-5}	4.06×10^{-5}	8.73×10^{-5}	1.93	1.87	1.59
$\rho\,E$	128	2.56×10^{-3}	1.19×10^{-3}	1.66×10^{-3}	–	–	–
	192	1.03×10^{-3}	4.98×10^{-4}	8.71×10^{-4}	2.27	2.15	1.58
	256	5.34×10^{-4}	2.73×10^{-4}	6.26×10^{-4}	2.28	2.10	1.15
	384	2.10×10^{-4}	1.16×10^{-4}	3.13×10^{-4}	2.30	2.11	1.71
	512	1.13×10^{-4}	6.46×10^{-5}	2.01×10^{-4}	2.15	2.03	1.54
α_1	128	8.34×10^{-3}	3.88×10^{-3}	5.38×10^{-3}	–	–	–
	192	3.33×10^{-3}	1.62×10^{-3}	2.83×10^{-3}	2.26	2.15	1.58
	256	1.73×10^{-3}	8.88×10^{-4}	2.04×10^{-3}	2.28	2.10	1.15
	384	6.81×10^{-4}	3.77×10^{-4}	1.02×10^{-3}	2.30	2.11	1.71
	512	3.67×10^{-4}	2.10×10^{-4}	6.52×10^{-4}	2.15	2.03	1.54
b_1	128	7.67×10^{-2}	3.88×10^{-2}	5.34×10^{-2}	–	–	–
	192	3.59×10^{-2}	2.04×10^{-2}	3.11×10^{-2}	1.87	1.59	1.33
	256	2.19×10^{-2}	1.32×10^{-2}	2.21×10^{-2}	1.72	1.52	1.19
	384	1.11×10^{-2}	7.05×10^{-3}	1.39×10^{-2}	1.68	1.54	1.15
	512	6.51×10^{-3}	4.38×10^{-3}	9.96×10^{-3}	1.86	1.65	1.15

for $x < 1/2$ and $u_2 = (u_2)_0$ for $x \geq 1/2$. The parametric quantities for this test are given as $(u_2)_0 = 0.1$, $\gamma_1 = \gamma_2 = 1.4$, $\Pi_1 = \Pi_2 = 0$, $\rho_0 = 1$, $c_\mathrm{s} = 1$.

The simulations are performed with the structure preserving semi-implicit finite volume scheme on a grid composed of 500 by 50 square control volumes up to a final time of $t = 0.4$. The exact solution of the incompressible Navier–Stokes equations for this test case is expressed in terms of the y-velocity component

Table 6.5. Numerical convergence results for the curl-preserving semi-implicit scheme applied to a droplet in equilibrium under surface tension forces moving in a uniform flow. With N_x we indicate the number of cells in one row of the Cartesian computational grid.

	N_x	$\mathcal{E}_{L_1}$	$\mathcal{E}_{L_2}$	$\mathcal{E}_{L_\infty}$	$\mathcal{O}_{L_1}$	$\mathcal{O}_{L_2}$	$\mathcal{O}_{L_\infty}$
$\alpha_1\,\rho_1$	128	3.45×10^{-2}	1.67×10^{-2}	1.70×10^{-2}	–	–	–
	192	2.27×10^{-2}	1.08×10^{-2}	1.10×10^{-2}	1.04	1.07	1.09
	256	1.71×10^{-2}	8.17×10^{-3}	8.02×10^{-3}	0.98	0.97	1.08
	384	1.16×10^{-2}	5.55×10^{-3}	5.27×10^{-3}	0.96	0.95	1.04
	512	8.83×10^{-3}	4.22×10^{-3}	4.02×10^{-3}	0.95	0.95	0.94
$\rho\,u$	128	3.69×10^{-2}	1.69×10^{-2}	1.75×10^{-2}	–	–	–
	192	2.44×10^{-2}	1.11×10^{-2}	1.13×10^{-2}	1.02	1.03	1.08
	256	1.87×10^{-2}	8.48×10^{-3}	8.31×10^{-3}	0.92	0.93	1.07
	384	1.29×10^{-2}	5.82×10^{-3}	5.47×10^{-3}	0.92	0.93	1.03
	512	9.83×10^{-3}	4.44×10^{-3}	4.20×10^{-3}	0.94	0.94	0.92
$\rho\,E$	128	1.06×10^{-2}	5.12×10^{-3}	5.24×10^{-3}	–	–	–
	192	6.96×10^{-3}	3.31×10^{-3}	3.37×10^{-3}	1.03	1.07	1.09
	256	5.26×10^{-3}	2.51×10^{-3}	2.47×10^{-3}	0.98	0.97	1.08
	384	3.57×10^{-3}	1.70×10^{-3}	1.62×10^{-3}	0.96	0.95	1.04
	512	2.71×10^{-3}	1.30×10^{-3}	1.24×10^{-3}	0.96	0.95	0.93
α_1	128	3.44×10^{-2}	1.66×10^{-2}	1.70×10^{-2}	–	–	–
	192	2.26×10^{-2}	1.08×10^{-2}	1.09×10^{-2}	1.03	1.07	1.09
	256	1.71×10^{-2}	8.15×10^{-3}	8.02×10^{-3}	0.98	0.97	1.08
	384	1.16×10^{-2}	5.54×10^{-3}	5.26×10^{-3}	0.96	0.95	1.04
	512	8.81×10^{-3}	4.21×10^{-3}	4.02×10^{-3}	0.96	0.95	0.93
b_1	128	3.29×10^{-1}	1.44×10^{-1}	1.43×10^{-1}	–	–	–
	192	2.17×10^{-1}	9.79×10^{-2}	1.03×10^{-1}	1.03	0.95	0.81
	256	1.60×10^{-1}	7.33×10^{-2}	7.98×10^{-2}	1.06	1.01	0.88
	384	1.05×10^{-1}	4.90×10^{-2}	5.55×10^{-2}	1.04	0.99	0.90
	512	7.70×10^{-2}	3.65×10^{-2}	4.34×10^{-2}	1.08	1.03	0.86

u_2 as

$$u_2(x,\ t) = (u_2)_0\,\mathrm{erf}\left(\frac{x}{2\,\sqrt{\nu\,t}}\right), \tag{6.31}$$

with $\nu = \mu/\rho_0$. The test is repeated for three different values of kinematic viscosity $\nu = 10^{-2}$, $\nu = 10^{-3}$, $\nu = 10^{-4}$. The comparison between the Navier–Stokes reference solution (6.31) and the numerical results obtained with the new scheme for the unified model of continuum mechanics are presented in Fig. 6.35, where one can observe an excellent agreement between the two for various kinematic viscosities ν. This proves that the proposed numerical algorithm can accurately capture the Navier–Stokes regime of the governing equations.

Double shear layer problem. Here we solve the double shear layer test problem [27, 58, 120, 310]. The computational domain is $\Omega = [0,\ 1]^2$ with periodic boundary conditions everywhere. The x-component of the velocity field is initialised as

$$u_1 = \begin{cases} \tanh\left[(y - 0.25)\,\tilde{\rho}\right], & \text{if } y \leq 0.5, \\ \tanh\left[(0.75 - y)\,\tilde{\rho}\right], & \text{if } y > 0.5, \end{cases} \tag{6.32}$$

and the y-components is $u_2 = \delta\,\sin(2\,\pi\,x)$, the density is initially uniform $\rho = \rho_0 = 1$ and the pressure is $p = 100/\gamma_1$. Since the test problem is usually adopted in a single-phase context, we set the volume fraction function is $\alpha = 0.5$ throughout the computational domain to emulate the single-phase equations: both phases compute the exact same solution, and one simply obtains the mixture (single-phase) density by direct sum of the two partial densities.

For this test case we set the parameters that determine the shape of the velocity field to $\delta = 0.05$ and $\tilde{\rho} = 30$. The viscosity coefficient is set to $\nu = \mu/\rho_0 = 2 \times 10^{-4}$ and $\nu = \mu/\rho_0 = 2 \times 10^{-3}$ in two separate runs of the test problem. The other parameters of the model are $\gamma_1 = \gamma_2 = 1.4$, $\Pi_1 = \Pi_2 = 0$,

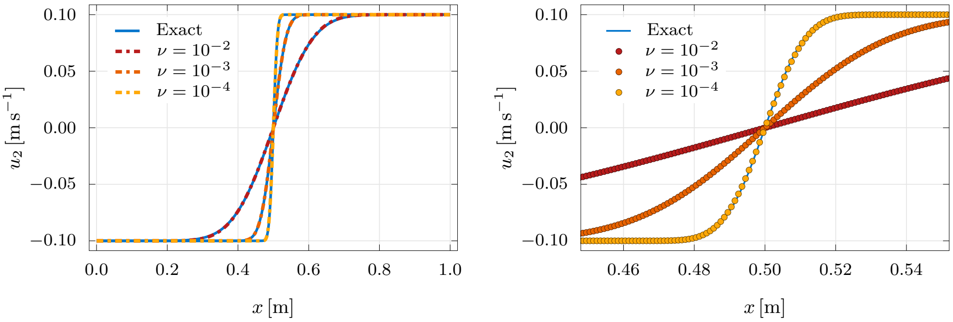


Figure 6.35. Numerical solutions of the first problem of Stokes, for three different values of the kinematic viscosity ν. The simulations are carried out with the semi-implicit structure-preserving Finite Volume scheme employing the semi-analytical integrator for strain relaxation. The right panel is a zoomed-in view about the location of the shear interface, showing perfect agreement of the GPR model with the analytical solution of the Navier–Stokes equations.

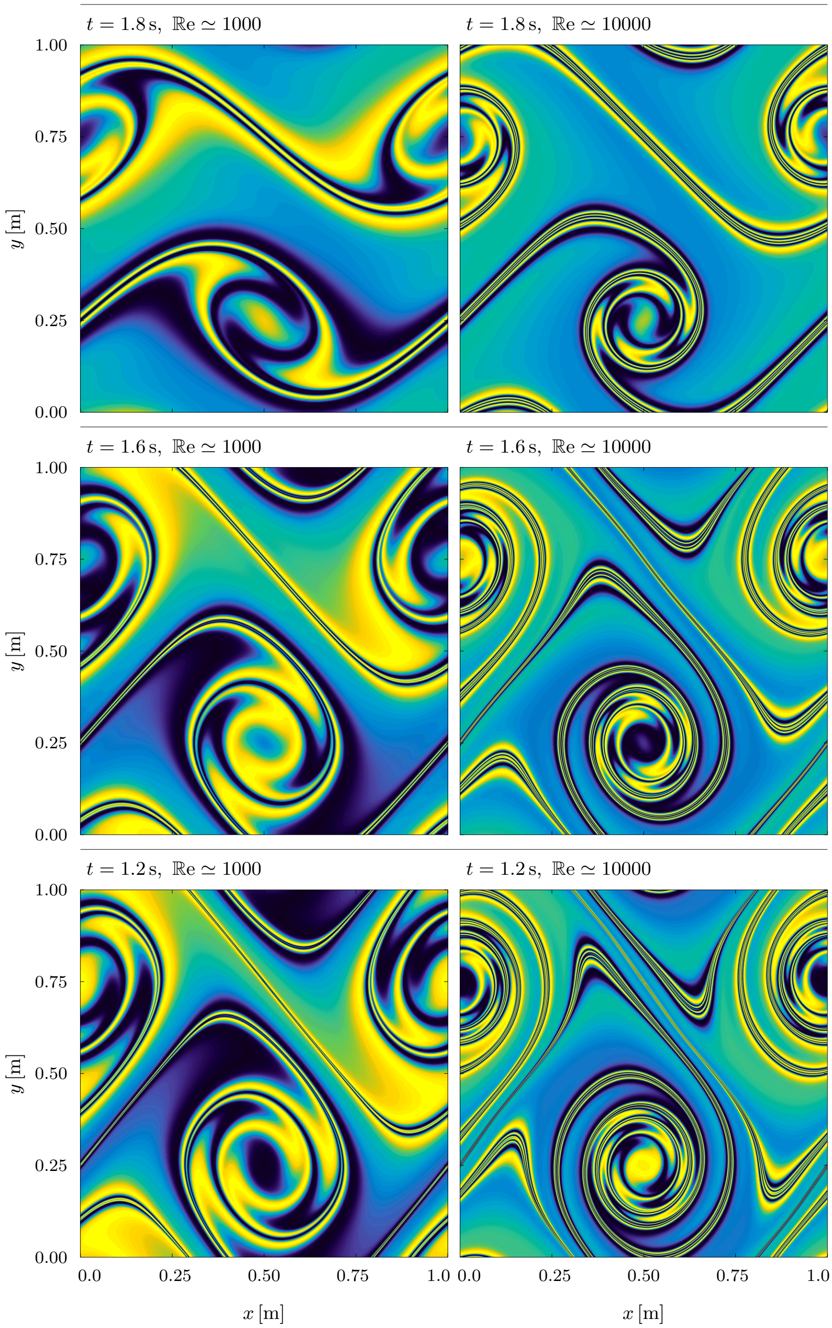


Figure 6.36. Filled contours of the A_{12} component of the distortion field $\mathbf{A}$ in the double shear layer problem for two values of kinematic viscosity $\nu = 2{\times}10^{-3}\,\mathrm{m}^2\,\mathrm{s}^{-1}$ ($\mathbb{R}\mathrm{e} \simeq 1000$) and $\nu = 2{\times}10^{-4}\,\mathrm{m}^2\,\mathrm{s}^{-1}$ ($\mathbb{R}\mathrm{e} \simeq 10000$).

$\rho_0 = 1$, $c_\mathrm{s} = 8$. The initial condition for the distortion field is $\mathbf{A} = \mathbf{I}$ and surface tension effects are not to be accounted for in this test, which means that we set $\mathbf{b} = \mathbf{0}$. Simulations are carried out with the new structure-preserving semi-implicit finite volume scheme up to a final time of $t = 1.8$. The computational mesh is composed of 5120×5120 control volumes. In Figure 6.36 we show the temporal evolution of the distortion field component A_{12}. The results, highlighting the incredibly rich structure found in the rotational component $\mathbf{R}$ of the distortion field $\mathbf{A}$, are in excellent agreement with those obtained in [58] using a thermodynamically compatible scheme on the same fine uniform grid.

6.4.4 Riemann problems and circular explosion problem

We continue the validation of the semi-implict scheme with two one-dimensional Riemann Problems showing that the semi-implicit numerical method can reproduce the correct wave structure of the Kapila model [201] of two-phase flow: in these tests, surface tension, shear, and gravity effects are not present, hence we set $\sigma = 0$ and $\mathbf{A} = \mathbf{I}$, $\mathbf{b} = \mathbf{0}$, $\mathbf{g} = \mathbf{0}$, $\tau = 10^{-14}$, $c_\mathrm{s} = 0$, and the governing equations reduce to Kapila's model exactly. First, we set up two simple planar Riemann problems RP1 and RP2 by partitioning the computational domain in two regions with constant state separated by a discontinuity normal to the x-direction, i.e. for RP1 we set $\rho_1 = 1$, $\rho_2 = 1$, $\mathbf{u} = 0$, $p = 1$, $\alpha_1 = 0.5$, if $x \leq 0.5$, and $\rho_1 = 1$, $\rho_2 = 0.1$, $\mathbf{u} = 0$, $p = 0.1$, $\alpha_1 = 0.5$ otherwise.

In Figures 6.37 and 6.38, we show the one-dimensional profiles of the partial densities $\alpha_1\,\rho_1$ and $\alpha_2\,\rho_2$, of the x-component of the velocity field u_1, and of the pressure p, obtained by applying the semi-implicit scheme on a mesh of 4000 by 400 square cells over the domain $\Omega = [0,\ 1] \times [0,\ 0.1]$. The results are compared with a reference solutions computed by a second order explicit path-conservative TVD MUSCL–Hancock method on a fine one-dimensional grid compsed of 40 000 uniform control volumes. The results match up to some minor artifacts observed in the solution obtained from the semi-implicit method, in this regard, it is important to note that such a semi-implicit scheme is designed specifically for low Mach number flows, rather than for the treatment of shockwaves. The observed artifacts are not a hinderance for large scale simulations lying within the design regime of the method.

Then, a two-dimensional explosion problem is set up in a similar fashion, with the discontinuity now representing an inner and outer state rather than a left and right one. The computational domain is the square $\Omega = [-0.8,\ 0.8] \times [-0.8,\ 0.8]$ and is disctretised with a mesh of 4096^2 uniform Cartesian control volumes. The inner state is given by $\rho_1^\mathrm{i} = 1000$, $\rho_2^\mathrm{i} = 1000$, $\mathbf{u}^\mathrm{i} = \mathbf{0}$, $p^\mathrm{i} = 10^{10}$, $\alpha_1^\mathrm{i} = 0.5$. The outer state is $\rho_1^\mathrm{o} = 1000$, $\rho_2^\mathrm{o} = 1$, $\mathbf{u}^\mathrm{o} = \mathbf{0}$, $p^\mathrm{o} = 10^5$, $\alpha_1^0 = 0.5$. The discontinuity is initially located at $r = \|\mathbf{x}\| = 1/2$, and the final simulation time is $t_{\text{end}} = 6 \times 10^{-5}$. The parameters of the stiffened gas equation of state are $\gamma_1 = 4.4$, $\gamma_2 = 1.4$, $\Pi_1 = 6 \times 10^8$, $\Pi_2 = 0$.

In Figure 6.39 we show the filled contour plots for the mixture density ρ, for the pressure p, for the volume fraction α_1, and for the x-component of the velocity field u_1. The results correctly preserve symmetries and are in agreement with the reference solutions obtained by means of an explicit path-conservative second order TVD MUSCL–Hancock method on the same mesh of 4096^2 cells.

6.4.5 Long-time evolution of an oscillating droplet at low Mach number

In this Section, we reproduce the numerical experiments shown in Section 6.2.4 with regard to high order ADER Discontinuous Galerkin $\mathbb{P}_N\mathbb{P}_N$ schemes with

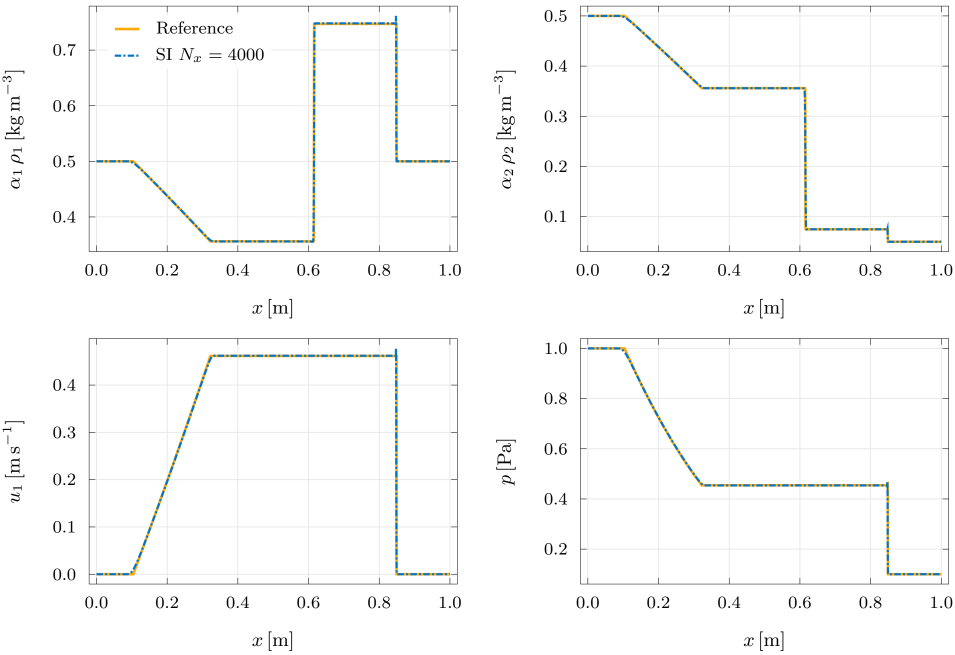


Figure 6.37. Numerical solution of the multiphase Riemann problem RP1 obtained with the semi-implict curl-preserving scheme on a uniform Cartesian grid with mesh size $\Delta x = 1/4000$, compared with a reference solution obtained with a standard explicit path-conservative second order TVD MUSCL–Hancock scheme on a mesh of 40 000 cells.

a posteriori Finite Volume subcell limiting, now using instead the novel semi-implicit curl-preserving scheme introduced in Chapter 5.

While we refer to Section 6.2.4 for a detailed exposition of the computational setup, we list here the parameters of the problem: the density fields are set to the uniform values ρ_1^0 and ρ_2^0 throughout the computational domain, as is the velocity field for which we set $\mathbf{u} = (0,\ 0,\ 0)^\mathsf{T}$. The numerical values employed for this test problem are: $\rho_1^0 = 1000\,\mathrm{kg\,m^{-3}}$, $\rho_2^0 = 1\,\mathrm{kg\,m^{-3}}$, $p_{\mathrm{atm}} = 100\,\mathrm{kPa}$, $R_x = 3\,\mathrm{mm}$, $R_y = 2\,\mathrm{mm}$, $\alpha_{\min} = 0.01$, $\alpha_{\max} = 0.99$, $\sigma = 60\,\mathrm{N\,m^{-1}}$. The parameters for the stiffened gas equation of state are: $\Pi_1 = 1\,\mathrm{MPa}$, $\Pi_2 = 0$, $\gamma_1 = 4$, $\gamma_2 = 1.4$. The domain is the square $\Omega = [-6\,\mathrm{mm},\ 6\,\mathrm{mm}] \times [-6\,\mathrm{mm},\ 6\,\mathrm{mm}]$ and additionally, the initial condition is rotated counter-clockwise by 30 degrees, in order to avoid mesh alignment. In a first batch of tests, we set $\varepsilon = 0.5\,\mathrm{mm}$ and discretise the computational domain with 384^2 square cells, which yields the same number of degrees of freedom previously employed for the same test with the ADER-DG $\mathbb{P}_5\mathbb{P}_5$ scheme. These simulations are intended to test the

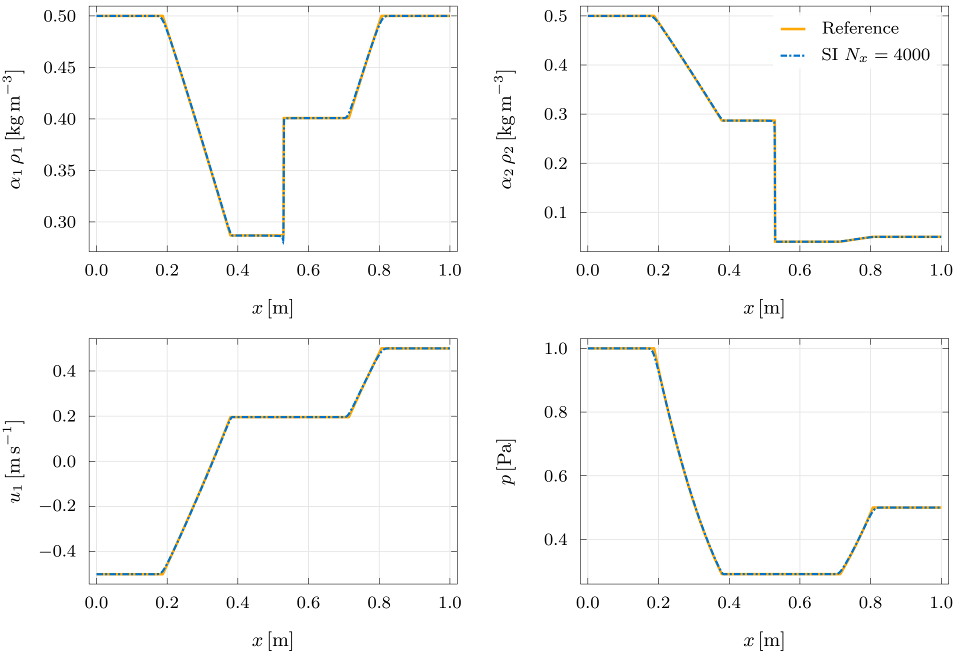


Figure 6.38. Numerical solution of the multiphase Riemann problem RP2 obtained with the semi-implicit curl-preserving scheme on a uniform Cartesian grid with mesh size $\Delta x = 1/4000$, compared with a reference solution obtained with a standard explicit path-conservative second order TVD MUSCL–Hancock scheme on a mesh of 40 000 cells.

capability of the method in a dynamical setting where the interface deforms significantly under the effect of strong surface tension, and show that the curl-preserving semi-implicit scheme, when applied to such low Mach problems, can be competitive even with very high order methods and in particular that that the GLM curl cleaning approach can deal with the violations of involution constraints that such deformations generate.

In Figure 6.40, we show the time evolution of the total kinetic energy and compare the results obtained without any enforcement of curl involutions (which we recall, as shown in Section 6.2.4, are totally unstable and quickly lead to the breakdown of computations), with those from GLM curl cleaning formulation of the governing equations (this time implemented in the semi-implicit code) with the results from the structure preserving methodology. Both curl-control

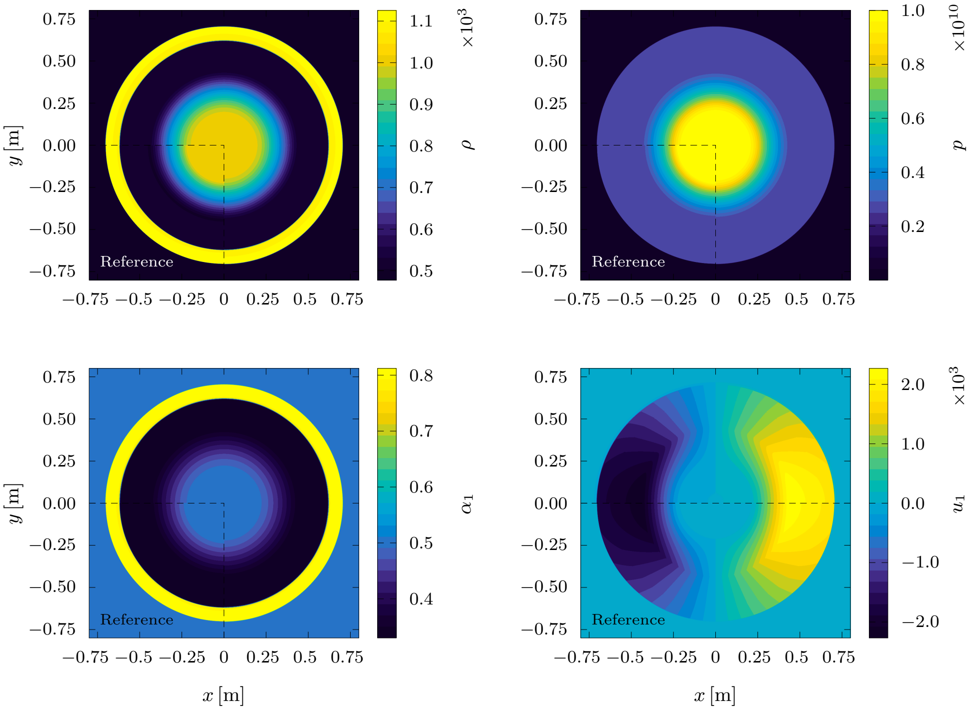


Figure 6.39. Filled contour plots for the two-phase inviscid circular explosion problem with the curl-preserving semi-implicit scheme for viscous two-phase flow on a fine uniform Cartesian grid counting 4096^2 cells. In the lower half of each panel, we show a reference solution computed with an explicit path-conservative second order method on the same fine uniform mesh of 4096^2 elements.

solutions yield stable results, but higher accuracy is achieved with the exactly curlfree scheme.

The same can be observed in Figure 6.41, where we plot a three-dimensional view of the second component of the velocity field u_2 for the at time $t = 6.91 \times 10^{-4}\,\mathrm{s}$ (half of the first oscillation cycle), computed on two different meshes of size 384^2 and 512^2, and for both the GLM variant of the method and for the exactly curlfree method. The difference in the quality of the results obtained with the two different approaches is immediately apparent by comparing the *specular highlights* in the plots on the left column, obtained with GLM curl cleaning (the surface is rough), with those on the right column, obtained with the exactly curlfree scheme (the surface is perfectly smooth). Moreover, note that the smoothness of the curlfree solution is not due to numerical diffusion (peak heights are the same) and that the observations are the same for different meshes.

In Figure 6.42 we show the time evolution of curl errors. Specifically, we compare the curl errors given by the GLM curl cleaning variant of the model with cleaning speed c_h ranging from $5\,\mathrm{m\,s^{-1}}$ to $40\,\mathrm{m\,s^{-1}}$, and those given by the curl-preserving semi-implicit method. The errors are computed in the

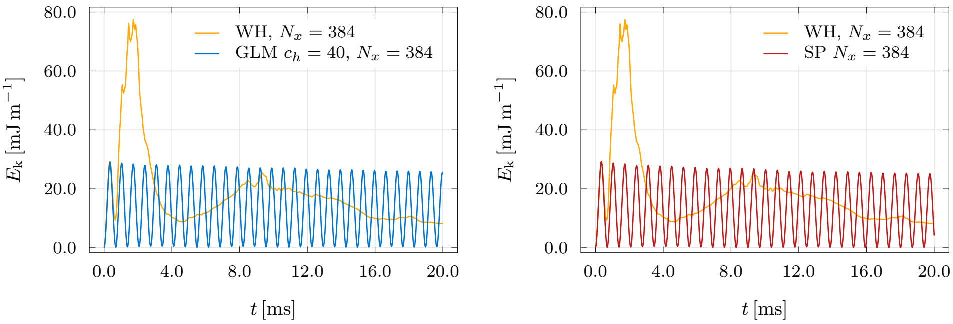


Figure 6.40. Time evolution of the total kinetic energy E_k for the elliptical droplet oscillation problem. On the left: solution obtained with the a second order semi-implicit Finite Volume scheme with GLM curl cleaning on a uniform Cartesian mesh composed of 384 × 384 elements. On the right: solution obtained on the same mesh with the curl-preserving variant of the same semi-implicit scheme. In both cases it is apparent that the numerical diffusion is much lower than that of standard second order TVD schemes (see Figures 6.21 and 6.22): the oscillation amplitude shows only mild decay for the full 15 oscillation cycles simulated in the numerical experiment. For comparison, without involution enforcement, the weakly hyperbolic model (WH) produces chaotic results before the end of the first oscillation cycle.

L_1 and the L_2 norms and are relative to a common uniform Cartesian mesh counting 192^2 elements. The curl-preserving semi-implicit method produces errors more than nine orders of magnitude smaller (less than $\sim 10^{-13}$ instead of more than $\sim 10^{-4}$) than those obtained with GLM curl cleaning in the

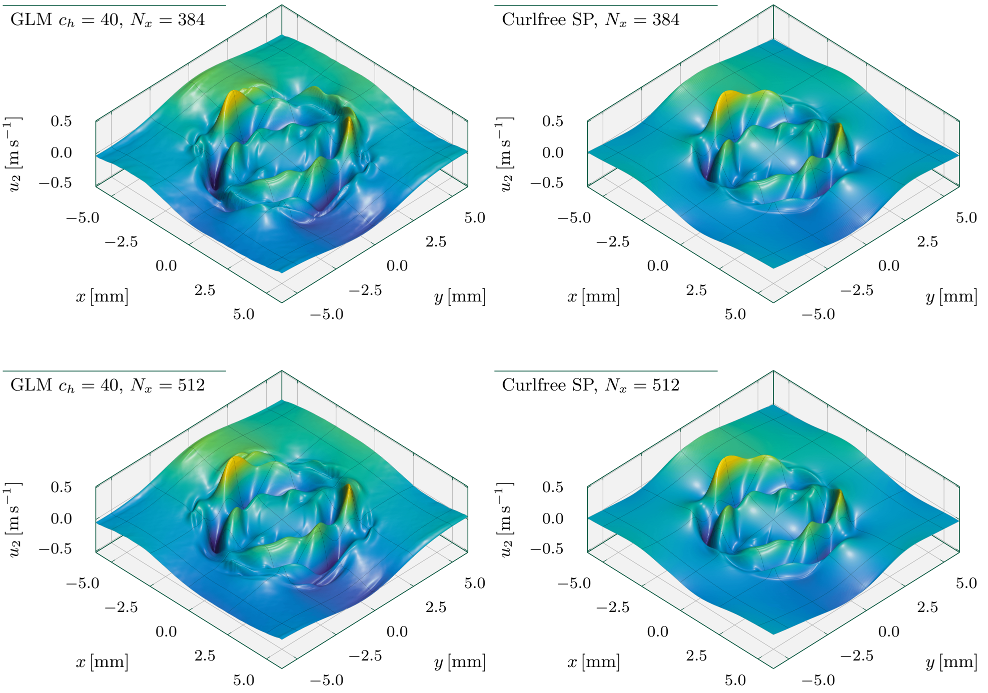


Figure 6.41. Three-dimensional view of the second component of the velocity field u_2 for the elliptical droplet oscillation problem at time $t = 6.91 \times 10^{-4}$ s (half of the first oscillation cycle). In the top row the uniform Cartesian mesh counts 384^2 elements, in the bottom row the mesh is refined to 512^2 elements. On the left, curl constraints are enforced with GLM curl-cleaning and the equations solved with the semi-implicit Finite Volume scheme presented in this work (without curlfree discretisation of the interface field), while on the right the results are computed with the curlfree structure-preserving semi-implicit method. The difference in the quality of the results obtained with the two different approaches is immediately apparent by comparing the *specular highlights* in the plots on the left column (from GLM curl cleaning, rough surface), with those on the right column (from the exactly curlfree scheme, smooth surface). Moreover, note that the smoothness of the curlfree solution is not due to numerical diffusion (peak heights are the same) and that the observations are the same for different meshes.

same semi-implicit framework. If involutions are not enforced (WH) curl errors grow uncontrollably and appear to decrease but the effect is only due to the complete breakdown of the simulation and the disappearance of *all* physical

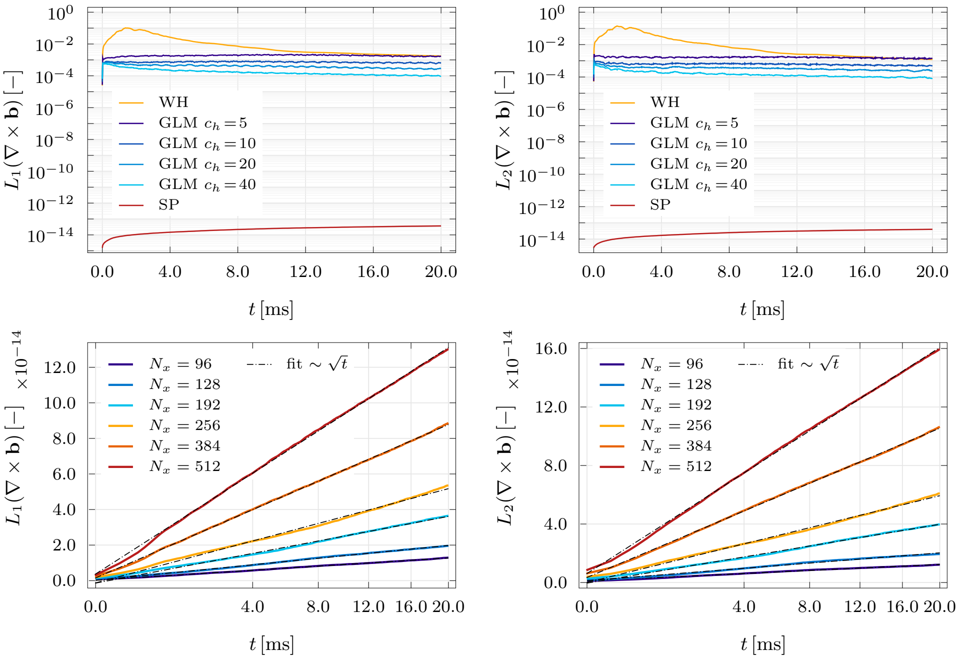


Figure 6.42. Time evolution of curl errors for the oscillating droplet test problem. In each panel of the top row, we compare the curl errors given by the GLM curl cleaning variant of the model with cleaning speed c_h ranging from $5\,\mathrm{m\,s^{-1}}$ to $40\,\mathrm{m\,s^{-1}}$, and those given by the curl-preserving semi-implicit method. The errors are computed in the L_1 (left panel) and the L_2 (right panel) norms, and are relative to a common uniform Cartesian mesh counting 192^2 elements. The curl-preserving semi-implicit method produces errors more than nine orders of magnitude smaller than those obtained with GLM curl cleaning in the same semi-implicit framework. If involutions are not enforced (WH) curl errors grow uncontrollably and appear to decrease but the effect is only due to the complete breakdown of the simulation. In the bottom row, we show, for a variety of uniform Cartesian meshes, that the errors of the curl-preserving semi-implicit Finite Volume scheme are due to statistical accumulation of roundoff errors: as expected from random-walk/Brownian [131, 140] processes, they grow in time proportionally to $\sqrt{t}$, and they grow faster for finer meshes which require more floating point operations in order to integrate the solution up to a given time.

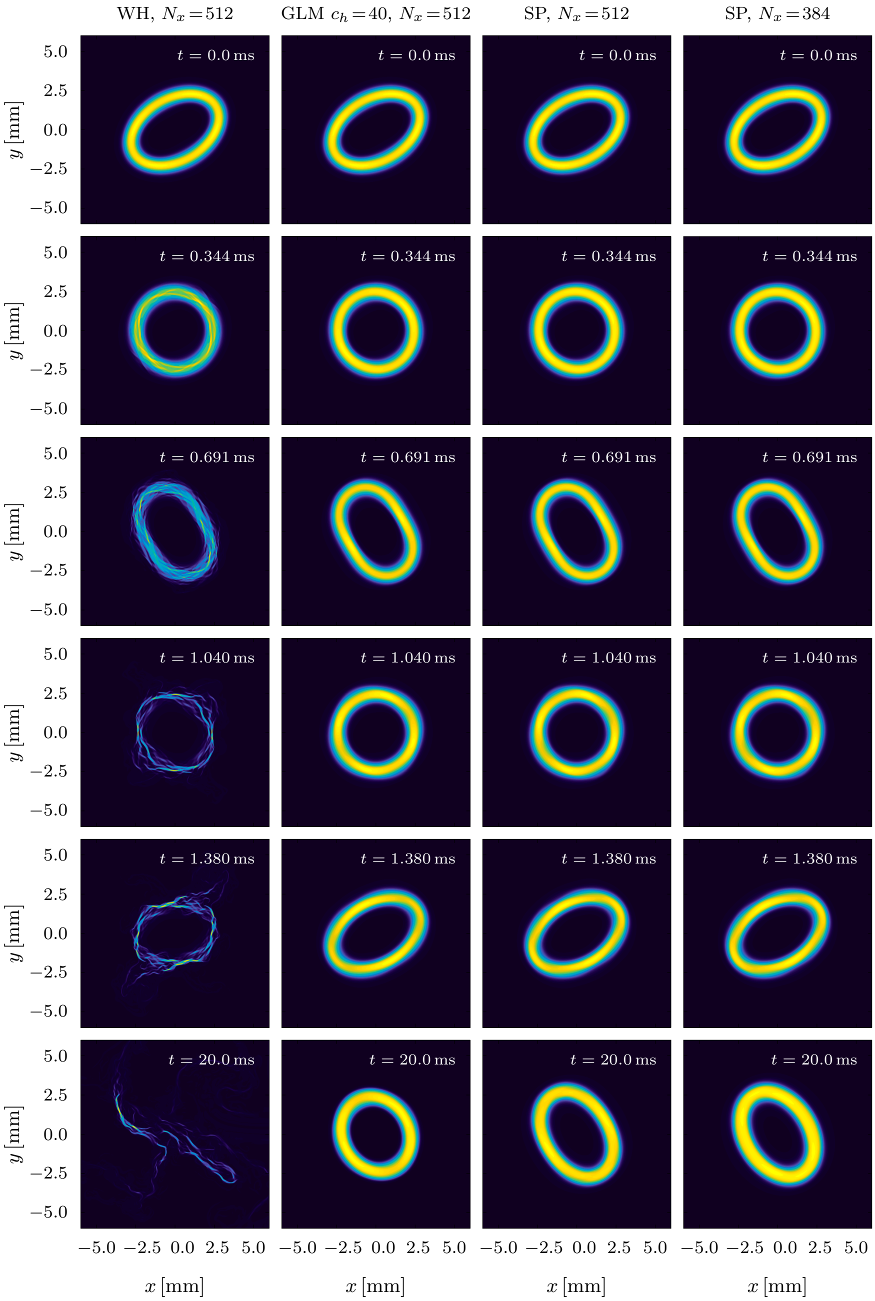


Figure 6.43. Snapshots of the interface energy $\sigma \, \|\mathbf{b}\|$ for the droplet oscillation problem with different mesh resolutions and for different schemes. WH is the unconstrained weakly hyperbolic model, GLM refers to the curl cleaning method, SP indicates the curl-free semi-implicit scheme.

flow features. Additionally, in order to argue that the curl errors produced by the exactly curlfree method originate from statistical accumulation of roundoff error, we plot their time evolution as a function of the square root of time $\sqrt{t}$ (essentially proportional to the square root of the number of timesteps as well). This scaling law shows that curl errors are growing in a perfectly linear fashion as a function of $\sqrt{t}$, as expected from a process of random-walk, or Brownian origin [131, 140]. Moreover the growth rate is higher on finer meshes, since more operations are required to integrate the solution up to a given time t. Nonetheless, this does not constitute an issue since the order of magnitude of the errors is that of machine-epsilon accumulated roundoff.

Finally, in Figure 6.43 we collect several snapshots of the global dynamics of the droplet oscillation problem, which clearly show that both the exactly curlfree methodology, and the GLM curl-cleaning method yield stable computations, counter to the unconstrained (weakly hyperbolic) variant of the model for hyperbolic surface tension, which catastrophically deteriorates within short times.

6.4.6 Binary droplet collision with high density ratio

Next, we continue our validation of the semi-implicit curl-preserving scheme for viscous two-phase flow with surface tension with an application to binary droplet collisions, motivated by [185, 263].

Two circular droplets are initialised according to the equilibrium solution derived in Section 2.2.6, with the centers of the droplets being

$$\begin{aligned} \mathbf{x}_{\mathrm{cl}} &= \left(-8\times10^{-4},\ -2\times10^{-4},\ 0\right)^{\mathsf{T}}, \\ \mathbf{x}_{\mathrm{cr}} &= \left(+8\times10^{-4},\ +2\times10^{-4},\ 0\right)^{\mathsf{T}}. \end{aligned} \tag{6.33}$$

Their radii are $R_{\mathrm{l}} = R_{\mathrm{r}} = 5 \times 10^{-4}$, and the interface thickness is $\varepsilon = 10^{-5}$. The initial volume fractions are $\alpha_1 = \alpha_{\min} = 10^{-4}$ for the gas phase and $\alpha_1 = \alpha_{\max} = 1 - 10^{-4}$ for the liquid phase. The atmospheric pressure is $p_{\mathrm{atm}} = 10^5$ and gravity effects are not present ($\mathbf{g} = \mathbf{0}$). The droplets are set on an off-center collision path by superimposing, to each droplet, a diffuse circular region in which the velocity field is $\mathbf{u} = (1,\ 0,\ 0)^{\mathsf{T}}$ for the left droplet and $\mathbf{u} = (-1,\ 0,\ 0)^{\mathsf{T}}$ for the right droplet, while the surrounding fluid is at rest. These circular regions are defined by the same smooth profile used for the droplets but with radius larger by a factor $k = 1.1$ with respect the droplet itself, and with interface thickness $\varepsilon = 5 \times 10^{-5}$. The strain relaxation times are $\tau_{\mathrm{w}} = 9.3750 \times 10^{-8}$ and $\tau_{\mathrm{a}} = 1.4064 \times 10^{-6}$ for the liquid and the gas respectively, which translates to kinematic viscosities $\nu_{\mathrm{w}} = 10^{-6}$ and

$\nu_\mathrm{a} = 1.5 \times 10^{-5}$. The densities are $\rho_1 = 10^3$ for the liquid phase and $\rho_2 = 1$ for the gas phase. The remaining material parameters are $\gamma_1 = 8.0$, $\gamma_2 = 1.4$, $\Pi_1 = 10^6$, $\Pi_2 = 0$ for the stiffened gas equation of state and $c_\mathrm{s} = 8$ for the mesoscale strain energy closure. Three separate numerical experiments are carried out with different values of the Weber number $\mathbb{W}\mathrm{e}$, defined by changing the surface tension coefficient from $\sigma = 0.2 \times 10^{-3}$ (corresponding to $\mathbb{W}\mathrm{e} \simeq 80$), to $\sigma = 0.1 \times 10^{-3}$ (i.e. $\mathbb{W}\mathrm{e} \simeq 160$), to $\sigma = 0.05 \times 10^{-3}$ ($\mathbb{W}\mathrm{e} \simeq 320$). The computational domain is $\Omega = [-4 \times 10^{-3},\ 4 \times 10^{-3}] \times [-3 \times 10^{-3},\ 3 \times 10^{-3}]$ and the mesh is of Cartesian type with 2048 and 1536 cells in the x and y directions respectively.

The qualitative behaviour is in agreement with experimental findings [4, 142, 301] about collision regimes in liquid droplets. In particular in Figure 6.46 one can clearly distinguish different separation modes taking place following the collision: at low Weber number ($\mathbb{W}\mathrm{e} \simeq 80$) the collision results in stretching and separation of the two droplets. At all values of Weber number, the effects of Rayleigh–Plateau instability are evident and in particular for the higher Weber

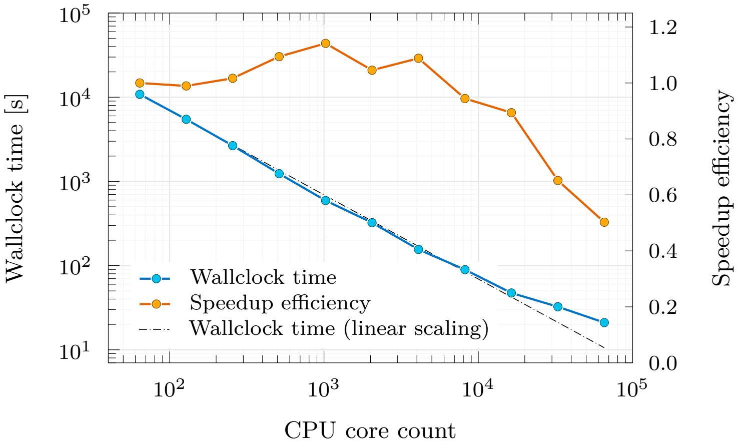


Figure 6.44. Strong scaling results (computational times and speedup efficiency) from 64 CPU cores to 65 536 CPU cores (512 nodes) of the HPE-Hawk supercomputer at HLRS in Stuttgart. The semi-implicit structure-preserving scheme for hyperbolic viscous two-phase flow with surface tension achieves excellent scaling performance (>∼ 95%) up to about 16k CPU cores, also thanks to cache effects mitigating the main bottleneck of the scheme which is the solution of the pressure system via matrix-free conjugate gradient method. On 65k CPU cores, the strong scaling efficiency drops to about 50%. The computational grid is composed of 33.5 million Cartesian cells.

numbers they result in the formation of multiple small secondary droplets. In the early stages of the simulations (Figure 6.45) one can also clearly see how sharp interfacial features are penalized by surface tension, which tends to reduce the curvature of interfaces, the more the stronger the capillarity forces with respect to convective effects. The same test, with $\mathbb{W}\mathrm{e} \simeq 80$ and a smaller domain $\Omega = [-3 \times 10^{-3},\ 3 \times 10^{-3}] \times [-1.5 \times 10^{-3},\ 1.5 \times 10^{-3}]$ is employed for a study of the strong scaling performance of the computational code. In this case the grid counts 8192 by 4096 cells and the simulations have been

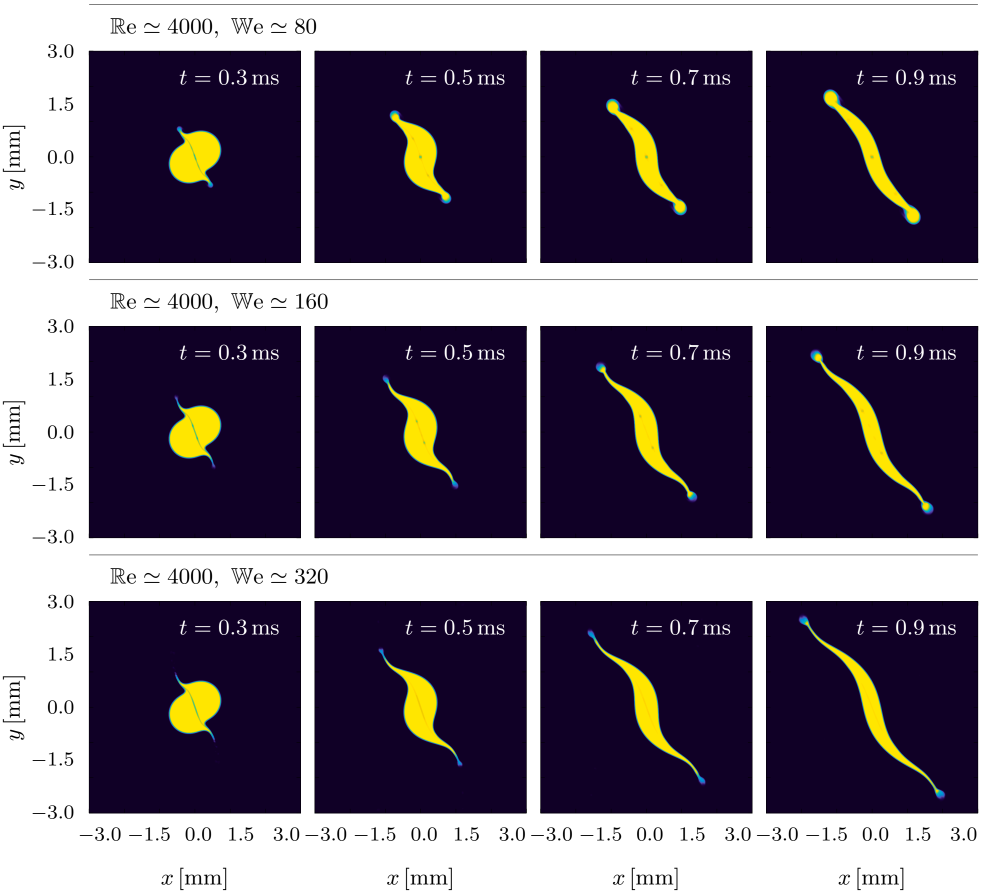


Figure 6.45. Early stages of the binary droplet collision simulation for three different values of the Weber number. The effects of surface tension penalising the formation of sharp features is evident: the lower the Weber number (corresponding to higher strength of capillarity forces), the stronger the tendency towards minimizing the interface curvature.

carried out with the aid of the HPE–Hawk supercomputer at the HLRS in Stuttgart, Germany, in order to test the scaling capabilities of our semi-implicit computational code on massively parallel distributed memory supercomputer architectures. The results of this latter test are summarised in Figure 6.44: the strong scaling tests starts from 64 CPU cores (half node) and extends up to 65 536 CPU cores (512 nodes) of the HPE–Hawk supercomputer at HLRS in Stuttgart. Our semi-implicit scheme achieves excellent scaling performance

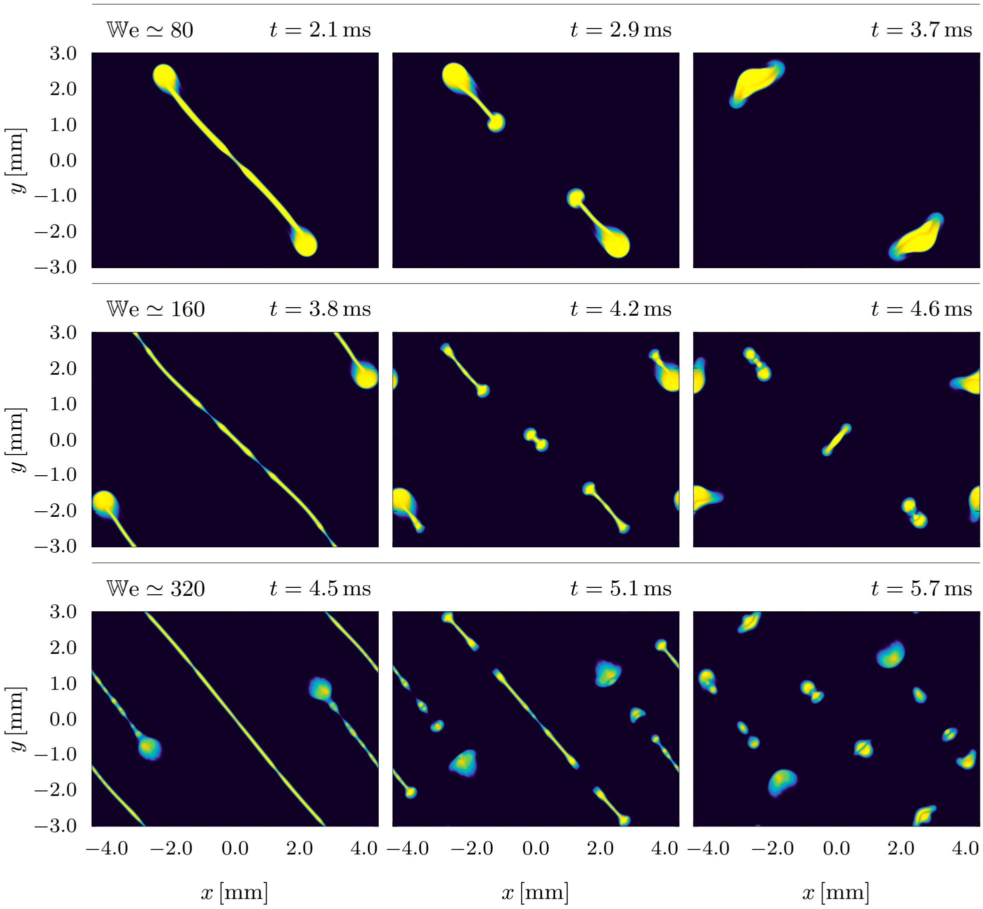


Figure 6.46. Late stages of the binary droplet collision simulation for three different values of the Weber number. At low Weber number ($\mathbb{W}\mathrm{e} \simeq 80$) the collision results in stretching and separation of the two droplets. At all values of Weber number the effects of Rayleigh–Plateau instability are evident and in particular for the higher Weber numbers they result in the formation of multiple small secondary droplets.

(more than $\sim 95\%$) up to about 16k CPU cores, also thanks to cache effects mitigating the main bottleneck of the scheme which is the solution of the pressure system via matrix-free conjugate gradient method and only on 65k CPU cores, we see the speedup efficiency starting to drop significantly to about 50%.

6.4.7 Multiphase Rayleigh–Taylor instability

Finally, we put all elements of the scheme to the test at the same time, by simulating a low Mach, genuinely two-phase, Rayleigh–Taylor instability with viscosity and surface tension. The setup follows [223], but with the notable modification that, instead of initialising a single fluid with two different densities (one above a horizontal material interface, one below), we define two separate density fields, each one with *constant* phase densities $\rho_1 = \rho_\mathrm{t}$ and $\rho_2 = \rho_\mathrm{b}$, then distinguishing the two fluids by means of a jump from $\alpha_1 = \alpha_\mathrm{min} = 10^{-4}$ to $\alpha_1 = \alpha_\mathrm{max} = 1 - 10^{-4}$ in the volume fraction field. This renders the problem much more challenging because near-vacuum states of one of the two phases are introduced almost throughout the computational domain. The curved material interface location is

$$y_I = 0.5 + 0.01\,\cos\left(6\,\pi\,x\right) \tag{6.34}$$

and we impose the transition between the two states by means of a smooth switch function

$$s = \frac{1}{2} + \frac{1}{2}\,\mathrm{erf}\left(\frac{y - y_I}{\delta}\right), \tag{6.35}$$

with interface thickness $\delta = \max(0.004,\ 6\,\Delta x)$. This is intended to suppress spurious instabilities that would be triggered by inaccurate representation of the initial condition on a discrete Cartesian grid (stairstepping), thus allowing only the physical instabilities to develop. The volume fraction field is then $\alpha_1 = s\,\alpha_\mathrm{min} + (1-s)\,\alpha_\mathrm{max}$, while the top and bottom pressures are

$$\begin{aligned} p_\mathrm{t} &= 1 + \rho_\mathrm{t}\,\mathbf{g}\cdot\hat{\mathbf{e}}_y\ (1-y)\,, \\ p_\mathrm{b} &= 1 + 0.6\,\rho_\mathrm{t}\,\mathbf{g}\cdot\hat{\mathbf{e}}_y + \rho_\mathrm{b}\ (1/2 - y)\,, \end{aligned} \tag{6.36}$$

and, just like the volume fraction α_1, the pressure field is given by $p = s\,p_\mathrm{t} + (1-s)\,p_\mathrm{b}$. We initially set $\mathbf{A} = \mathbf{I}$, $\mathbf{u} = \mathbf{0}$, and the remaining parameters common to all simulations carried out with regard to this test problem are $\rho_\mathrm{t} = 2$, $\rho_\mathrm{b} = 1$, the gravity vector is $\mathbf{g} = (0,\ -0.1,\ 0)^\intercal$, and finally $\mathbf{b}$ is initialised as a compatible discrete gradient of an auxiliary colour function field s_c given by $s_c = 1/2 + \mathrm{erf}\left[(y - y_I)/(2\,\delta)\right]/2$. With this setup, we carry out a parametric study of the behaviour of the instability.

First, in order to verify mesh convergence of the solution algorithm, we set the Reynolds number $\mathbb{R}\mathrm{e} \simeq 2000$ (which translates to $c_s = 0.3$ and $\tau = 2 \times 10^{-3}$) and $\mathbb{W}\mathrm{e} \simeq \infty$ (i.e. we neglect surface tension), and we carry out two simulation on two different meshes, one composed of 1024×3072 elements, and one of 2048×6144 elements, and show, in Figure 6.47, that indeed the structure of the solution does not depend on mesh effects. Furthermore, again in Figure 6.47, we also show that the method is robust with respect to the choice of the scaling

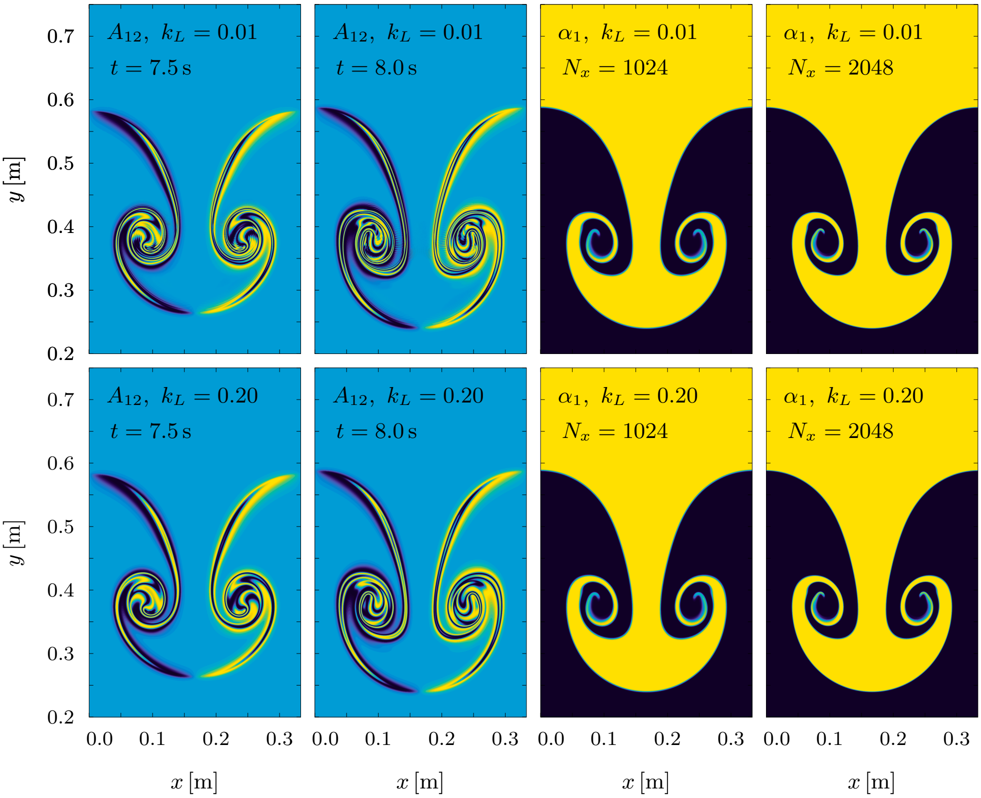


Figure 6.47. Mesh convergence test for the Rayleigh–Taylor instability problem: the rightmost panels show that mesh convergence has been achieved with a uniform Cartesian mesh counting 1024 by 3072 cells, since no new flow features appear if the mesh resolution is doubled. In the left panels we show the effects of choosing a different reduction coefficient k_L for the compatible vector Laplacian diffusion operator applied to the distortion matrix $\mathbf{A}$. By comparing the top and bottom rows one can see that increasing k_L has a visible effect on the finer features of the distortion field, but this does not translate to comparably visible effects in the volume fraction contours on the right.

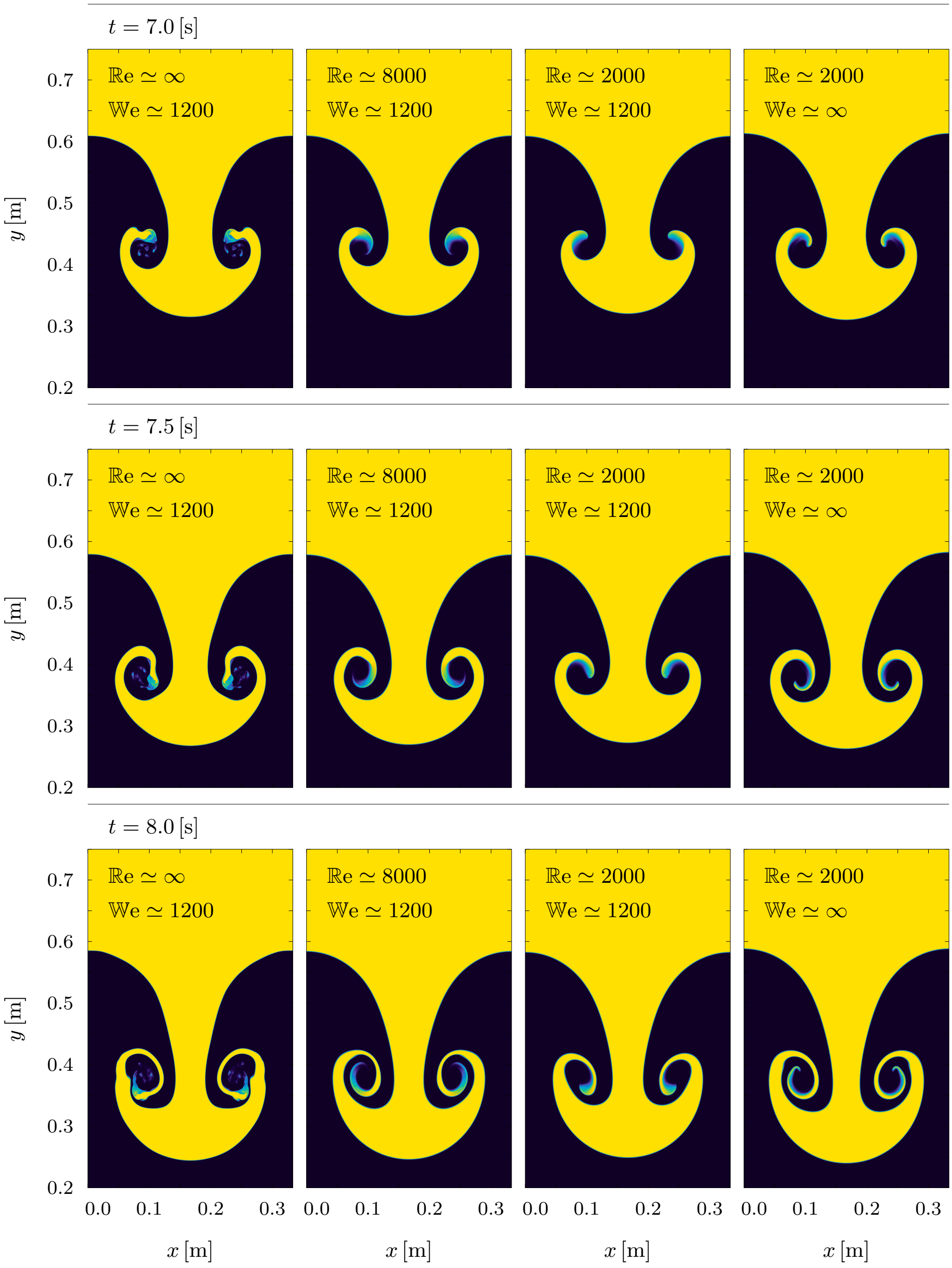


Figure 6.48. Volume fraction contours for the Rayleigh–Taylor instability problem with varying viscosity and surface tension at different times.

factor for the compatible numerical viscosity k_L: by varying such a factor from $k_L = 0.01$ to $k_L = 0.2$ we see that some of the fine scale structures in the distortion field $\mathbf{A}$ are lost to numerical dissipation. However, this does not translate into a visible effect in the shape of the interface separating the two fluids. This can be explained by the fact that the rotational component $\mathbf{R}$ of the distortion field $\mathbf{A}$ is significantly affected by numerical viscosity, but instead this is not the case for the stress $\boldsymbol{\sigma}_{\mathrm{s}} = -\rho\, c_{\mathrm{s}}^2\, \mathbf{G}\, \mathrm{dev}\, \mathbf{G}$, which does *not* depend on these rotations, thus leading to the same global dynamics for both choices of the numerical viscosity coefficient k_L.

In Figure 6.48 we report a variety of snapshots for this test problem at different times between $t = 7\,\mathrm{s}$ and $t = 8\,\mathrm{s}$, at different values of the Weber number $\mathbb{W}\mathrm{e} = \rho\, U^2\, L/\sigma$ and at different values of the Reynolds number $\mathbb{R}\mathrm{e} = U\, L/\nu$.

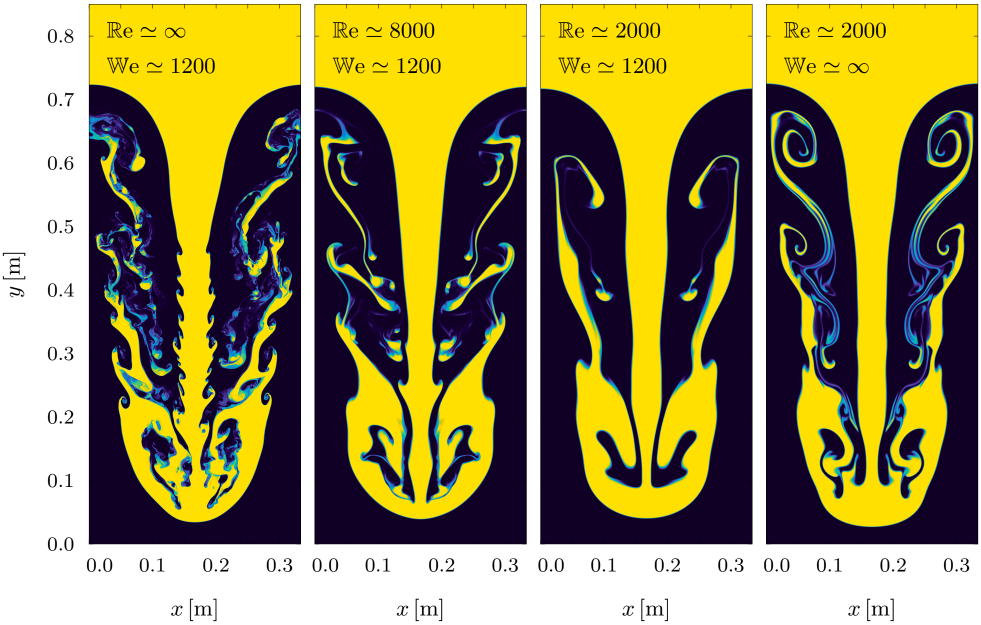


Figure 6.49. Development of the Rayleigh–Taylor instability at time $t = 12.5\,\mathrm{s}$, with varying viscosity and surface tension. From left to right: the first panel shows the solution with Weber number $\mathbb{W}\mathrm{e} \simeq 1200$ in inviscid flow; the second panel shows how the solution changes if mild viscosity is introduced ($\mathbb{R}\mathrm{e} \simeq 8000$); the third panels is relative to a simulation with increased viscosity ($\mathbb{R}\mathrm{e} \simeq 2000$); the fourth and last panel depicts the results of a simulation of viscous flow without surface tension. The curl-preserving semi-implicit scheme with grid size 4096 by 12 288 has been employed for all simulations.

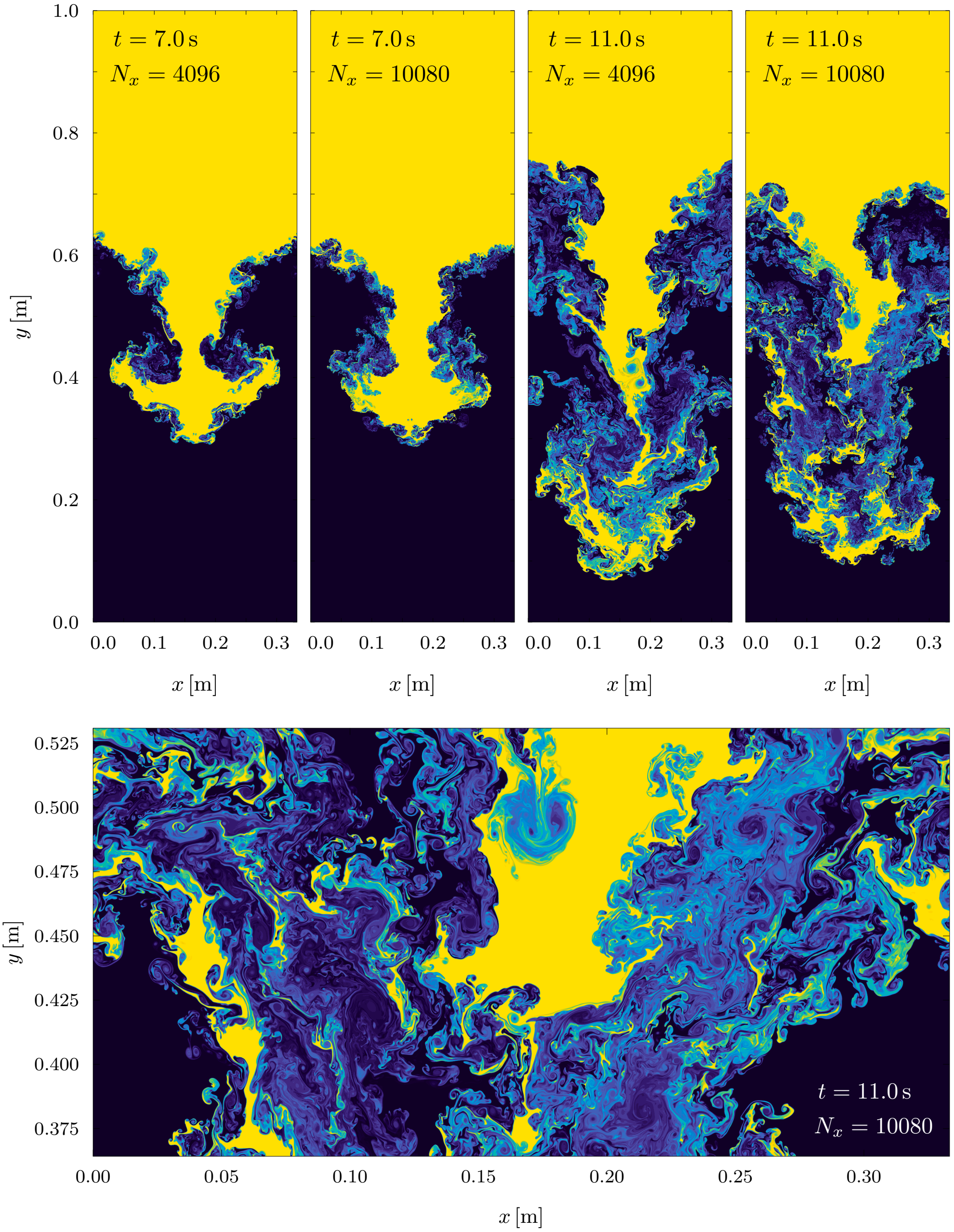


Figure 6.50. Volume fraction plots for the inviscid two-phase Rayleigh–Taylor instability solved with the Structure Preserving Semi-Implicit Finite Volume scheme applied to the unified model of continuum mechanics in the inviscid limit ($\tau = 10^{-14}$), at different mesh resolutions ($4096 \times 12\,288$ and $10\,080 \times 30\,240$).

These runs employ a uniform Cartesian grid counting 4096 by 12 288 elements. With $\mathbb{W}\mathrm{e} \simeq 1200$ we indicate simulations carried out with $\sigma = 2 \times 10^{-5}$, and $\mathbb{W}\mathrm{e} \simeq \infty$ corresponds to $\sigma = 0$. With $\mathbb{R}\mathrm{e} \simeq 2000$ we indicate simulations carried out with $c_\mathrm{s} = 0.3$ and $\tau = 2 \times 10^{-3}$, and the label $\mathbb{R}\mathrm{e} \simeq 8000$ corresponds to $c_\mathrm{s} = 0.3$ and $\tau = 5 \times 10^{-4}$, while the inviscid limit $\mathbb{R}\mathrm{e} \simeq \infty$ is given by $c_\mathrm{s} = 0.3$ and $\tau = 10^{-14}$. The results show the stabilising effects of surface tension and viscosity and the characteristic morphology they regulate. See [247] for striking experimental results on the Richtmyer–Meshkov instability, featuring similar flow structures.

In Figure 6.49, we plot the results of the same battery of simulations at a later time $t = 12.5\,\mathrm{s}$. Again, the distinctive features determined by the presence of viscosity and surface tension, as well as Rayleigh–Plateau secondary instabilities, can be clearly identified.

In a final set of runs we study the $\mathbb{R}\mathrm{e} \to \infty$, $\mathbb{W}\mathrm{e} \to \infty$ limit of the governing equations, by setting $\tau = 10^{-14}$, $\sigma = 0$, $c_\mathrm{s} = 0.1$. With this test we intend to verify the applicability of our MPI-parallel computational code to large scale simulations, its resolution properties, and its robustness in turbulent multiphase flows. We report, in Figure 6.50, the results of a large scale simulation of turbulent multiphase flow, run on 32k CPU cores of the HPE–Hawk supercomputer at the HLRS in Stuttgart, with a grid counting 10 080 by 30 240 uniform Cartesian elements. For comparison, in the same Figure are also plotted the results of the same setup on a coarser grid of 4096 by 12 288 cells.

7 Conclusions and outlook

The development of new numerical methods and that of novel mathematical models based on partial differential equations are strongly tied together: the mathematical model object of this work not only did present several challenges from the standpoint of numerical analysis, but also has delivered insight about the strategies that one might adopt for finding more robust, more efficient, more accurate solutions.

In particular, the study of the structure of the cobasis equation has been originally motivated by the need to overcome the complexities found in integrating the very nonlinear and multiscale processes regulating strain relaxation. Despite this purely practical motivation, it has proven of great value for developing an understanding of the underlying physical (and geometrical) mechanisms of strain generation and relaxation. In the same way, involution constraints are not only an obstacle to be dealt with, but rather they are a marker of the geometrical origin of some of the governing equations, which is what ultimately confers such an extensive generality to the unified model of continuum mechanics.

Thus one of the main findings of this work is that getting acquainted with the governing equations, their structure, their variants, their solutions, is an effective strategy for developing new numerical methods.

On a more practical note, the research carried out in this work spans several different topics: (*i*) the extension of the GLM hyperbolic divergence cleaning to the curl involutions present in the equations of hyperbolic surface tension part of unified model of continuum mechanics and the solution of these equations by means of high order ADER Discontinuous Galerkin and Finite Volume schemes; (*ii*) the development of robust semi-analytical and exponential-type integrators for finite-rate stiff relaxation processes in Baer–Nunziato compressible multiphase flows and (*iii*) for the strain relaxation equations of the unified mode of continuum mechanics; (*iv*) the diffuse interface modelling of crack propagation, material damage, and fatigue; (*v*) the generation of high quality moving meshes for Arbitrary-Lagrangian-Eulerian ADER Discontinuous Galerkin and Finite Volume on polygonal grids, (*vi*) the construction and implementation of efficient semi-implicit structure-preserving schemes for compressible multiphase flows with applicability in massively parallel distributed memory supercomputers.

In congruity with the statement given in the first paragraph of this chapter, not a single piece of work among those listed here has been carried out without yielding relevant ideas for at least another one.

Future work concerning the unified model of continuum mechanics developed in this thesis will proceed in at least four distinct directions.

Semi-implicit schemes. First, our semi-implicit curl-preserving scheme currently eliminates the timestep restrictions due to acoustic waves and can solve very efficiently and accurately fluid flows with low Mach number. On the other hand, with regard to the timestep restriction due to shear waves, the scheme behaves like an explicit one, and must satisfy a CFL condition based not only on convective and capillarity effects, but also on shear waves. A natural solution would be solving implicitly a second wave equation for the coupled strain/momentum system and hence further lifting this timestep restriction, leaving only capillarity and convection to be integrated explicitly. Again regarding semi-implicit discretisations, while carrying out the work presented in this thesis, it became apparent that a very effective strategy for achieving uniform accuracy in space and time will be the use of Runge–Kutta IMEX schemes [9, 35–38, 204, 250] for time integration.

Revisiting Baer–Nunziato semi-analytic solvers. Second, in the context of relaxation processes in Baer–Nunziato flows, only solutions based on the splitting approach [305, 332] have been considered in this work. In the latter stages of developments of the semi-analytical time integration techniques here introduced, it became apparent that significant improvements in the quality of the results can be obtained by restoring coupling between source terms and hyperbolic solution operators. Testing whether or not the same improvements can be ported back from the unified model of continuum mechanics to the Baer–Nunziato system will be verified in the future.

Extensions of the theory. Third, many more physical processes, not investigated in this work, can be found in the structure of the governing equations of SHTC systems. One notable direction in which the unified model of continuum mechanics can be extended is motivated by the expansion of the degrees of freedom of the cobasis $\mathbf{A}$ observed in the fluid regime, where the involution constraint $\nabla \mathbf{A} - \nabla \mathbf{A}^{\intercal} = \mathbf{0}$ is replaced by a set of evolutive equation for the curls of the cobasis vectors. Recall that adaptation of the Godunov–Romenski model of Eulerian hyperelasticity to describe also fluids is achieved by adding an appropriate relaxation source on the right hand side of the cobasis equation, which in turn originated as a geometrical compatibility condition. Then, the addition of the source term gives access to an expanded space of possible configurations for $\mathbf{A}$, since the matrix is no longer obtainable as the gradient of the Eulerian-to-Lagrangian coordinate mapping of physical space, i.e. $\mathbf{A} \neq \partial \mathbf{X} / \partial \mathbf{x}$.

We expect that such a procedure can be repeated for the newly originated curl matrix $\boldsymbol{\Omega}_{\mathbf{A}} = \nabla \mathbf{A} - \nabla \mathbf{A}^{\mathsf{T}}$ (called torsion in [258]), again transforming the constraint $\boldsymbol{\Omega}_{\mathbf{A}} = \mathbf{0}$ in an evolutive equation for $\boldsymbol{\Omega}_{\mathbf{A}}$, which, just like the rows of $\mathbf{A}$, will be treated as a state variable and evolved in time accordingly, rather than computed as a gradient.

We anticipate that informations about the microstructure of continua can be encoded in these higher order tensors, with potential applications in turbulence modelling or in the study of optical, acoustic and thermal metamaterials.

Simplification and communication. Last but not least, some effort should be devoted to simplifying the numerical methods necessary to solve the equations of the unified model of continuum mechanics, as well as finding better ways to communicate how the model can be useful in practical applications and as a theoretical tool. Towards this goal, a first step consists in continuing to rewrite the PDEs in new forms, looking for the ones that are easier to handle and gradually digest both on paper and in numerical codes, finding limits and approximations, invariants, symmetries.

List of Figures

List of Tables

References

[1] R. Abgrall and S. Karni. Computations of Compressible Multifluids. *Journal of Computational Physics*, 169, 594–623, 2001.

[2] R. Abgrall. How to Prevent Pressure Oscillations in Multicomponent Flow Calculations: A Quasi Conservative Approach. *Journal of Computational Physics*, 125, 150–160, 1996.

[3] R. Abgrall, K. Lipnikov, N. Morgan, and S. Tokareva. Multidimensional Staggered Grid Residual Distribution Scheme for Lagrangian Hydrodynamics. *SIAM Journal on Scientific Computing*, 42, A343–A370, 2020.

[4] K. H. Al-Dirawi and A. E. Bayly. An experimental study of binary collisions of miscible droplets with non-identical viscosities. *Experiments in Fluids*, 61, 2020.

[5] A. H. Al-Mohy and N. J. Higham. A New Scaling and Squaring Algorithm for the Matrix Exponential. *SIAM Journal on Matrix Analysis and Applications*, 31, 970–989, 2009.

[6] A. H. Al-Mohy and N. J. Higham. Computing the Action of the Matrix Exponential, with an Application to Exponential Integrators. *SIAM Journal on Scientific Computing*, 33, 488–511, 2011.

[7] D. Anderson, G. McFadden, and A. Wheeler. Diffuse-interface methods in fluid mechanics. *Annual Review of Fluid Mechanics*, 30, cited By 1456, 139–165, 1998.

[8] D. Arnold, R. Falk, and R. Winther. Finite element exterior calculus, homological techniques, and applications. *Acta Numerica*, 15, 1–155, 2006.

[9] U. M. Ascher, S. J. Ruuth, and R. J. Spiteri. Implicit-explicit Runge-Kutta methods for time-dependent partial differential equations. *Applied Numerical Mathematics*, 25, Special Issue on Time Integration, 151–167, 1997.

[10] M. R. Baer and J. W. Nunziato. A two-phase mixture theory for the deflagration-to-detonation transition (DDT) in reactive granular materials. *International Journal of Multiphase Flow*, 12, 861–889, 1986.

[11] D. Balsara. Divergence-free adaptive mesh refinement for magnetohydrodynamics. *Journal of Computational Physics*, 174, 614–648, 2001.

[12] D. Balsara. Multidimensional HLLE Riemann solver: Application to Euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 229, 1970–1993, 2010.

[13] D. Balsara. Multidimensional Riemann Problem with Self-Similar Internal Structure - Part I - Application to Hyperbolic Conservation Laws on Structured Meshes. *Journal of Computational Physics*, 277, 163–200, 2014.

[14] D. Balsara. Three dimensional HLL Riemann solver for conservation laws on structured meshes; Application to Euler and magnetohydrodynamic flows. *Journal of Computational Physics*, 295, 1–23, 2015.

[15] D. Balsara and M. Dumbser. Divergence-free MHD on unstructured meshes using high order finite volume schemes based on multidimensional Riemann solvers. *Journal of Computational Physics*, 299, 687–715, 2015.

[16] D. Balsara and M. Dumbser. Multidimensional Riemann Problem with Self-Similar Internal Structure - Part II - Application to Hyperbolic Conservation Laws on Unstructured Meshes. *Journal of Computational Physics*, 287, 269–292, 2015.

[17] D. Balsara, M. Dumbser, and R. Abgrall. Multidimensional HLLC Riemann Solver for Unstructured Meshes - With Application to Euler and MHD Flows. *Journal of Computational Physics*, 261, 172–208, 2014.

[18] D. Balsara, S. Garain, and C. Shu. An efficient class of WENO schemes with adaptive order. *Journal of Computational Physics*, 326, 780–804, 2016.

[19] D. Balsara, S. Garain, A. Taflove, and G. Montecinos. Computational electrodynamics in material media with constraint-preservation, multidimensional Riemann solvers and sub-cell resolution – Part II, higher order FVTD schemes. *Journal of Computational Physics*, 354, 613–645, 2018.

[20] D. Balsara and J. Kim. A subluminal relativistic magnetohydrodynamics scheme with ADER-WENO predictor and multidimensional Riemann solver-based corrector. *Journal of Computational Physics*, 312, 357–384, 2016.

[21] D. Balsara and D. Spicer. A Staggered Mesh Algorithm Using High Order Godunov Fluxes to Ensure Solenoidal Magnetic Fields in Magnetohydrodynamic Simulations. *Journal of Computational Physics*, 149, 270–292, 1999.

[22] T. J. Barth and P. O. Frederickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. *AIAA paper no. 90-0013, 28th Aerospace Sciences Meeting January*, 1990.

[23] P. Barton and E. Romenski. On Computational Modelling Of Strain-Hardening Material Dynamics. *Communications in Computational Physics*, 11, 1525–1546, 2012.

[24] P. T. Barton. An interface-capturing Godunov method for the simulation of compressible solid-fluid problems. *Journal of Computational Physics*, 390, 25–50, 2019.

[25] C. Bassi, L. Bonaventura, S. Busto, and M. Dumbser. A hyperbolic reformulation of the Serre-Green-Naghdi model for general bottom topographies. *Computers & Fluids*, 212, 104716, 2020.

[26] R. Becker. Stosswelle und Detonation. *Physik*, 8, 321, 1923.

[27] J. B. Bell, P. Coletta, and H. M. Glaz. A Second-Order Projection Method for the Incompressible Navier-Stokes Equations. *Journal of Computational Physics*, 85, 257–283, 1989.

[28] M. Berndt, J. Breil, S. Galera, M. Kucharik, P. Maire, and M. Shashkov. Two–step hybrid conservative remapping for multimaterial arbitrary Lagrangian–Eulerian methods. *Journal of Computational Physics*, 230, 6664–6687, 2011.

[29] R. Berry, R. Saurel, F. Petitpas, E. Daniel, O. Le Métayer, and S. Gavrilyuk. Progress in the Development of Compressible , Multiphase Flow Modeling Capability for Nuclear Reactor Flow Applications. *Idaho National Laboratory*, 2008.

[30] J. Besseling. A thermodynamic approach to rheology. In: *Irreversible Aspects of Continuum Mechanics and Transfer of Physical Characteristics in Moving Fluids.* Ed. by H. Parkus and L. Sedov. IUTAM Symposia. Springer Vienna, 1968. 16–53.

[31] W. Bo and M. Shashkov. Adaptive reconnection-based arbitrary Lagrangian Eulerian method. *Journal of Computational Physics*, 299, 902–939, 2015.

[32] W. Boettinger, J. Warren, C. Beckermann, and A. Karma. Phase-field simulation of solidification. *Annual Review of Materials Science*, 32, cited By 1211, 163–194, 2002.

[33] C. Bona, T. Ledvinka, C. Palenzuela, and M. Zacek. General-covariant evolution formalism for numerical relativity. *Physical Review D*, 67, 104005, 2003.

[34] C. Bona, T. Ledvinka, C. Palenzuela, and M. Zacek. Symmetry-breaking mechanism for the Z4 general-covariant evolution system. *Physical Review D*, 69, 064036, 2004.

[35] S. Boscarino, L. Pareschi, and G. Russo. Implicit-Explicit Runge–Kutta Schemes for Hyperbolic Systems and Kinetic Equations in the Diffusion Limit. *SIAM Journal on Scientific Computing*, 35, A22–A51, 2013.

[36] S. Boscarino. Error Analysis of IMEX Runge–Kutta Methods Derived from Differential-Algebraic Systems. *SIAM Journal on Numerical Analysis*, 45, 1600–1621, 2007.

[37] S. Boscarino. On an accurate third order implicit-explicit Runge–Kutta method for stiff problems. *Applied Numerical Mathematics*, 59, 1515–1528, 2009.

[38] S. Boscarino and G. Russo. On a Class of Uniformly Accurate IMEX Runge–Kutta Schemes and Applications to Hyperbolic Systems with Relaxation. *SIAM Journal on Scientific Computing*, 31, 1926–1945, 2009.

[39] W. Boscheri. An efficient high order direct ALE ADER finite volume scheme with a posteriori limiting for hydrodynamics and magnetohydrodynamics. *International Journal for Numerical Methods in Fluids*, 84, 76–106, 2017.

[40] W. Boscheri and M. Dumbser. Arbitrary–Lagrangian–Eulerian One–Step WENO Finite Volume Schemes on Unstructured Triangular Meshes. *Communications in Computational Physics*, 14, 1174–1206, 2013.

[41] W. Boscheri and M. Dumbser. A direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume scheme on unstructured tetrahedral meshes for conservative and non-conservative hyperbolic systems in 3D. *Journal of Computational Physics*, 275, 484 –523, 2014.

[42] W. Boscheri and M. Dumbser. High order accurate direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume schemes on moving curvilinear unstructured meshes. *Computers and Fluids*, 136, 48–66, 2016.

[43] W. Boscheri, M. Dumbser, M. Ioriatti, I. Peshkov, and E. Romenski. A structure-preserving staggered semi-implicit finite volume scheme for continuum mechanics. *Journal of Computational Physics*, 424, 109866, 2021.

[44] W. Boscheri, M. Dumbser, and M. Righetti. A semi-implicit scheme for 3D free surface flows with high-order velocity reconstruction on unstructured Voronoi meshes. *International journal for numerical methods in fluids*, 72, 607–631, 2013.

[45] W. Boscheri, E. Gaburro, and M. Dumbser. Continuous finite element subgrid basis functions for Discontinuous Galerkin schemes on unstructured polygonal Voronoi meshes. *Communications in Computational Physics (submitted)*, 2022.

[46] W. Boscheri. High order direct Arbitrary-Lagrangian–Eulerian (ALE) finite volume schemes for hyperbolic systems on unstructured meshes. *Archives of Computational Methods in Engineering*, 24, 751–801, 2017.

[47] W. Boscheri, S. Chiocchetti, and I. Peshkov. A cell-centered implicit-explicit Lagrangian scheme for a unified model of nonlinear continuum mechanics on unstructured meshes. *Journal of Computational Physics*, 451, 110852, 2022.

[48] W. Boscheri, R. Loubère, and P.-H. Maire. A 3D cell-centered ADER MOOD Finite Volume method for solving updated Lagrangian hyperelasticity on unstructured grids. *Journal of Computational Physics*, 449, 110779, 2022.

[49] W. Boscheri, G. R. Pisaturo, and M. Righetti. High-order divergence-free velocity reconstruction for free surface flows on unstructured Voronoi meshes. *International Journal for Numerical Methods in Fluids*, 90, 296–321, 2019.

[50] F. Bossen and P. Heckbert. A Pliant Method for Anisotropic Mesh Generation. *Proceedings of the 5th International Meshing Roundtable*, 1998.

[51] B. Bourdin, G. A. Francfort, and J.-J. Marigo. *The Variational Approach to Fracture*. Dordrecht: Springer, 2008.

[52] J. Brackbill, D. Kothe, and Z. Ca. A Continuum Method for Modeling Surface Tension. *Journal of Computational Physics*, 100, 1992.

[53] A. de Brauer, A. Iollo, and T. Milcent. A Cartesian Scheme for Compressible Multimaterial Hyperelastic Models with Plasticity. *Communications in Computational Physics*, 22, 1362–1384, 2017.

[54] V. V. Brazhkin, Y. D. Fomin, A. G. Lyapin, and K. Ryzhov V. N .and Trachenko. Two liquid states of matter: A dynamic line on a phase diagram. *Physical Review E*, 85, 031203, 2012.

[55] J. Breil, S. Galera, and P. Maire. Multi-material ALE computation in inertial confinement fusion code CHIC. *Computers and Fluids*, 46, 161–167, 2011.

[56] P. Buchmüller, J. Dreher, and C. Helzel. Finite volume WENO methods for hyperbolic conservation laws on Cartesian grids with adaptive mesh refinement. *Applied Mathematics and Computation*, 272, 460–478, 2016.

[57] P. Buchmüller and C. Helzel. Improved accuracy of high-order WENO finite volume methods on Cartesian grids. *Journal of Scientific Computing*, 61, 343–368, 2014.

[58] S. Busto, M. Dumbser, I. Peshkov, and E. Romenski. On thermodynamically compatible finite volume schemes for continuum mechanics. *SIAM Journal on Scientific Computing*, (in press), 2021.

[59] S. Busto, S. Chiocchetti, M. Dumbser, E. Gaburro, and I. Peshkov. High Order ADER Schemes for Continuum Mechanics. *Frontiers in Physics*, 8, 2020.

[60] S. Busto, M. Dumbser, C. Escalante, N. Favrie, and S. Gavrilyuk. On High Order ADER Discontinuous Galerkin Schemes for First Order Hyperbolic Reformulations of Nonlinear Dispersive Systems. *Journal of Scientific Computing*, 87, 48, 2021.

[61] S. Busto, M. Dumbser, S. Gavrilyuk, and K. Ivanova. On Thermodynamically Compatible Finite Volume Methods and Path-Conservative ADER Discontinuous Galerkin Schemes for Turbulent Shallow Water Flows. *Journal of Scientific Computing*, 88, 28, 2021.

[62] S. Busto, M. Dumbser, and L. Río-Martín. Staggered Semi-Implicit Hybrid Finite Volume/Finite Element Schemes for Turbulent and Non-Newtonian Flows. *Mathematics*, 9, 2021.

[63] J. C. Butcher. Implicit Runge–Kutta processes. *Mathematics of Computation*, 18, 50–64, 1964.

[64] J. C. Butcher. *The Numerical Analysis of Ordinary Differential Equations: Runge-Kutta and General Linear Methods.* New York, NY, USA: Wiley-Interscience, 1987.

[65] J. Cantarella, D. DeTurck, and H. Gluck. Vector calculus and the topology of domains in 3-space. *The American Mathematical Monthly*, 109, 409–442, 2002.

[66] T. Carney, N. Morgan, S. K. Sambasivan, and M. Shashkov. A cell–centered Lagrangian Godunov–like method for solid dynamics. *Computers & Fluids*, 83, 33–47, 2013.

[67] G. Carré, S. D. Pino, B. Després, and E. Labourasse. A cell-centered Lagrangian hydrodynamics scheme on general unstructured meshes in arbitrary dimension. *Journal of Computational Physics*, 228, 5160 –5183, 2009.

[68] M. Castro, J. Gallardo, and C. Parés. High Order Finite Volume Schemes Based on Reconstruction of States for Solving Hyperbolic Systems with Nonconservative Products. Applications to Shallow-Water Systems. *Mathematics of Computation*, 75, 1103–1134, 2006.

[69] M. Castro, J. Gallardo, J. López, and C. Parés. Well-balanced high order extensions of Godunov's method for semilinear balance laws. *SIAM Journal of Numerical Analysis*, 46, 1012–1039, 2008.

[70] V. Casulli. Semi-implicit finite difference methods for the two–dimensional shallow water equations. *Journal of Computational Physics*, 86, 56–74, 1990.

[71] V. Casulli. A semi-implicit finite difference method for non-hydrostatic free-surface flows. *International Journal for Numerical Methods in Fluids*, 30, 425–440, 1999.

[72] V. Casulli. A high-resolution wetting and drying algorithm for free-surface hydrodynamics. *International Journal for Numerical Methods in Fluids*, 60, 391–408, 2009.

[73] V. Casulli and R. Cheng. Semi-implicit finite difference methods for three-dimensional shallow water flow. *International Journal of Numerical Methods in Fluids*, 15, 629–648, 1992.

[74] V. Casulli and D. Greenspan. Pressure method for the numerical solution of transient, compressible fluid flows. *International Journal for Numerical Methods in Fluids*, 4, 1001–1012, 1984.

[75] V. Casulli and G. S. Stelling. Semi-implicit subgrid modelling of three-dimensional free-surface flows. *International Journal for Numerical Methods in Fluids*, 67, 441–449, 2011.

[76] V. Casulli and P. Zanolli. Semi-implicit numerical modeling of nonhydrostatic free-surface flows for environmental problems. *Mathematical and Computer Modelling*, 36, 1131–1149, 2002.

[77] V. Casulli and P. Zanolli. A nested Newton–type algorithm for finite volume methods solving Richards' equation in mixed form. *SIAM Journal on Scientific Computing*, 32, 2255–2273, 2009.

[78] V. Casulli and P. Zanolli. Iterative solutions of mildly nonlinear systems. *Journal of Computational and Applied Mathematics*, 236, 3937–3947, 2012.

[79] C. Cattaneo. Sulla conduzione del calore. *Atti sem. Mat. Fis. Univ. Modena*, 3, 1948.

[80] J. Certaine. The solution of ordinary differential equations with large time constants. *Mathematical Methods for Digital Computers*, 128–132, 1960.

[81] J. Cheng and C. Shu. A high order ENO conservative Lagrangian type scheme for the compressible Euler equations. *Journal of Computational Physics*, 227, 1567–1596, 2007.

[82] J. Cheng and C. Shu. A cell-centered Lagrangian scheme with the preservation of symmetry and conservation properties for compressible fluid flows in two–dimensional cylindrical geometry. *Journal of Computational Physics*, 229, 7191–7206, 2010.

[83] J. Cheng and C. Shu. Improvement on spherical symmetry in two–dimensional cylindrical coordinates for a class of control volume Lagrangian schemes. *Communications in Computational Physics*, 11, 1144–1168, 2012.

[84] L. Chew. Guaranteed-quality triangular meshes. *Technical Report 89-983*, 1989.

[85] S. Chiocchetti and M. Dumbser. An exactly curl-free staggered semi-implicit finite volume scheme for a first order hyperbolic model of viscous flow with surface tension. *Journal of Scientific Computing*, (in preparation), 2022.

[86] S. Chiocchetti and C. Müller. A Solver for Stiff Finite-Rate Relaxation in Baer-Nunziato Two-Phase Flow Models. *Fluid Mechanics and its Applications*, 121, 31–44, 2020.

[87] S. Chiocchetti, I. Peshkov, S. Gavrilyuk, and M. Dumbser. High order ADER schemes and GLM curl cleaning for a first order hyperbolic formulation of compressible flow with surface tension. *Journal of Computational Physics*, 426, 109898, 2021.

[88] S. Clain, S. Diot, and R. Loubère. A high-order finite volume method for systems of conservation laws – Multi-dimensional Optimal Order Detection (MOOD). *Journal of Computational Physics*, 230, 4028–4050, 2011.

[89] B. Cockburn and C. Shu. Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems. *Journal of Scientific Computing*, 16, 173–261, 2001.

[90] B. Cockburn and C.-W. Shu. TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework. *Mathematics of Computation*, 52, 411–435, 1989.

[91] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzgleichungen der mathematische Physik. *Mathematische Annalen*, 100, 32–74, 1928.

[92] I. Cravero, G. Puppo, M. Semplice, and G. Visconti. CWENO: uniformly accurate reconstructions for balance laws. *Mathematics of Computation*, 87, 1689–1719, 2018.

[93] C. M. Dafermos. Quasilinear hyperbolic systems with involutions. *Archive for Rational Mechanics and Analysis*, 94, 373–389, 1986.

[94] A. Dedner, F. Kemm, D. Kröner, C. Munz, T. Schnitzer, and M. Wesenberg. Hyperbolic Divergence Cleaning for the MHD Equations. *Journal of Computational Physics*, 175, 645–673, 2002.

[95] D. Derigs, A. Winters, G. Gassner, S. Walch, and M. Bohm. Ideal GLM-MHD: About the entropy consistent nine-wave magnetic field divergence diminishing ideal magnetohydrodynamics equations. *Journal of Computational Physics*, 364, 420–467, 2018.

[96] B. Després. *Numerical Methods for Eulerian and Lagrangian Conservation Laws.* Frontiers in Mathematics. Springer International Publishing, 2017.

[97] B. Després and C. Mazeran. Lagrangian gas dynamics in two-dimensions and lagrangian systems. *Archive for Rational Mechanics and Analysis*, 178, 327–372, 2005.

[98] C. DeVore. Flux-corrected transport techniques for multidimensional compressible magnetohydrodynamics. *Journal of Computational Physics*, 92, 142–160, 1991.

[99] F. Dhaouadi, N. Favrie, and S. Gavrilyuk. Extended Lagrangian approach for the defocusing nonlinear Schrödinger equation. *Studies in Applied Mathematics*, 142, 336–358, 2019.

[100] S. Diot, S. Clain, and R. Loubère. Improved detection criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials. *Computers & Fluids*, 64, 43–63, 2012.

[101] S. Diot, R. Loubère, and S. Clain. The Multidimensional Optimal Order Detection method in the three dimensional case: very high order finite volume method for hyperbolic systems. *International Journal for Numerical Methods in Fluids*, 73, 362–392, 2013.

[102] M. Dumbser. Arbitrary high order PNPM schemes on unstructured meshes for the compressible Navier-Stokes equations. *Computers & Fluids*, 39, 60–76, 2010.

[103] M. Dumbser and D. S. Balsara. A New Efficient Formulation of the HLLEM Riemann Solver for General Conservative and Non-conservative Hyperbolic Systems. *Journal of Computational Physics*, 304, 275–319, 2016.

[104] M. Dumbser, D. Balsara, M. Tavelli, and F. Fambri. A divergence-free semi-implicit finite volume scheme for ideal, viscous and resistive magnetohydrodynamics. *International Journal for Numerical Methods in Fluids*, 89, 16–42, 2019.

[105] M. Dumbser, D. Balsara, E. Toro, and C. Munz. A Unified Framework for the Construction of One-step Finite Volume and Discontinuous Galerkin Schemes on Unstructured Meshes. *Journal of Computational Physics*, 227, 8209–8253, 2008.

[106] M. Dumbser and W. Boscheri. High-order unstructured Lagrangian one–step WENO finite volume schemes for non-conservative hyperbolic systems: Applications to compressible multi–phase flows. *Computers & Fluids*, 86, 405 –432, 2013.

[107] M. Dumbser, W. Boscheri, M. Semplice, and G. Russo. Central weighted ENO schemes for hyperbolic conservation laws on fixed and moving unstructured meshes. *SIAM Journal on Scientific Computing*, 39, A2564–A2591, 2017.

[108] M. Dumbser, M. Castro, C. Parés, and E. Toro. ADER Schemes on Unstructured Meshes for Non-Conservative Hyperbolic Systems: Applications to Geophysical Flows. *Computers and Fluids*, 38, 1731–1748, 2009.

[109] M. Dumbser and V. Casulli. A conservative, weakly nonlinear semi-implicit finite volume method for the compressible Navier-Stokes equations with general equation of state. *Applied Mathematics and Computation*, 272, 479–497, 2016.

[110] M. Dumbser, S. Chiocchetti, and I. Peshkov. In: On Numerical Methods for Hyperbolic PDE with Curl Involutions. G. V. Demidenko, E. Romenski, E. Toro, and M. Dumbser, eds. 125–134. Cham: Springer International Publishing, 2020.

[111] M. Dumbser, S. Chiocchetti, and I. Peshkov. On Numerical Methods for Hyperbolic PDE with Curl Involutions. In: *Continuum Mechanics, Applied Mathematics and Scientific Computing: Godunov's Legacy*. Cham: Springer International Publishing, 2020. 125–134.

[112] M. Dumbser, C. Enaux, and E. Toro. Finite Volume Schemes of Very High Order of Accuracy for Stiff Hyperbolic Balance Laws. *Journal of Computational Physics*, 227, 3971–4001, 2008.

[113] M. Dumbser, F. Fambri, E. Gaburro, and A. Reinarz. On GLM curl cleaning for a first order reduction of the CCZ4 formulation of the Einstein field equations. *Journal of Computational Physics*, 404, 109088, 2020.

[114] M. Dumbser, F. Fambri, M. Tavelli, M. Bader, and T. Weinzierl. Efficient Implementation of ADER Discontinuous Galerkin Schemes for a Scalable Hyperbolic PDE Engine. *Axioms*, 7, 63, 2018.

[115] M. Dumbser, F. Guercilena, S. Koeppel, L. Rezzolla, and O. Zanotti. Conformal and covariant Z4 formulation of the Einstein equations: Strongly hyperbolic first-order reduction and solution with discontinuous Galerkin schemes. *Physical Review D*, 97, 084053, 2018.

[116] M. Dumbser, A. Hidalgo, M. Castro, C. Parés, and E. Toro. FORCE schemes on unstructured meshes II: Non-conservative hyperbolic systems. *Computer Methods in Applied Mechanics and Engineering*, 199, 625–647, 2010.

[117] M. Dumbser and M. Käser. Arbitrary high order non-oscillatory Finite Volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221, 693–723, 2007.

[118] M. Dumbser, M. Käser, V. Titarev, and E. Toro. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *Journal of Computational Physics*, 226, 204–243, 2007.

[119] M. Dumbser and C. Munz. Building Blocks for Arbitrary High Order Discontinuous Galerkin Schemes. *International Journal of Applied Mathematics and Computer Science*, 17, 297–310, 2006.

[120] M. Dumbser, I. Peshkov, E. Romenski, and O. Zanotti. High order ADER schemes for a unified first order hyperbolic formulation of continuum mechanics: Viscous heat-conducting fluids and elastic solids. *Journal of Computational Physics*, 314, 824–862, 2016.

[121] M. Dumbser, I. Peshkov, E. Romenski, and O. Zanotti. High order ADER schemes for a unified first order hyperbolic formulation of Newtonian continuum mechanics coupled with electro-dynamics. *Journal of Computational Physics*, 348, 298–342, 2017.

[122] M. Dumbser, T. Schwartzkopff, and C. Munz. Arbitrary high order finite volume schemes for linear wave propagation. In: *Computational Science and High Performance Computing II*. Ed. by E. Krause, Y. Shokin, M. Resch, and N. Shokina. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006. 129–144.

[123] M. Dumbser and E. Toro. A Simple Extension of the Osher Riemann Solver to Non-Conservative Hyperbolic Systems. *Journal of Scientific Computing*, 48, 70–88, 2011.

[124] M. Dumbser, O. Zanotti, A. Hidalgo, and D. Balsara. ADER-WENO finite volume schemes with space-time adaptive mesh refinement. *Journal of Computational Physics*, 248, 257–286, 2013.

[125] M. Dumbser, O. Zanotti, R. Loubère, and S. Diot. A Posteriori Subcell Limiting of the Discontinuous Galerkin Finite Element Method for Hyperbolic Conservation Laws. *Journal of Computational Physics*, 2014.

[126] M. Dumbser and W. Boscheri. High-order unstructured Lagrangian one-step WENO finite volume schemes for non-conservative hyperbolic systems: Applications to compressible multi-phase flows. *Computers & Fluids*, 86, 405–432, 2013.

[127] M. Dumbser, W. Boscheri, M. Semplice, and G. Russo. Central weighted ENO schemes for hyperbolic conservation laws on fixed and moving unstructured meshes. *SIAM Journal on Scientific Computing*, 39, A2564–A2591, 2017.

[128] K. Duraisamy, G. Iaccarino, and H. Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51, cited By 373, 357–377, 2019.

[129] B. Einfeldt, P. Roe, C. Munz, and B. Sjogreen. On Godunov-type methods near low densities. *Journal of Computational Physics*, 92, 273–295, 1991.

[130] B. Einfeldt. On Godunov-Type Methods for Gas Dynamics. *SIAM Journal on Numerical Analysis*, 25, 294–318, 1988.

[131] A. Einstein. Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der Physik*, 322, 549–560, 1905.

[132] A. Einstein. Zur Elektrodynamik bewegter Körper. *Annalen der Physik*, 322, 891–921, 1905.

[133] H. Erten and A. Üngör. Triangulations with Locally Optimal Steiner Points. *Proceedings of the Fifth Eurographics Symposium on Geometry Processing*, 143–152, 2007.

[134] S. Fasoulas, C.-D. Munz, M. Pfeiffer, J. Beyer, T. Binder, S. Copplestone, A. Mirza, P. Nizenkov, P. Ortwein, and W. Reschke. Combining particle-in-cell and direct simulation Monte Carlo for the simulation of reactive plasma flows. *Physics of Fluids*, 31, cited By 14, 2019.

[135] N. Favrie and S. Gavrilyuk. Mathematical and numerical model for nonlinear viscoplasticity. *Philosophical Transactions of the Royal Society A*, 369, 2864–2880, 2011.

[136] N. Favrie and S. Gavrilyuk. A rapid numerical method for solving Serre-Green-Naghdi equations describing long free surface gravity waves. *Nonlinearity*, 30, 2718–2736, 2017.

[137] N. Favrie, S. Gavrilyuk, and R. Saurel. Solid–fluid diffuse interface model in cases of extreme deformations. *Journal of computational physics*, 228, 6037–6077, 2009.

[138] N. Favrie, S. Gavrilyuk, and R. Saurel. Solid–fluid diffuse interface model in cases of extreme deformations. *Journal of Computational Physics*, 228, 6037–6077, 2009.

[139] N. Favrie and S. Gavrilyuk. Diffuse interface model for compressible fluid-Compressible elastic-plastic solid interaction. *Journal of Computational Physics*, 231, 2695–2723, 2012.

[140] R. P. Feynman. *The Feynman lectures on physics.* Reading, Mass.: Addison-Wesley Pub. Co., 1963.

[141] D. A. Field. Laplacian smoothing and Delaunay triangulations. *Communications in Applied Numerical Methods*, 4, 709–712, 1988.

[142] G. Finotello, R. F. Kooiman, J. T. Padding, K. A. Buist, A. Jongsma, F. Innings, and J. A. M. Kuipers. The dynamics of milk droplet–droplet collisions. *Experiments in Fluids*, 59, 2017.

[143] A. Forestier and S. Gavrilyuk. Criterion of hyperbolicity for non–conservative quasilinear systems admitting a partially convex conservation law. *Mathematical Methods in the Applied Sciences*, 34, 2148–2158, 2011.

[144] R. Fortenbach, C. Munz, E. Sonnnendrücker, and R. Klein. *Asymptotic Considerations on the Navier-Stokes-Equations in a Regime of Low Mach-Number for an Application in Aeroacoustics.* Tech. rep. Universität Stuttgart, IAG, 1999.

[145] G. Francfort and J.-J. Marigo. Revisiting brittle fracture as an energy minimization problem. *Journal of the Mechanics and Physics of Solids*, 46, 1319–1342, 1998.

[146] D. Fyfe, E. Oran, and M. Fritts. Surface tension and viscosity with lagrangian hydrodynamics on a triangular mesh. *Journal of Computational Physics*, 76, 349–384, 1988.

[147] A.-A. Gabriel, D. Li, S. Chiocchetti, M. Tavelli, I. Peshkov, E. Romenski, and M. Dumbser. A unified first-order hyperbolic model for nonlinear dynamic rupture processes in diffuse fracture zones. *Philosophical Transactions of the Royal Society A*, 379, 2021.

[148] E. Gaburro. A unified framework for the solution of hyperbolic PDE systems using high order direct Arbitrary-Lagrangian–Eulerian schemes on moving unstructured meshes with topology change. *Archives of Computational Methods in Engineering*, 28, 1249–1321, 2021.

[149] E. Gaburro, W. Boscheri, S. Chiocchetti, C. Klingenberg, V. Springel, and M. Dumbser. High order direct Arbitrary–Lagrangian–Eulerian schemes on moving Voronoi meshes with topology changes. *Journal of Computational Physics*, 407, 109167, 2020.

[150] E. Gaburro, M. J. Castro, and M. Dumbser. Well-balanced Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming meshes for the Euler equations of gas dynamics with gravity. *Monthly Notices of the Royal Astronomical Society*, 477, 2251–2275, 2018.

[151] E. Gaburro and M. Dumbser. A Posteriori Subcell Finite Volume Limiter for General PNPM Schemes: Applications from Gasdynamics to Relativistic Magnetohydrodynamics. *Journal of Scientific Computing*, 86, 1–41, 2021.

[152] E. Gaburro, M. Dumbser, and M. J. Castro. Direct Arbitrary-Lagrangian-Eulerian finite volume schemes on moving nonconforming unstructured meshes. *Computers and Fluids*, 159, 254–275, 2017.

[153] T. Gardiner and J. Stone. An unsplit Godunov method for ideal MHD via constrained transport. *Journal of Computational Physics*, 205, 509–539, 2005.

[154] G. Gassner, M. Dumbser, F. Hindenlang, and C. Munz. Explicit one-step time discretizations for discontinuous Galerkin and finite volume schemes based on local predictors. *Journal of Computational Physics*, 230, Special issue High Order Methods for CFD Problems, 4232–4247, 2011.

[155] G. Gassner, F. Lörcher, and C. D. Munz. A discontinuous Galerkin scheme based on a space-time expansion II. Viscous flow equations in multi dimensions. *Journal of Scientific Computing*, 34, 260–286, 2008.

[156] G. Gassner, F. Lörcher, and C. Munz. A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes. *Journal of Computational Physics*, 224, 1049–1063, 2007.

[157] G. J. Gassner, F. Lörcher, C.-D. Munz, and J. S. Hesthaven. Polymorphic nodal elements and their application in discontinuous Galerkin methods. *Journal of Computational Physics*, 228, 1573–1590, 2009.

[158] S. Gavrilyuk, K. Ivanova, and N. Favrie. Multi-dimensional shear shallow water flows: Problems and solutions. *Journal of Computational Physics*, 366, 252–280, 2018.

[159] S. Gavrilyuk, N. Favrie, and R. Saurel. Modelling wave dynamics of compressible elastic materials. *Journal of Computational Physics*, 227, 2941–2969, 2008.

[160] M. Germano, U. Piomelli, P. Moin, and W. Cabot. A dynamic subgrid-scale eddy viscosity model. *Physics of Fluids A*, 3, cited By 5489, 1760–1765, 1991.

[161] P. Gómez-Gálvez, P. Vicente-Munuera, A. Tagua, C. Forja, A. M. Castro, M. Letrán, A. Valencia-Expósito, C. Grima, M. Bermúdez-Gallardo, s. Serrano-Pérez-Higueras, F. Cavodeassi, S. Sotillos, M. D. Martín-Bermudo, A. Márquez, J. Buceta, and L. M. Escudero. Scutoids are a geometrical solution to three-dimensional packing of epithelia. *Nature Communications*, 9, 2960, 2018.

[162] S. K. Godunov. A Finite Difference Method for the Computation of Discontinuous Solutions of the Equations of Fluid Dynamics. *Matematicheskii Sbornik*, 47, 357–393, 1959.

[163] S. K. Godunov. An interesting class of quasilinear systems. *Dokl. Akad. Nauk SSSR*, 139(3), 521–523, 1961.

[164] S. K. Godunov. Symmetric form of the magnetohydrodynamic equation. *Numerical Methods for Mechanics of Continuum Medium*, 3, 26–34, 1972.

[165] S. K. Godunov. *Elements of mechanics of continuous media.* Nauka, 1978.

[166] S. K. Godunov, T. Y. Mikhaîlova, and E. Romenskiî. Systems of thermodynamically coordinated laws of conservation invariant under rotations. *Siberian Mathematical Journal*, 37, 690–705, 1996.

[167] S. K. Godunov and I. Peshkov. Thermodynamically consistent nonlinear model of elastoplastic Maxwell medium. *Computational Mathematics and Mathematical Physics*, 50(8), 1409–1426, 2010.

[168] S. K. Godunov and E. Romenski. Nonstationary equations of the nonlinear theory of elasticity in Euler coordinates. *Journal of Applied Mechanics and Technical Physics*, 13, 868–885, 1972.

[169] S. K. Godunov and E. Romenski. *Elements of mechanics of continuous media.* Nauchnaya Kniga, 1998.

[170] S. K. Godunov and E. I. Romenskii. *Elements of continuum mechanics and conservation laws.* Kluwer Academic/Plenum Publishers, 2003.

[171] S. Gottlieb and C.-W. Shu. Total Variation Diminishing Runge-Kutta Schemes. *Mathematics of Computation*, 67, 73–85, 1998.

[172] J.-L. Guermond, C. Kees, B. Popov, and E. Tovar. Hyperbolic relaxation technique for solving the dispersive Serre–Green–Naghdi equations with topography. *Journal of Computational Physics*, 450, 110809, 2022.

[173] D. Gueyffier, J. Li, A. Nadim, R. Scardovelli, and S. Zaleski. Volume-of-Fluid Interface Tracking with Smoothed Surface Stress Methods for Three-Dimensional Flows. *Journal of Computational Physics*, 152, 423–456, 1999.

[174] H. Haeri, K. Shahriar, M. F. Marji, and P. Moarefvand. Experimental and numerical study of crack propagation and coalescence in pre-cracked rock-like disks. *International Journal of Rock Mechanics and Mining Sciences*, 67, 20–28, 2014.

[175] S. Hank, S. Gavrilyuk, N. Favrie, and J. Massoni. Impact simulation by an Eulerian model for interaction of multiple elastic-plastic solids and fluids. *International Journal of Impact Engineering*, 109, 104–111, 2017.

[176] A. Harten, P. Lax, and B. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25, 35–61, 1983.

[177] A. Hazra, P. Chandrashekar, and D. Balsara. Globally constraint-preserving FR/DG scheme for Maxwell's equations at all orders. *Journal of Computational Physics*, 394, 298–328, 2019.

[178] F. W. Hehl and Y. N. Obukhov. Elie Cartan's torsion in geometry and in field theory, an essay. *Annales de la Fondation Louis de Broglie*, 32, 157–194, 2007.

[179] H. Helmoltz. Über diskontinuierliche Füssigkeitsbewegungen. *Berliner Monatsberichte*, 1868.

[180] L. R. Herrmann. Laplacian-Isoparametric Grid Generation Scheme. In. 1976.

[181] A. Hidalgo and M. Dumbser. ADER schemes for nonlinear systems of stiff advection–diffusion–reaction equations. *Journal of Scientific Computing*, 48, 173–189, 2011.

[182] N. J. Higham. The Scaling and Squaring Method for the Matrix Exponential Revisited. *SIAM Journal on Matrix Analysis and Applications*, 26, 1179–1193, 2005.

[183] A. C. Hindmarsh. LSODE and LSODI, Two New Initial Value Ordinary Differential Equation Solvers. *SIGNUM Newsl.*, 15, 10–11, 1980.

[184] A. C. Hindmarsh. In: ODEPACK, A Systematized Collection of ODE Solvers. R. S. Stepleman, ed. 55–64. Amsterdam: North-Holland, 1983.

[185] H. Hinterbichler, C. Planchette, and G. Brenn. Ternary drop collisions. *Experiments in Fluids*, 56, 190/1–190/12, 2015.

[186] R. Hiptmair. Finite elements in computational electromagnetism. *Acta Numerica*, 11, 237–339, 2002.

[187] C. W. Hirt and B. D. Nichols. Volume of Fluid (VOF) Method for Dynamics of Free Boundaries. *Journal of Computational Physics*, 39, 201–225, 1981.

[188] T. Hitz, M. Heinen, J. Vrabec, and C.-D. Munz. Comparison of macro- and microscopic solutions of the Riemann problem I. Supercritical shock tube and expansion into vacuum. *Journal of Computational Physics*, 402, 109077, 2020.

[189] T. Hitz, S. Jöns, M. Heinen, J. Vrabec, and C.-D. Munz. Comparison of macro- and microscopic solutions of the Riemann problem II. Two-phase shock tube. *Journal of Computational Physics*, 429, 110027, 2021.

[190] M. Hochbruck and A. Ostermann. Exponential integrators. *Acta Numerica*, 19, cited By 594, 209–286, 2010.

[191] C. Hu and C. Shu. Weighted essentially non-oscillatory schemes on triangular meshes. *Journal of Computational Physics*, 150, 97–127, 1999.

[192] J. Hyman and M. Shashkov. Natural discretizations for the divergence, gradient, and curl on logically rectangular grids. *Computers and Mathematics with Applications*, 33, 81–104, 1997.

[193] D. Igra and K. Takayama. A study of shock wave loading on a cylindrical water column. *Reports of the Institute of Fluid Science*, 13, 19–36, 2002.

[194] H. Jackson and N. Nikiforakis. A numerical scheme for non-Newtonian fluids and plastic solids under the GPR model. *Journal of Computational Physics*, 387, 410–429, 2019.

[195] H. Jackson and N. Nikiforakis. A unified Eulerian framework for multi-material continuum mechanics. *Journal of Computational Physics*, 401, 109022, 2019.

[196] H. Jackson. On the eigenvalues of the ADER-WENO Galerkin predictor. *Journal of Computational Physics*, 333, 409–413, 2017.

[197] D. Jacqmin. Calculation of Two-Phase Navier-Stokes Flows Using Phase-Field Modeling. *Journal of Computational Physics*, 155, cited By 1034, 96–127, 1999.

[198] R. Jeltsch and M. Torrilhon. On curl–preserving finite volume discretizations for shallow water equations. *BIT Numerical Mathematics*, 46, S35–S53, 2006.

[199] G.-S. Jiang and C. W. Shu. Efficient Implementation of Weighted ENO Schemes. *Journal of Computational Physics*, 126, 202–228, 1996.

[200] J. Kamm and F. Timmes. On efficient generation of numerically robust Sedov solutions. *Technical Report LA-UR-07-2849,LANL*, 2007.

[201] A. K. Kapila, R. Menikoff, J. B. Bdzil, S. F. Son, and D. S. Stewart. Two-phase modelling of deflagration-to-detonation in granular materials: reduced equations. *Physics of Fluids*, 13, 3002–3024, 2001.

[202] M. Käser and A. Iske. ADER Schemes on adaptive triangular meshes for scalar conservation laws. *Journal of Computational Physics*, 205, 486–508, 2005.

[203] F. Kemm, E. Gaburro, F. Thein, and M. Dumbser. A simple diffuse interface approach for compressible flows around moving solids of arbitrary shape based on a reduced Baer–Nunziato model. *Computers & Fluids*, 204, 104536, 2020.

[204] C. A. Kennedy and M. H. Carpenter. Additive Runge–Kutta schemes for convection–diffusion–reaction equations. *Applied Numerical Mathematics*, 44, 139–181, 2003.

[205] S. Klainermann and A. Majda. Singular Limits of Quasilinear Hyperbolic Systems with Large Parameters and the Incompressible Limit of Compressible Fluid. *Comm. Pure Appl. Math.*, XXXIV, 481–524, 1981.

[206] S. Klainermann and A. Majda. Compressible and incompressible fluids. *Communications on Pure and Applied Mathematics*, 35, 629–651, 1982.

[207] R. Klein, N. Botta, T. Schneider, C. Munz, S.Roller, A. Meister, L. Hoffmann, and T. Sonar. Asymptotic adaptive methods for multi-scale problems in fluid mechanics. *Journal of Engineering Mathematics*, 39, 261–343, 2001.

[208] H. Kleinert. *Multivalued Fields.* World Scientific, 2008.

[209] B. Larouturou. How to preserve the mass fraction positive when computing compressible multi-component flows. *Journal of Computational Physics*, 5, 1991.

[210] C. L. Lawson. Transforming Triangulations. *Discrete Mathematics*, 1972.

[211] P. Lax and B. Wendroff. Systems of conservation laws. *Communications on Pure and Applied Mathematics*, 13, 217–237, 1960.

[212] B. van Leer. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *Journal of Computational Physics*, 32, 101–136, 1979.

[213] D. Levy, G. Puppo, and G. Russo. Central WENO schemes for hyperbolic systems of conservation laws. *ESAIM: Mathematical Modelling and Numerical Analysis*, 33, 547–571, 1999.

[214] D. Levy, G. Puppo, and G. Russo. A third order central WENO scheme for 2D conservation laws. *Applied Numerical Mathematics*, 33, 415–421, 2000.

[215] D. Levy, G. Puppo, and G. Russo. Compact central WENO schemes for multidimensional conservation laws. *SIAM Journal on Scientific Computing*, 22, 656–672, 2000.

[216] D. Levy, G. Puppo, and G. Russo. A fourth-order central WENO scheme for multidimensional hyperbolic systems of conservation laws. *SIAM Journal on Scientific Computing*, 24, 480–506, 2002.

[217] K. Lipnikov, G. Manzini, and M. Shashkov. Mimetic finite difference method. *Journal of Computational Physics*, 257, 1163–1227, 2014.

[218] W. Liu, J. Cheng, and C. Shu. High order conservative Lagrangian schemes with Lax–Wendroff type time discretization for the compressible Euler equations. *Journal of Computational Physics*, 228, 8872–8891, 2009.

[219] S. Lloyd. Least squares quantization in PCM. *IEEE transactions on information theory*, 28, 129–137, 1982.

[220] F. Lörcher, G. Gassner, and C. Munz. A discontinuous Galerkin scheme based on a space-time expansion. I. Inviscid compressible flow in one space dimension. *Journal of Scientific Computing*, 32, 175–199, 2007.

[221] R. Loubère, M. Dumbser, and S. Diot. A New Family of High Order Unstructured MOOD and ADER Finite Volume Schemes for Multidimensional Systems of Hyperbolic Conservation Laws. *Communications in Computational Physics*, 16, 718–763, 2014.

[222] R. Loubère, P. H. Maire, and M. Shashkov. ReALE: A Reconnection Arbitrary-Lagrangian–Eulerian method in cylindrical geometry. *Computers and Fluids*, 46, 59–69, 2011.

[223] R. Loubère, P. H. Maire, M. Shashkov, J. Breil, and S. Galera. ReALE: A reconnection-based arbitrary-Lagrangian–Eulerian method. *Journal of Computational Physics*, 229, 4724–4761, 2010.

[224] P.-H. Maire. A high-order cell-centered Lagrangian scheme for compressible fluid flows in two–dimensional cylindrical geometry. *Journal of Computational Physics*, 228, 6882–6915, 2009.

[225] P.-H. Maire. A unified sub-cell force-based discretization for cell-centered Lagrangian hydrodynamics on polygonal grids. *International Journal for Numerical Methods in Fluids*, 65, 1281–1294, 2011.

[226] P.-H. Maire and J. Breil. A second-order cell-centered Lagrangian scheme for two–dimensional compressible flow problems. *International Journal for Numerical Methods in Fluids*, 56, 1417–1423, 2007.

[227] P. Maire. A high-order cell-centered Lagrangian scheme for two–dimensional compressible fluid flows on unstructured meshes. *Journal of Computational Physics*, 228, 2391 –2425, 2009.

[228] P. Maire. A high-order one-step sub-cell force-based discretization for cell-centered Lagrangian hydrodynamics on polygonal grids. *Computers and Fluids*, 46(1), 341–347, 2011.

[229] P. Maire, R. Abgrall, J. Breil, and J. Ovadia. A cell-centered Lagrangian scheme for two–dimensional compressible flow problems. *SIAM Journal on Scientific Computing*, 29, 1781–1824, 2007.

[230] P. Maire and B. Nkonga. Multi-scale Godunov-type method for cell-centered discrete Lagrangian hydrodynamics. *Journal of Computational Physics*, 228, 799–821, 2009.

[231] G. Margolin, M. Shashkov, and P. Smolarkiewicz. A discrete operator calculus for finite difference approximations. *Computer Methods in Applied Mechanics and Engineering*, 187, 365–383, 2000.

[232] G. D. Maso, P. LeFloch, and F. Murat. Definition and weak stability of nonconservative products. *Journal de Mathématiques Pures et Appliquées*, 74, 483–548, 1995.

[233] P. J. Mason. Large-eddy simulation: A critical review of the technique. *Quarterly Journal of the Royal Meteorological Society*, 120, 1–26, 1994.

[234] J. C. Maxwell. *A treatise on electricity and magnetism.* Oxford: Clarendon Press, 1873.

[235] E. Meshkov. Instability of the interface of two gases accelerated by a shock wave. *Fluid Dynamics*, 4, 101–104, 1969.

[236] N. Moelans, B. Blanpain, and P. Wollants. An introduction to phase-field modeling of microstructure evolution. *Calphad: Computer Coupling of Phase Diagrams and Thermochemistry*, 32, cited By 583, 268–294, 2008.

[237] P. Monk. *Finite Element Methods for Maxwell's Equations.* Oxford University Press, Oxford, 2003.

[238] C. Munz, M. Dumbser, and S. Roller. Linearized acoustic perturbation equations for low Mach number flow with variable density and temperature. *Journal of Computational Physics*, 224, 352–364, 2007.

[239] C. Munz, R. Klein, S. Roller, and K. Geratz. The extension of incompressible flow solvers to the weakly compressible regime. *Computers and Fluids*, 2003.

[240] C. Munz, P. Omnes, R. Schneider, E. Sonnendrücker, and U. Voss. Divergence Correction Techniques for Maxwell Solvers Based on a Hyperbolic Model. *Journal of Computational Physics*, 161, 484–511, 2000.

[241] C. Munz, S. Roller, R. Klein, and K. Geratz. The extension of incompressible flow solvers to the weakly compressible regime. *Computers and Fluids*, 32, 173–196, 2003.

[242] S. Ndanou, N. Favrie, and S. Gavrilyuk. Criterion of hyperbolicity in hyperelasticity in the case of the stored energy in separable form. *Journal of Elasticity*, 115, 1–25, 2014.

[243] S. Ndanou, N. Favrie, and S. Gavrilyuk. Multi–solid and multi–fluid diffuse interface model: Applications to dynamic fracture and fragmentation. *Journal of Computational Physics*, 295, 523–555, 2015.

[244] J. Nédélec. Mixed finite elements in R3. *Numerische Mathematik*, 35, 315–341, 1980.

[245] J. Nédélec. A new family of mixed finite elements in R3. *Numerische Mathematik*, 50, 57–81, 1986.

[246] F. Nicoud and F. Ducros. Subgrid-scale stress modelling based on the square of the velocity gradient tensor. *Flow, Turbulence and Combustion*, 62, cited By 2295, 183–200, 1999.

[247] C. E. Niederhaus and J. W. Jacobs. Experimental study of the Richtmyer–Meshkov instability of incompressible fluids. *Journal of Fluid Mechanics*, 485, 243–277, 2003.

[248] R. Pakmor, V. Springel, A. Bauer, P. Mocz, D. J. Munoz, S. T. Ohlmann, K. Schaal, and C. Zhu. Improving the convergence properties of the moving-mesh code AREPO. *Monthly Notices of the Royal Astronomical Society*, 455, 1134–1143, 2015.

[249] C. Parés. Numerical methods for nonconservative hyperbolic systems: a theoretical framework. *SIAM Journal on Numerical Analysis*, 44, 300–321, 2006.

[250] L. Pareschi and G. Russo. Implicit-explicit runge-kutta schemes and applications to hyperbolic systems with relaxation. *Journal of Scientific Computing*, 25, 129–155, 2005.

[251] Pareschi, Lorenzo and Russo, Giovanni. An introduction to Monte Carlo method for the Boltzmann equation. *ESAIM: Proc.*, 10, 35–75, 2001.

[252] J. H. Park and C. D. Munz. Multiple pressure variables methods for fluid flow at all Mach numbers. *International Journal for Numerical Methods in Fluids*, 49, 905–931, 2005.

[253] M. Pelanti and K.-M. Shyue. A mixture-energy-consistent six-equation two-phase numerical model for fluids with interfaces, cavitation and evaporation waves. *Journal of Computational Physics*, 259, 331–357, 2014.

[254] P.-O. Persson and G. Strang. A simple mesh generator in MATLAB. *SIAM review*, 46, 329–345, 2004.

[255] I. Peshkov, W. Boscheri, R. Loubère, E. Romenski, and M. Dumbser. Theoretical and numerical comparison of hyperelastic and hypoelastic formulations for Eulerian non–linear elastoplasticity. *Journal of Computational Physics*, 387, 481–521, 2019.

[256] I. Peshkov, M. Pavelka, E. Romenski, and M. Grmela. Continuum mechanics and thermodynamics in the Hamilton and the Godunov-type formulations. *Continuum Mechanics and Thermodynamics*, 30, 1343–1378, 2018.

[257] I. Peshkov and E. Romenski. A hyperbolic model for viscous Newtonian flows. *Continuum Mechanics and Thermodynamics*, 28, 85–104, 2016.

[258] I. Peshkov, E. Romenski, and M. Dumbser. Continuum mechanics with torsion. *Continuum Mechanics and Thermodynamics*, 31, 1517–1541, 2019.

[259] I. Peshkov, M. Dumbser, W. Boscheri, E. Romenski, S. Chiocchetti, and M. Ioriatti. Simulation of non-Newtonian viscoplastic flows with a unified first order hyperbolic model and a structure-preserving semi-implicit scheme. *Computers & Fluids*, 224, 104963, 2021.

[260] V. Peshkov. Second Sound in Helium II. *Journal of Physics (USSR)*, 8, 381, 1944.

[261] R. Pidatella, G. Puppo, G. Russo, and P. Santagati. Semi–conservative finite volume schemes for conservation laws. *SIAM Journal on Scientific Computing*, 41, B576–B600, 2019.

[262] H. Pitsch. Large-eddy simulation of turbulent combustion. *Annual Review of Fluid Mechanics*, 38, cited By 842, 453–482, 2006.

[263] C. Planchette, H. Hinterbichler, M. Liu, D. Bothe, and G. Brenn. Colliding drops as coalescing and fragmenting liquid springs. *Journal of Fluid Mechanics*, 814, 277–300, 2017.

[264] D. A. Pope. An Exponential Method of Numerical Integration of Ordinary Differential Equations. *Communications of the ACM*, 6, 491–493, 1963.

[265] K. Powell. An Approximate Riemann Solver for Magnetohydrodynamics. In: *Upwind and High-Resolution Schemes*. Ed. by v. L. B. M.Y. and V. R. J. Berlin, Heidelberg: Springer Berlin Heidelberg, 1997. 570–583.

[266] K. Powell, P. Roe, T. Linde, T. Gombosi, and D. D. Zeeuw. A Solution-Adaptive Upwind Scheme for Ideal Magnetohydrodynamics. *Journal of Computational Physics*, 154, 284–309, 1999.

[267] J. Qiu, M. Dumbser, and C. Shu. The discontinuous Galerkin method with Lax-Wendroff type time discretizations. *Computer Methods in Applied Mechanics and Engineering*, 194, 4528–4543, 2005.

[268] D. H. R. Courant. *Methods of Mathematical Physics: Partial Differential Equations, Volume II.* Wiley-VCH Verlag GmbH & Co. KGaA, 1962.

[269] B. Re, C. Dobrzynski, and A. Guardone. An interpolation-free ALE scheme for unsteady inviscid flows computations with large boundary displacements over three-dimensional adaptive grids. *Journal of Computational Physics*, 340, 26–54, 2017.

[270] W. Reschke, T. Binder, J. Kleinert, A. Mirza, P. Nizenkov, M. Pfeiffer, S. Fasoulas, S. Copplestone, P. Ortwein, and C. Munz. Recent developments of DSMC within the reactive plasma flow solver PICLas. In. Vol. 1786. cited By 3. 2016.

[271] W. Reschke, M. Pfeiffer, and S. Fasoulas. Enabling Simulations of Droplets with the Direct Simulation Monte Carlo Method. *Fluid Mechanics and its Applications*, 121, cited By 0, 57–68, 2020.

[272] A. D. Resnyansky, E. I. Romensky, and N. K. Bourne. Constitutive modeling of fracture waves. *Journal of Applied Physics*, 93, 1537–1545, 2003.

[273] S. Rhebergen, O. Bokhove, and J. W. van der Vegt. Discontinuous Galerkin finite element methods for hyperbolic nonconservative partial differential equations. *Journal of Computational Physics*, 227, 1887–1922, 2008.

[274] R. D. Richtmyer. Taylor instability in shock acceleration of compressible fluids. *Communications on Pure and Applied Mathematics*, 13, 297–319, 1960.

[275] A. A. Rodriguez and A. Valli. Finite element potentials. *Applied Numerical Mathematics*, 95, 2–14, 2015.

[276] P. L. Roe and D. S. Balsara. Notes on the Eigensystem of Magnetohydrodynamics. *SIAM Journal on Applied Mathematics*, 56, 57–67, 1996.

[277] E. Romenski. Hyperbolic systems of thermodynamically compatible conservation laws in continuum mechanics. *Mathematical and computer modelling*, 28(10), 115–130, 1998.

[278] E. I. Romenskii. Deformation model for brittle materials and the structure of failure waves. *Journal of Applied Mechanics and Technical Physics*, 48, 437–444, 2007.

[279] E. I. Romensky. Thermodynamics and hyperbolic systems of balance laws in continuum mechanics. In: *Godunov Methods: Theory and Applications*. Ed. by E. Toro. New York: Springer US, 2001.

[280] G. Rossi, M. Dumbser, and A. Armanini. A well-balanced path conservative SPH scheme for nonconservative hyperbolic systems with applications to shallow water and multi-phase flows. *Computers and Fluids*, 154, 102–122, 2017.

[281] J. Ruppert. A new and simple algorithm for quality 2-dimensional mesh generation. *Proceedings of the 4th ACM-SIAM Symposium on Discrete Algorithms*, 83–92, 1993.

[282] R. Saurel and R. Abgrall. A multiphase Godunov method for compressible multifluid and multiphase flows. *Journal of Computational Physics*, 150, 425–467, 1999.

[283] R. Saurel, O. Le Métayer, J. Massoni, and S. Gavrilyuk. Shock jump relations for multiphase mixtures with stiff mechanical relaxation. *Shock Waves*, 16, 209–232, 2007.

[284] R. Saurel, F. Petitpas, and R. A. Berry. Simple and efficient relaxation methods for interfaces separating compressible fluids, cavitating flows and shocks in multiphase mixtures. *Journal of Computational Physics*, 228, 1678–1712, 2009.

[285] R. Saurel, F. Petitpas, and R. Abgrall. Modelling phase transition in metastable liquids: application to cavitating and flashing flows. *Journal of Fluid Mechanics*, 607, 313—350, 2008.

[286] K. Schmidmayer, F. Petitpas, E. Daniel, N. Favrie, and S. Gavrilyuk. A model and numerical method for compressible flows with capillary effects. *Journal of Computational Physics*, 334, 468–496, 2017.

[287] C. Schreiber, C. Kuhn, R. Müller, and T. Zohdi. A phase field modeling approach of cyclic fatigue crack growth. *International Journal of Fracture*, 225, 89–100, 2020.

[288] T. Schwartzkopff, M. Dumbser, and C. Munz. Fast High Order ADER Schemes for Linear Hyperbolic Equations. *Journal of Computational Physics*, 197, 532–539, 2004.

[289] T. Schwartzkopff, C. Munz, and E. Toro. ADER: A High Order Approach For Linear Hyperbolic Systems in 2D. *Journal of Scientific Computing*, 17, 231–240, 2002.

[290] T. Schwartzkopff, C. Munz, and E. Toro. ADER: A High-Order Approach for Linear Hyperbolic Systems in 2D. *Journal of Scientific Computing*, 17, 231–240, 2002.

[291] M. Semplice, A. Coco, and G. Russo. Adaptive Mesh Refinement for Hyperbolic Systems based on Third-Order Compact WENO Reconstruction. *Journal of Scientific Computing*, 66, 692–724, 2016.

[292] J. R. Shewchuk. Triangle: Engineering a 2d quality mesh generator and Delaunay triangulator. *Proceedings of the First Workshop on Applied Computational Geometry*, 124–133, 1996.

[293] J. R. Shewchuk. Adaptive Precision Floating-Point Arithmetic and Fast Robust Geometric Predicates. *Discrete & Computational Geometry*, 18, 305–363, 1997.

[294] J. R. Shewchuk. Delaunay refinement mesh generation. *Ph.D. thesis*, 1997.

[295] J. R. Shewchuk. Delaunay refinement algorithms for triangular mesh generation. *Computational Geometry*, 22, 21–74, 2002.

[296] K. Shimada and D. C. Gossard. Bubble Mesh: Automated Triangular Meshing of Non-manifold Geometry by Sphere Packing. In: *Proceedings of the Third ACM Symposium on Solid Modeling and Applications*. SMA ’95. ACM, 1995. 409–419.

[297] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77, 439–471, 1988.

[298] C. Shu. Total-Variation-Diminishing Time Discretizations. *SIAM Journal on Scientific and Statistical Computing*, 9, 1073–1084, 1988.

[299] C. Shu. Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws. *Technical Report, Institute for Computer Applications in Science and Engineering (ICASE)*, 1997.

[300] M. Shur, P. Spalart, M. Strelets, and A. Travin. A hybrid RANS-LES approach with delayed-DES and wall-modelled LES capabilities. *International Journal of Heat and Fluid Flow*, 29, cited By 1330, 1638–1649, 2008.

[301] M. Sommerfeld and L. Pasternak. Advances in modelling of binary droplet collision outcomes in Sprays: A review of available knowledge. *International Journal of Multiphase Flow*, 117, 182–205, 2019.

[302] M. Sonntag and C. Munz. Shock Capturing for discontinuous Galerkin methods using Finite Volume Subcells. In: *Finite Volumes for Complex Applications VII.* Ed. by J. Fuhrmann, M. Ohlberger, and C. Rohde. Springer, 2014. 945–953.

[303] P. Spalart, S. Deck, M. Shur, K. Squires, M. Strelets, and A. Travin. A new version of detached-eddy simulation, resistant to ambiguous grid densities. *Theoretical and Computational Fluid Dynamics*, 20, cited By 1832, 181–195, 2006.

[304] V. Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. *Monthly Notices of the Royal Astronomical Society (MNRAS)*, 401, 791–851, 2010.

[305] G. Strang. On the Construction and Comparison of Difference Schemes. *SIAM Journal on Numerical Analysis*, 5, 506–517, 1968.

[306] A. Stroud. *Approximate Calculation of Multiple Integrals.* Englewood Cliffs, New Jersey: Prentice-Hall Inc., 1971.

[307] J. W. Strutt. On the Capillary Phenomena of Jets. *Proceedings of the Royal Society of London*, 29, 71–97, 1879.

[308] M. Sun and C. Ebner. Molecular-dynamics simulation of compressible fluid flow in two–dimensional channels. *Phys. Rev. A*, 46, 4813–4818, 1992.

[309] A. Taube, M. Dumbser, D. Balsara, and C. Munz. Arbitrary High-Order Discontinuous Galerkin Schemes for the Magnetohydrodynamic Equations. *Journal of Scientific Computing*, 30, 441–464, 2007.

[310] M. Tavelli and M. Dumbser. A staggered space–time discontinuous Galerkin method for the incompressible Navier–Stokes equations on two–dimensional triangular meshes. *Computers and Fluids*, 119, 235–249, 2015.

[311] M. Tavelli, M. Dumbser, D. Charrier, L. Rannabauer, T. Weinzierl, and M. Bader. A simple diffuse interface approach on adaptive Cartesian grids for the linear elastic wave equations with complex topography. *Journal of Computational Physics*, 386, 158–189, 2019.

[312] M. Tavelli and W. Boscheri. A high order parallel Eulerian-Lagrangian algorithm for advection-diffusion problems on unstructured meshes. *International Journal for Numerical Methods in Fluids*, in press, 2019.

[313] M. Tavelli, S. Chiocchetti, E. Romenski, A.-A. Gabriel, and M. Dumbser. Space-time adaptive ADER discontinuous Galerkin schemes for nonlinear hyperelasticity with material failure. *Journal of Computational Physics*, 422, 109758, 2020.

[314] W. Thomson. XLVI. Hydrokinetic solutions and observations. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, 42, 362–377, 1871.

[315] V. Titarev and E. Toro. ADER: Arbitrary High Order Godunov Approach. *Journal of Scientific Computing*, 17, 609–618, 2002.

[316] V. Titarev and E. Toro. ADER schemes for three-dimensional nonlinear hyperbolic systems. *Journal of Computational Physics*, 204, 204–715, 2005.

[317] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics. A Practical Introduction, Third edition.* Berlin: Springer-Verlag, 2009.

[318] E. F. Toro and V. A. Titarev. Derivative Riemann solvers for systems of conservation laws and ADER methods. *Journal of Computational Physics*, 212, 150–165, 2006.

[319] E. Toro, R. Millington, and L. M. Nejad. In: Towards Very High Order Godunov Schemes. E. Toro, ed. 907–940. Boston, MA: Springer US, 2001.

[320] E. Toro and V. Titarev. Solution of the generalized Riemann problem for advection–reaction equations. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 458, 271–281, 2002.

[321] E. Toro and V. Titarev. ADER Schemes for Scalar Non-linear Hyperbolic Conservation Laws with Source Terms in Three-space Dimensions. *Journal of Computational Physics*, 202, 196–215, 2005.

[322] E. Toro and M. Vázquez-Cendón. Flux splitting schemes for the Euler equations. *Computers and Fluids*, 70, 1–12, 2012.

[323] M. Torrilhon and M. Fey. Constraint-preserving upwind methods for multidimensional advection equations. *SIAM Journal on Numerical Analysis*, 42, 1694–1728, 2004.

[324] A. Üngör. Off-centers: A new type of Steiner points for computing size-optimal quality-guaranteed Delaunay triangulations. *Proceedings of LATIN*, 152–161, 2004.

[325] A. Üngör. Off-centers: A new type of Steiner points for computing size-optimal quality-guaranteed Delaunay triangulations. *Computational Geometry*, 42, 109–118, 2009.

[326] B. van Leer. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics*, 14, 361–370, 1974.

[327] B. van Leer. Towards the ultimate conservative difference scheme.V. A second-order sequel to Godunov's method. *Journal of Computational Physics*, 32, 101–136, 1979.

[328] M. D. W. Boscheri and R. Loubere. Cell centered direct Arbitrary-Lagrangian-Eulerian ADER-WENO finite volume schemes for nonlinear hyperelasticity. *Computers and Fluids*, 134–135, 111–129, 2016.

[329] A. Wood. *A Textbook of Sound.* London: B. Bell and Sons LTD, 1930.

[330] J.-Y. Wu, V. P. Nguyen, C. T. Nguyen, D. Sutula, S. Sinaie, and S. P. Bordas. Chapter One - Phase-field modeling of fracture. In: ed. by S. P. Bordas and D. S. Balint. Vol. 53. Advances in Applied Mechanics. Elsevier, 2020. 1–183.

[331] V. Yakhot and S. Orszag. Renormalization group analysis of turbulence. I. Basic theory. *Journal of Scientific Computing*, 1, cited By 3354, 3–51, 1986.

[332] N. N. Yanenko. *The method of fractional steps.* Springer, 1971.

[333] K. Yee. Numerical solution of initial voundary value problems involving Maxwell equation in isotropic media. *IEEE Transactions on Antennas and Propagation*, 14, 302–307, 1966.

[334] P. Yue, J. Feng, C. Liu, and J. Shen. A diffuse-interface method for simulating two-phase flows of complex fluids. *Journal of Fluid Mechanics*, 515, cited By 649, 293–317, 2004.

[335] O. Zanotti and M. Dumbser. Efficient conservative ADER schemes based on WENO reconstruction and space-time predictor in primitive variables. *Computational Astrophysics and Cosmology*, 3, 2015.

This dissertation is a contribution to the development of a unified model of continuum mechanics, describing both fluids and elastic solids as a general continua, with a simple material parameter choice being the distinction between inviscid or viscous fluid, or elastic solids or visco-elasto-plastic media. Additional physical effects such as surface tension, rate-dependent material failure and fatigue can be, and have been, included in the same formalism.

The model extends a hyperelastic formulation of solid mechanics in Eulerian coordinates to fluid flows by means of stiff algebraic relaxation source terms. The governing equations are then solved by means of high order ADER Discontinuous Galerkin and Finite Volume schemes on fixed Cartesian meshes and on moving unstructured polygonal meshes with adaptive connectivity, the latter constructed and moved by means of a in-house Fortran library for the generation of high quality Delaunay and Voronoi meshes.

Further, the thesis introduces a new family of exponential-type and semi-analytical time-integration methods for the stiff source terms governing friction and pressure relaxation in Baer-Nunziato compressible multiphase flows, as well as for relaxation in the unified model of continuum mechanics, associated with viscosity and plasticity, and heat conduction effects.

Theoretical consideration about the model are also given, from the solution of weak hyperbolicity issues affecting some special cases of the governing equations, to the computation of accurate eigenvalue estimates, to the discussion of the geometrical structure of the equations and involution constraints of curl type, then enforced both via a GLM curl cleaning method, and by means of special involution-preserving discrete differential operators, implemented in a semi-implicit framework.

Concerning applications to real-world problems, this thesis includes simulation ranging from low-Mach viscous two-phase flow, to shockwaves in compressible viscous flow on unstructured moving grids, to diffuse interface crack formation in solids.

Simone Chiocchetti graduated with a master's degree in Environmental Engineering from the University of Trento, Italy. His doctoral research involves the solution of hyperbolic partial differential equations with high order Finite Volume and Discontinuous Galerkin schemes on fixed and moving grids, treatment of involution constraints and stiff relaxation sources, semi-implicit methods, and generation of unstructured Delaunay and Voronoi grids.