On Runge-Kutta discontinuous Galerkin methods for compressible Euler equations and the ideal magneto- hydrodynamical model

A dissertation submitted to

INSTITUT FÜR MATHEMATIK
DER UNIVERSITÄT WÜRZBURG

for the degree of
doctor rerum naturalium (Dr. rer. nat.)

presented by
Juan Pablo Gallego Valencia
M. Sc. Technomathematik

Referees:
Prof. Dr. Christian Klingenberg (Universität Würzburg), examiner
Prof. Dr. Gabriella Puppo (Università degli Studi dell'Insubria), co-examiner

Würzburg, 2017

ii

Contents

Abstract

An explicit Runge-Kutta discontinuous Galerkin (RKDG) method is used to device numerical schemes for both the compressible Euler equations of gas dynamics and the ideal magneto- hydrodynamical (MHD) model. These systems of conservation laws are known to have discontinuous solutions. Discontinuities are the source of spurious oscillations in the solution profile of the numerical approximation, when a high order accurate numerical method is used. Different techniques are reviewed in order to control spurious oscillations. A shock detection technique is shown to be useful in order to determine the regions where the spurious oscillations appear such that a Limiter can be used to eliminate these numeric artifacts. To guarantee the positivity of specific variables like the density and the pressure, a positivity preserving limiter is used. Furthermore, a numerical flux, proven to preserve the entropy stability of the semi-discrete DG scheme for the MHD system is used. Finally, the numerical schemes are implemented using the `deal.II C++` libraries in the `dflo` code. The solution of common test cases show the capability of the method.

Zusammenfassung

Ein explizite Runge-Kutta discontinous Galerkin (RKDG) Verfahren wird ange-wendet, um numerische Diskretisierungen, sowohl für die kompressiblen Euler-gleichungen der Gasdynamik, als auch für die idealen Magnetohydrodynamik (MHD) Gleichungen zu entwickeln. Es ist bekannt, dass diese System von Erhal-tungsgleichungen unstetige Lösungen besitzen. Unstetigkeiten sind die Quelle von störenden Oszillationen im Lösungsprofil der numerischen Näherung, wenn ein numerisches Verfahren von hoher Ordnung verwendet wird. Verschiedene Techniken werden miteinander verglichen um störende Oszillationen zu kontrol-lieren, die bei der Approximation von Unstetigkeiten in der Lösung auftreten. Ein Verfahren zur Lokalisierung von Schockwellen wird vorgestellt und es wird gezeigt, dass dieses Verfahren nützlich ist um Regionen, in denen störende Os-zillationen auftreten, zu bestimmen, so dass ein Limiter verwendet werden kann um diese numerischen Artefakte zu eliminieren. Um die Positivität spezieller Variablen, wie die Dichte und den Druck, zu bewahren, wird ein spezieller posi-tivitätserhaltender Limiter verwendet. Des Weiteren wird ein numerischer Fluss, für den bewiesenermaen das semi-diskrete DG Verfahren für das MHD System Entropie-Stabil ist, verwendet. Abschlieend werden die numerischen Verfahren unter Verwendung der `deal.II C++` Bibliotheken im `dflo` code implementiert. Simulationen bekannter Testbeispiele zeigen das Potential dieses numerischen Verfahrens.

Acknowledgments

I would like to thank...

A mi familia por su apoyo incondicional en la distancia. Gracias por entender que, aunque amo mi tierra, el mundo es más que una ciudad, un departamento o un país y quiero explorarlo mientras pueda. Al final lo que nos define como personas no es lo que tenemos sino lo que hemos vivido y cómo usamos nuestras experiencias para abrir nuestra mente y tratar de transformar el mundo.

To my friends with whom I have spend these last years, Dr. Alejandro, Dr. Gero, Dr. Gabriele, (soon Dr.) Markus, Lili, Andrea, thank you so much for your friendship and support, for listening to me and cheering me up when I was down. Also, thanks a lot to the working (or better gaming) group with whom I spend so many hours forgetting the stress of finishing the PhD either by playing, watching football or procrastinating on coffee breaks.

Grazie ai miei cari amici italiani, perché mi aiutate a imparare la vostra bella lingua, perché mi trasmettete la vostra cultura (principalmente culinaria), perché fate sempre casino e ovviamente perché cucinate. Vi ho nel mio cuore.

A very special recognition to my friend Dr. Gero Schnücke with whom I started my PhD and with whom I have had long lasting discussions about our research projects, life, the universe and everything. Thank you for all the hours you invested reading this work and all your suggestions to make this document understandable. Thanks also to Jens, Marlies, Jayesh and Dr. Stephan Schmidt for their great suggestions and corrections.

All my gratitude to Prof. Christian Klingenberg who gave me the chance to do my PhD at the University of Würzburg, who supported me with the organization of all the different applications for the necessary funding, and who ensured a constant visit of researchers into the working group in oder to discuss topics and generate possible collaboration projects. Because of these visits I got to work with Prof. Praveen Chandrashekar, who I greatly admire and from whom I have learned so much, thanks to him, to Prof. Yinhua Xia and to Prof. Gabriella Puppo for their help and the valuable discussions.

I also would like to thank the DFG Graduate School GRK1147 and the DAAD STIBET of the University of Würzburg for their support, which not only helped me during my studies but also gave me the opportunity to assist to scientific meetings while getting to know other cultures around the world. This was an invaluable experience.

All the people who have shared with me during my time in Würzburg since the very beginning, thank you very much for the good moments, for sharing your time with me, for making me feel like at home.

x

Introduction

Scalar and systems of conservation laws (CL) have been used to describe phenomena from a broad spectrum of fields, including finance [69], fluid dynamics [8] including chemical reactions [52, 58], porous media [56], traffic flow [40, 19], population dynamics [35] and so on.

It is also a very broad field of research in terms of techniques used to find numerical approximations to solutions [44, 77]. Not only they lead to a better understanding of the underlying phenomena and the methods used to solve specific problems, but also they are of great interest in terms of theory, since well posedness is not clear for systems of CL in several space dimensions up to now.

In 1965 Glimm showed existence of solutions for systems of CL in one dimension [29], and later in 1972 Kurzkov proved existence and uniqueness of solutions for scalar CL in several dimensions [48]. But on the other hand, the work by De Lellis and Székelyhidi [21, 20] showed in 2009 and 2010 a new approach in which infinite many solutions can be found for systems of CL in several dimensions.

A specific application field in which systems of CL are used is in Astrophysics, where the Euler equations of gas dynamics and the Magnetohydrodynamical (MHD) models are solved in order to study different problems, like star formation [55, 63], the fluid movement inside or around a star [36, 67], the interaction between two stars [57], the evolution of entire galaxies [74] and even the evolution of the universe starting from an initial condition like the Cosmic Microwave Background [73, 28].

To study these kind of astrophysical problems, several computer packages have been implemented including the Athena code [26, 27], GADGET [75], GADGET-2 [71] and the Arepo code [72]. These codes are based on Finite Volume (FV) methods which provide great flexibility to solve partial differential equations (PDE). The precision in the results is then limited by the computation power used to perform the simulations, and the ability of the code to take advantage of this computational power.

On the other hand, new technologies had provide higher computational power by means of a change in the paradigm. In order to fully take advantage of the nowadays super-computers, the structure of algorithms to perform simulations need to adapt to the machine's architecture. Meaning that the simulation codes need to be able to run in computers with parallel structures.

The present work was developed under the EXAMAG Project ("Exascale simulations of the evolution of the universe including magnetic fields"), part of the SPPEXA Priority project funded by the German Research Committee (Deutsches Forschungsgemeinschaft - DFG) in order to create efficient algorithms able to run in exascale computing systems currently under development

(which can process 10^{18} flops per second).

The EXAMAG project is a collaboration between astrophysicist from the Heidelberg Institute for Theoretical Studies (HITS) and Mathematicians from the University of Würzburg. The main purpose of group in Würzburg is to enhance the Arepo code with a higher order of approximation preserving the current properties of the code like Galilean invariability and high scalability for its use in super-computers.

The Arepo code was developed by Prof. Dr. Volker Springel, and it is a second order Finite Volume (FV) scheme that works on a moving mesh. The moving mesh feature is essential to preserve the Galilean invariability as it was shown in [72] with numerical experiments, where it can be seen how the solution can lost its features as a background speed is added to the equations.

Since the Arepo code is based on a FV method, in its current state a higher order implementation will rely on the use of bigger stencils. This can become highly inefficient for unstructured meshes due to the amount of information that would need to be stored related to neighboring cells in the mesh [1].

As a suitable alternative to guarantee high order accuracy, flexibility in unstructured meshes and usability in parallel computational architectures, the implementation of a discontinuous Galerkin (DG) method was proposed. The work in the project was divided in two different tasks, the first one investigating moving meshes and how would it be possible to implement DG schemes over them. The results were shown in the PhD. Thesis of Gero Schücke [68].

The second task of the project, which is presented in this work, is an implementation of the DG scheme for the compressible Euler equations of gas dynamics and the ideal Magneto- hydrodynamical (MHD) model on a Cartesian mesh, ensuring high order accuracy, handling near vacuum conditions and including the necessary tools to deal with spurious oscillations around discontinuities. The future work is then to join both tasks in order to have a moving mesh DG scheme for the Arepo code.

This work is structured as follows: Chapter 1 introduces the basic concepts related to linear and non-linear CL, the type of solutions admitted by these systems and the test cases which will be studied in later sections. The numerical scheme will be presented in Chapter 2, including the domain discretization, the finite element (FE) space and the time integration method. To deal with discontinuous solutions some very well known techniques are reviewed in Chapter 3, which help eliminate spurious oscillations that may appear in the numerical approximation. The implementation of the selected numerical scheme is tested using different cases of study, which results are displayed in Chapter 4. Finally, some concluding remarks and ideas which can be done in upcoming projects are shown.

---

[1]The Arepo code does not even store a mesh, it is generated for every time step.

Chapter 1

Basic Concepts

Introduction

The basic concepts concerning scalar and systems of conservation laws are introduced in this chapter for a better understanding of the later sections. Additionally, some examples of this kind of equation and their characteristics are presented.

1.1 Conservation laws

Consider the open set $D \subset \mathbb{R}^d, 0 < d < 3$ as a control volume. If the temporal change of a variable's density $\mathbf{q}$ is determined by its variation $\mathbf{F}(\mathbf{q})$ through the boundaries ∂D of the control volume, as

$$\partial_t \int_D \mathbf{q} dV = -\int_{\partial D} \mathbf{F}(\mathbf{q}) \cdot \mathbf{n} dS, \qquad (1.1)$$

where $\mathbf{n}$ is an outer normal unit vector, then $\mathbf{q}$ is said to be conserved, or to satisfy a **conservation law** and equation (1.1) is called the **integral form**. The variable $\mathbf{q} : (t, \mathbf{x}) \in \mathbb{R}^+ \times D \to \Omega \subset \mathbb{R}^m$, $\mathbf{q} < \infty$ is called the vector of conserved variables and $\mathbf{F} : \mathbb{R}^m \to \mathbb{R}^{m \times d}$ are called flux functions. Assuming that the control volume D does not change with time and that $\mathbf{F}$ is differentiable, applying the divergence theorem equation (1.1) can be written as

$$\int_D \left(\partial_t \mathbf{q} + \nabla_{\mathbf{x}} \cdot \mathbf{F}(\mathbf{q}) \right) dV = 0. \qquad (1.2)$$

Then, dividing (1.2) by the volume of the set D and shrinking it to a point, the **differential form** of the conservation law is obtained as

$$\partial_t \mathbf{q} + \nabla_{\mathbf{x}} \cdot \mathbf{F}(\mathbf{q}) = 0. \qquad (1.3)$$

At this point, some important things can be noticed. First of all, even if the spatial domain of the problem is one-dimensional ($d = 1$), $\mathbf{q}$ is not restricted to be scalar. Furthermore, (1.3) is a first order partial differential equation (PDE), which will allow solutions in the classical sense only when $\mathbf{q} \in C^1(\mathbb{R}^+ \times D)$, however, as it will be shown later, the solutions allowed by a conservation laws are not restricted to be classical.

Equation (1.3) can also be written in quasilinear form as

$$\partial_t \mathbf{q} + \sum_{\kappa=1}^{d} \mathbf{A}_\kappa \partial_{x_\kappa} \mathbf{q} = 0, \tag{1.4}$$

where $\mathbf{A}_\kappa$ is defined as

$$\mathbf{A}_\kappa = \left[\frac{\partial F_i}{\partial q_j} \right]_\kappa \tag{1.5}$$

the Jacobian matrix resulting from partially derive each function of $F_{i,\kappa}$ with respect to q_j, where $1 \le i, j \le m$ and $1 \le \kappa \le d$.

If all the eigenvalues of the matrices $\mathbf{A}_\kappa$, $\kappa = 1, ..., d$, are real, the system of equation 1.3 is said to be **hyperbolic**. Moreover, Depending on the type of eigenvalues that the system has, it can be characterized as either **strictly**, **strongly** or **weakly** hyperbolic.

1.1.1   Linear conservation laws

The simplest example of a scalar hyperbolic conservation law is called the **advection equation**. Let q be scalar, taking $d = 1$ and assigning an initial condition $q_0(x)$ to equation (1.3), a initial value problem (IVP) can be written as

$$\begin{cases} \partial_t q + a \partial_x q = 0, \\ q(0, x) = q_0(x), \end{cases} \tag{1.6}$$

where a is a given constant. The time evolution of this equation can be interpreted as the movement of the profile $q_0(x)$ at constant speed a to the left (if $a < 0$) or to the right (if $a > 0$). This means that the solution at any point of space and time can be calculated as $q(t, x) = q_0(x - a \cdot t)$ using the method of characteristics.

Consider now systems of linear equations, that is, when $\mathbf{A}$ in equation (1.4) is a constant matrix.

$$\begin{cases} \partial_t \mathbf{q} + \mathbf{A} \nabla_{\mathbf{x}} \mathbf{q} = 0, \\ \mathbf{q}(0, \mathbf{x}) = \mathbf{q}_0(\mathbf{x}). \end{cases} \tag{1.7}$$

If $\mathbf{A}$ can be diagonalizable such that $\mathbf{A} = \mathbf{R}\mathbf{\Lambda}\mathbf{R}^{-1}$, where $\mathbf{R}$ is the matrix of right eigenvectors and $\mathbf{\Lambda}$ is a diagonal matrix with the eigenvalues of $\mathbf{A}$, then the system (1.7) can be transformed into a system of advection equations as follows

$$\partial_t \mathbf{q} + \mathbf{A} \nabla_{\mathbf{x}} \mathbf{q} = 0,$$

$$\partial_t \mathbf{q} + \left(\mathbf{R}\mathbf{\Lambda}\mathbf{R}^{-1} \right) \nabla_{\mathbf{x}} \mathbf{q} = 0.$$

Pre-multiplying by $\mathbf{R}^{-1}$ follows

$$\mathbf{R}^{-1} \partial_t \mathbf{q} + \mathbf{R}^{-1} \left(\mathbf{R}\mathbf{\Lambda}\mathbf{R}^{-1} \right) \nabla_{\mathbf{x}} \mathbf{q} = 0.$$

Since $\mathbf{R}^{-1}$ does not depend on time or the space variables then

$$\partial_t \left(\mathbf{R}^{-1} \mathbf{q} \right) + \mathbf{\Lambda} \nabla_{\mathbf{x}} \left(\mathbf{R}^{-1} \mathbf{q} \right) = 0,$$

and finally

$$\begin{cases} \partial_t \mathbf{z} + \mathbf{\Lambda} \nabla_\mathbf{x} \mathbf{z} = 0, \\ \mathbf{z}(0, \mathbf{x}) = \mathbf{z}_0(\mathbf{x}), \end{cases} \tag{1.8}$$

where $\mathbf{z} = \mathbf{R}^{-1} \mathbf{q}$ is called the vector of **characteristic** variables and also $\mathbf{z}_0(\mathbf{x}) = \mathbf{R}^{-1} \mathbf{q}_0(\mathbf{x})$. The velocities at which the solution of the IVP (1.8) would be moving with respect to time in every space direction are given by the eigenvalues λ_i of the matrix $\mathbf{A}$, which appear at the diagonal of the matrix $\mathbf{\Lambda}$.

1.1.2 Non-linear conservation laws

Consider the scalar conservation laws IVP given by

$$\begin{cases} \partial_t q + \partial_x f(q) = 0, \\ q(0, x) = q_0(x), \end{cases} \tag{1.9}$$

where $f \in C^1(\Omega)$, $\Omega \subset \mathbb{R}$ is a non-linear function. The quasilinear form is given by

$$\begin{cases} \partial_t q + f'(q) \partial_x q = 0, \\ q(0, x) = q_0(x), \end{cases} \tag{1.10}$$

Let $\psi(t)$ be a parametrization of the space component x in terms of the time t, along which the solution $q(t, x)$ of the IVP (1.10) remains constant in time. The following expression is then satisfied

$$\frac{d}{dt} q(t, \psi(t)) = \partial_t q(t, \psi(t)) + \psi'(t) \partial_x q(t, \psi(t)) = 0,$$

only when $\psi'(t) = f'(q(t, x(t)))$. Then, the expression $\psi(t)$ is called **characteristic curve**, and it is described by the ODE

$$\begin{cases} \psi'(t) = f'(q(t, \psi(t))), \\ \psi(0) = x_0. \end{cases} \tag{1.11}$$

Now, probably one of the most interesting things about hyperbolic conservation laws, is that they can develop discontinuous solutions starting from smooth initial conditions, and this can occur when the characteristic intersect. In order to illustrate this issue, consider the simplest non-linear scalar conservation law

$$\partial_t q(t, x) + \partial_x \left(\frac{q^2(t, x)}{2} \right) = 0, \tag{1.12}$$

where $(t, x) \in \mathbb{R}_+ \times \mathbb{R}$. This equation is called the **inviscid Burgers' equation**. If equation (1.12) is written in quasilinear form as in (1.10), and an initial condition is assigned, the following IVP for expression (1.12) is obtained

$$\begin{cases} \partial_t q(t, x) + q(t, x) \partial_x q(t, x) = 0, \\ q(0, x) = q_0(x). \end{cases} \tag{1.13}$$

Due to the direct dependency of the slope of the characteristic lines to the conserved variable in equation (1.13), the characteristic lines could eventually cross, in which case a discontinuity will appear in the solution.
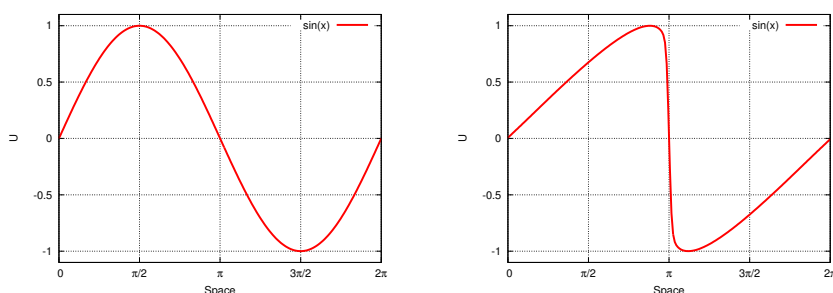
(a) Initial condition for the IVP (1.10)   (b) Shock formation of the IVP (1.10)

Figure 1.1: Time evolution of the IVP (1.10)

As an example, take the IVP (1.13) with $q_0(x) = \sin(x)$, $x \in [0, 2\pi]$ as illustrated in Figure 1.1a. According to (1.11), there are three different possible range of values for the characteristic slope

$$\psi'(t) = \begin{cases} q_0(x) > 0, & x \in (0, \pi), \\ q_0(x) = 0, & x = 0, x = \pi, x = 2\pi, \\ q_0(x) < 0, & x \in (\pi, 2\pi). \end{cases} \tag{1.14}$$

This means that for $x \in (0, \pi)$ the solution will move towards the right, and similarly, for $x \in (\pi, 2\pi)$ the solution will move towards the left. After a brief period of time the characteristic curves around $x = \pi$ will cross generating a discontinuity (Figure 1.1b), where the conserved variable q will jump from having a positive value to a negative value.

The fact that the solution is not continuous anymore generates a contradiction, as a discontinuous function cannot be differentiated (at least in the classical sense.) Therefore, the set of solutions in which $q(t, x)$ lays should include also discontinuous functions that allow the so called weak derivatives, these solutions are then called **weak solutions**. Then, multiplying equation (1.3) by a test function $\phi(\mathbf{x})$ and integrating over space and time follows

$$\int_{\mathbb{R}^d} \int_{\mathbb{R}_+} \left[\partial_t \mathbf{q} + (\nabla_x \cdot \mathbf{f(q)}) \right] \phi \, dt \, d\mathbf{x} = 0. \tag{1.15}$$

Then, integrating by parts the following condition should be satisfied by weak solutions

$$\int_{\mathbb{R}^d} \int_{\mathbb{R}_+} \left[\phi \partial_t \mathbf{q} - \mathbf{f(q)} \nabla_x \phi \right] dt \, d\mathbf{x} + \int_{\mathbb{R}^d} \phi(0, x) \mathbf{q}_0(x) d\mathbf{x} = 0, \tag{1.16}$$

for all $\phi \in C_0^\infty \left(\mathbb{R}^d \right)$ [47]. This solution is also called a generalized solution for PDE (1.3). Note that the definition is given for vector valued solutions $\mathbf{q}$ and $\mathbf{x} \in \mathbb{R}^d$, so it is not restricted to scalar solutions q, which means that it applies also for systems of conservation laws in multiple dimensions.

Since not all discontinuous functions are included in the set of admissible solutions, a generalized solution must satisfy the so called **Rankine-Hugoniot jump conditions** explained as follows [51]. Let $S(t)$ be a smooth surface moving with t, and let a generalized solution $\mathbf{q}$ of (1.3) be discontinuous along
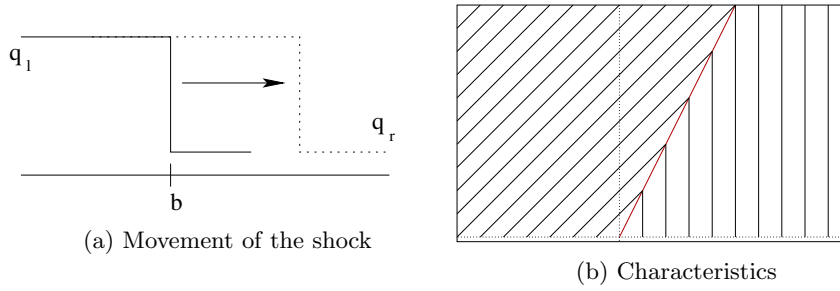
(a) Movement of the shock

(b) Characteristics

Figure 1.2: Solution for Case 1

$S(t)$, but be continuous differentiable in either side of $S(t)$. Then

$$s(t)\left[q^j\right] = \left[f^j\left(\mathbf{q}\right)\right] \cdot \mathbf{n} \tag{1.17}$$

must be satisfied at each point of $S(t)$. Here $\left[q^j\right]$ and $\left[f^j\left(\mathbf{q}\right)\right]$ are respectively the differences between the values of $\mathbf{q}$ and $\mathbf{F}$ on the two sides of $S(t)$, $\mathbf{n}$ is a vector normal to $S(t)$ and $s(t)$ is the speed at which $S(t)$ propagates in the direction $\mathbf{n}$. For scalar problems, one can simply write

$$\left(q_r - q_l\right)s(t) = f\left(q_l\right) - f\left(q_r\right), \tag{1.18}$$

and then

$$s(t) = \frac{f\left(q_l\right) - f\left(q_r\right)}{q_r - q_l}. \tag{1.19}$$

1.1.3 The Riemann Problem

Consider the following initial condition

$$q_0(x) = \begin{cases} q_l & \text{if } x < b, \\ q_r & \text{if } x > b, \end{cases} \tag{1.20}$$

for any $b \in \mathbb{R}$ and assume for now that $q_l, q_r > 0$. A conservation law (1.9) with an initial condition of the kind (1.20) is called a **Riemann Problem**. Taking once again as illustration example the scalar Burgers equation (1.12), two different kinds of solutions arise from this problem depending on the values of q_l and q_r.
Case 1: $q_l > q_r$

In this case the characteristic curves at the left of the jump "crash" against the characteristic curves at the right, merging into what is called a **shock wave** (Figure 1.2b). The speed s at which the discontinuity (the shock wave) moves, can be computed using the Rankine–Hugoniot jump conditions. In the particular case of scalar Burgers' equation it is be given by

$$s = \frac{q_l + q_r}{2}, \tag{1.21}$$

Case 2: $q_l < q_r$

In this case the characteristic curves at the left side of the discontinuity will have a slower speed than the one at the right side, which produces a "gap"
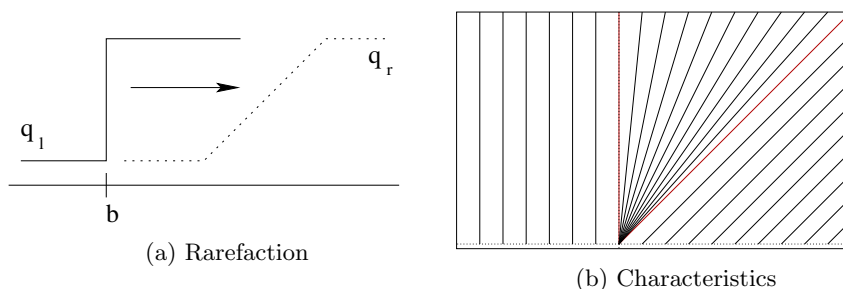
(a) Rarefaction

(b) Characteristics

Figure 1.3: Solution for Case 2

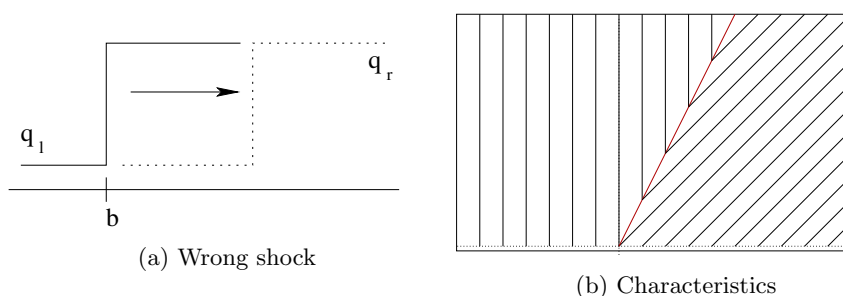(a) Wrong shock

(b) Characteristics

Figure 1.4: Incorrect solution for Case 2

in the domain with no defined solution. Then, many different weak solutions can be used, including the one from in case 1 (a discontinuity propagating with speed s, from where characteristic curves would emerge as in figure 1.4b). A different approach (which is indeed the correct one) would be to create a transition between the left and right solutions as time evolves, as it is shown in figure 1.3b. This transition is called a **rarefaction wave**, which is in fact a vanishing viscosity generalized solution. This solution is defined as

$$\begin{cases} q_l & \text{if } x < b + q_l t, \\ x/l & \text{if } b + q_l t < x < b + q_l t, \\ q_l & \text{if } x > b + q_r t. \end{cases} \tag{1.22}$$

Unfortunately, as it was shown in case 2, weak solutions are often not unique and it is still considered an open problem to prove uniqueness of solutions to general systems of conservation laws in more than one space dimension.

A criteria to find the correct weak solution for a conservation law problem has to be defined. For scalar problems, this can be done by adding restrictions in the way weak solutions are found. These kind of restrictions are called **entropy conditions**, by analogy to physical systems where solutions that are "physically meaningful" are selected.

1.1.4   Entropy conditions and vanishing viscosity

In order to ensure existence and uniqueness of solutions for scalar conservation laws, and therefore reduce the ambiguity in selecting the "correct" solution, the
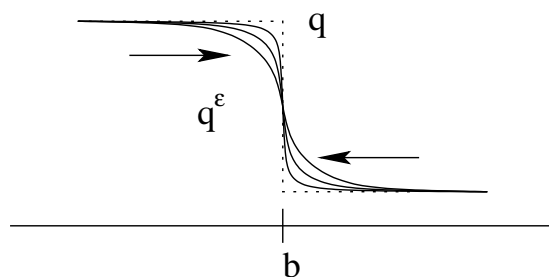
Figure 1.5: As $\varepsilon \to 0$ the solution q^ε becomes sharper. Its limit should be q, the solution to the hyperbolic conservation law

vanishing viscosity method is proposed as follows. Consider an additional term in the previously defined conservation law (1.9), transforming the IVP to

$$\begin{cases} \partial_t q + \partial_x f(q) = \varepsilon \partial_x^2 q, \\ q(0, x) = q_0(x), \end{cases} \tag{1.23}$$

where $0 < \varepsilon \ll 1$. Equation (1.23) is called a **viscous approach** of the conservation law (1.9). The solutions q^ε for this new IVP are smooth, thus one would like to find a unique limit q as $\varepsilon \to 0$ which in turn would be the solution for the conservation law IVP (1.9).

Consider a pair of functions $\eta, \Phi \in C^2(\mathbb{R})$, such that η is convex and $\Phi' = \eta' f'$, and let q^ε be a smooth solution for (1.23). Then, it follows

$$\begin{aligned} \partial_t \eta(q^\varepsilon) + \partial_x \Phi(q^\varepsilon) - \varepsilon \partial_x^2 \eta(q^\varepsilon) &= \eta'(q^\varepsilon) \partial_t q^\varepsilon + \Phi'(q^\varepsilon) \partial_x q^\varepsilon \\ &\quad - \varepsilon \partial_x \left(\eta'(q^\varepsilon) \partial_x q^\varepsilon \right) \\ &= -\varepsilon \eta'(q^\varepsilon) \partial_x^2 q^\varepsilon \le 0, \end{aligned} \tag{1.24}$$

since η is convex. By multiplying the previous expression by a non-negative test function ϕ and integrate over time and space

$$\int_{\mathbb{R}} \int_{\mathbb{R}^+} \left[\eta(q^\varepsilon) \partial_t \phi + \Phi(q^\varepsilon) \partial_x \phi + \varepsilon \eta(q^\varepsilon) \partial_x^2 \phi \right] \ge 0. \tag{1.25}$$

As $\varepsilon \to 0$ it can be proven [47] that

$$\begin{aligned} q^\varepsilon &\to q, \\ \eta^\varepsilon(q^\varepsilon) &\to \eta(q), \\ \Phi^\varepsilon(q^\varepsilon) &\to \Phi(q), \end{aligned} \tag{1.26}$$

in L^1, $|\eta(q^\varepsilon)| < C$, C constant as $\varepsilon \to 0$. Then finally

$$\partial_t \eta(q) + \partial_x \Phi(q) \le 0. \tag{1.27}$$

Equation (1.27) is known as the **entropy inequality**. Thus, if it is possible to find a convex function η and a function Φ such that $\Phi = \eta' f'$, which also satisfy (1.27), the solution q is said to be an entropy satisfying solution for (1.9), and the pair (η, Φ) is called an entropy pair for the conservation law (1.9).

In the case of systems of conservation laws, this method works only for particular cases, and it is still consider an open problem to prove uniqueness of solutions.

1.2   Cases of study

Some examples of conservation law systems will be introduced in the following pages. These examples will be treated in later chapters, therefore a brief introduction about their physical meaning is important.

1.2.1   Euler equations

$$\rho_t + \nabla \cdot (\mathbf{m}) = 0,$$

$$(\mathbf{m})_t + \nabla \cdot \left(\frac{1}{\rho} \mathbf{m}\mathbf{m} + pI \right) = 0, \tag{1.28}$$

$$E_t + \nabla \cdot \left((E + p)\, \frac{\mathbf{m}}{\rho} \right) = 0.$$

This system of conservation laws, also known as the compressible model of gas dynamics, gathers the conservation of **mass**, **momentum** and **energy** respectively. It can be written as the equation (1.3) for $d = 2$, where

$$\mathbf{q} = \left[\rho, \mathbf{m}^T, E \right], \quad \mathbf{F}(\mathbf{q}) = \left[\mathbf{f}_1(\mathbf{q}), \mathbf{f}_2(\mathbf{q}) \right], \tag{1.29}$$

and

$$\mathbf{f}_1(\mathbf{q}) = \begin{bmatrix} m_1 \\ \frac{m_1^2}{\rho} + p \\ \frac{m_1 m_2}{\rho} \\ \frac{m_1}{\rho}(E + p) \end{bmatrix}, \quad \mathbf{f}_2(\mathbf{q}) = \begin{bmatrix} m_2 \\ \frac{m_1 m_2}{\rho} \\ \frac{m_2^2}{\rho} + p \\ \frac{m_2}{\rho}(E + p) \end{bmatrix}. \tag{1.30}$$

Here, $\rho > 0$ is the density, $\mathbf{m} = [m_1, m_2]$ is the vector of momentum in each space dimension, $p > 0$ is the pressure and E is the total energy. This is a system of $d + 2$ equations with $d + 3$ variables, which means that an additional equation is needed to close the system. This closure is achieved with an extra expression for the total energy, which is the sum of the internal pressure and the kinetic energy due to the gas momentum, given by

$$E = \frac{p}{\gamma - 1} + \frac{1}{2\rho} |\mathbf{m}|^2.$$

where γ is the adiabatic constant dependent on the type of gas. These equations neglect the effect of viscosity and thermal diffusion. Then, for this problem the admissible set of solutions $\mathcal{U}_{\mathrm{ad}}$ is defined as

$$\mathcal{U}_{\mathrm{ad}} = \{ \mathbf{q} \in \mathbb{R}^4 : \rho > 0, \quad p(\mathbf{q}) > 0 \}$$

Writing equation (1.28) in quasilinear form as in (1.4), the Jacobian matrices $\mathbf{A}_1 = \mathbf{f}_1'(\mathbf{q})$ and $\mathbf{A}_2 = \mathbf{f}_2'(\mathbf{q})$ can be computed in terms of the conserved variables. The eigenvalues of these matrices are respectively

$$\lambda_1 = u - c, \quad \lambda_2 = u, \quad \lambda_3 = u + c.$$

for $\mathbf{A}_1$, and

$$\lambda_1 = v - c, \quad \lambda_2 = v, \quad \lambda_3 = v + c.$$

for $\mathbf{A}_2$. Here, c is the so called speed of sound which is given by $c = \sqrt{\frac{\gamma p}{\rho}}$, and the values $u = \frac{m_1}{\rho}$ and $v = \frac{m_2}{\rho}$ are respectively the speeds of the fluid in each space dimension.

1.2.2 Ideal MHD equations

The ideal magneto-hydrodynamical model, or ideal MHD equations, considers the case in which, additionally to the conservation of mass, momentum and energy described by the Euler equations, the fluid under study can be affected by magnetic fields. The MHD equations are the following

$$\rho_t + \nabla \cdot (\mathbf{m}) = 0$$

$$(\mathbf{m})_t + \nabla \cdot \left(\frac{1}{\rho} \mathbf{m}\mathbf{m}^T + \left(p + \frac{1}{2} |\mathbf{B}|^2 \right) I - \mathbf{B}\mathbf{B}^T \right) = 0$$

$$\mathbf{B}_t + \nabla \cdot \left(\frac{1}{\rho} \mathbf{B}\mathbf{m}^T - \frac{1}{\rho} \mathbf{m}\mathbf{B}^T \right) = 0 \qquad (1.31)$$

$$E_t + \nabla \cdot \left(\frac{1}{\rho} \left(E + p + \frac{|\mathbf{B}|^2}{2} \right) \mathbf{m} - \frac{1}{\rho} (\mathbf{B} \cdot \mathbf{m}) \mathbf{B} \right) = 0$$

where $\mathbf{B}$ is the vector of magnetic components in each space dimension defined as $\mathbf{B} = [B_1, B_2, B_3]$. The magnetic interaction is added by coupling the Maxwell equations to the existent compressible hydrodynamical model.

It is an inherently three dimensional problem, due to the vectorial interaction of the magnetic field and the momentum vectors, therefore any problem described in one and two dimensions are simplifications of the three dimensional case when some symmetrical behavior is assumed. This means that even for one-dimensional problems all three components of the momentum and magnetic field vectors have to be considered.

In two dimensions this system of equations can be written as the conservation law (1.3), with

$$\mathbf{q} = \left[\rho, \mathbf{m}^T, \mathbf{B}^T, E \right]^T, \quad \mathbf{F}(\mathbf{q}) = [\mathbf{f}(\mathbf{q}_1), \mathbf{f}_2(\mathbf{q})], \qquad (1.32)$$

$$
\mathbf{f}_1 = \begin{bmatrix}
m_1 \\
\frac{m_1^2}{\rho} + p_{tot} - B_1^2 \\
\frac{m_1 m_2}{\rho} - B_1 B_2 \\
\frac{m_1 m_3}{\rho} - B_1 B_3 \\
0 \\
\frac{m_1}{\rho} B_2 - \frac{m_2}{\rho} B_1 \\
\frac{m_1}{\rho} B_3 - \frac{m_3}{\rho} B_1 \\
\frac{m_1}{\rho} (E + p_{tot}) - \frac{B_1}{\rho} (\mathbf{m} \cdot \mathbf{B})
\end{bmatrix}
\quad
\mathbf{f}_2 = \begin{bmatrix}
m_2 \\
\frac{m_2 m_1}{\rho} - B_2 B_1 \\
\frac{m_2^2}{\rho} + p_{tot} - B_2^2 \\
\frac{m_2 m_3}{\rho} - B_2 B_3 \\
\frac{m_2}{\rho} B_1 - \frac{m_1}{\rho} B_2 \\
0 \\
\frac{m_2}{\rho} B_3 - \frac{m_3}{\rho} B_2 \\
\frac{m_2}{\rho} (E + p_{tot}) - \frac{B_2}{\rho} (\mathbf{m} \cdot \mathbf{B})
\end{bmatrix}
\qquad (1.33)
$$

where $\rho > 0$ is the density and $\mathbf{m}$ stands for the momentum vector as in the Euler equations. The total pressure is the sum of the thermal and the magnetic pressure $p_{tot} = p + \frac{1}{2} |\mathbf{B}|^2$, $\mathbf{B}$ stands for the magnetic field vector and E is the total energy.

As in the previous case of study, an additional equation is needed to close the system. Here the total energy is a function of the thermal pressure, the kinetic energy and the magnetic pressure

$$E = \frac{p}{\gamma - 1} + \frac{1}{2\rho} |\mathbf{m}|^2 + \frac{1}{2} |\mathbf{B}|^2.$$

where γ is the adiabatic constant dependent on the type of gas. Then, for this problem the admissible set of solutions $\mathcal{U}_{\mathrm{ad}}$ is defined as

$$\mathcal{U}_{\mathrm{ad}} = \{\mathbf{q} \in \mathbb{R}^8 : \rho > 0, \quad p(\mathbf{q}) > 0\}.$$

Finally, the magnetic field of this system of equations should satisfy the divergence free condition

$$\mathbf{div}\,(\mathbf{B}) = 0, \tag{1.34}$$

which can be interpreted physically as the impossibility for magnetic monopoles to exist. Even though this result is satisfied theoretically, it is not immediately satisfied when a numerical approximation is implemented. Moreover, it can be found in the literature as one of the the main challenges when implementing numerical solutions for the MHD system.

As in the case of the Euler systems of equations, one can write the two-dimensional MHD system in the quasilinear form (1.4) by finding the Jacobian matrices $\mathbf{A}_1 = \frac{\partial \mathbf{f_1}}{\partial \mathbf{q}}$ and $\mathbf{A}_2 = \frac{\partial \mathbf{f_2}}{\partial \mathbf{q}}$. It turns out that the matrices $\mathbf{A}_1$ and $\mathbf{A}_2$ are singular, which means that the system is non-strictly hyperbolic. The set of eigenvalues for $\mathbf{A}_i$, for $i = \{1, 2\}$, which give the speed of propagation of the waves in the i-th spatial component, are found to be

- a zero eigenvalue $\lambda_0 = 0$,

- one entropy wave with speed $\lambda_e = u_i$,

- two Alfvén waves with speeds $\lambda_a^\pm = u_i \pm \frac{B_i}{\sqrt{\rho}}$, and

- four magneto acoustic waves with speeds $\lambda_{f,s}^\pm = u_1 \pm c_{f,s}$

where u_i stands for the i-th component of the speed vector $\mathbf{u}$, B_i stands for the i-th component of the magnetic field vector $\mathbf{B}$, and finally c_f and c_s stand for the fast and slow speeds given by

$$c_{f,s}^2 = \frac{1}{2}\left(a^2 + b^2\right) \pm \frac{1}{2}\sqrt{\left(a^2 + b^2\right)^2 - 4a^2(b \cdot \mathbf{n})^2}.$$

Here, $a = \sqrt{\gamma \frac{p}{\rho}}$ stands for the speed of sound, $b := \frac{1}{\sqrt{\rho}}\mathbf{B}$, which means that $b^2 = \frac{1}{\rho}|\mathbf{B}|^2$ and $\mathbf{n}$ is a normal vector.

The fact that the system is non-strictly hyperbolic (e.i., it has a zero eigenvalue) generates an additional challenge when finding a numerical solution to it. This problem will be addressed in future sections.

Summary

- A general description of scalar and systems of conservation laws was introduced in this chapter. The set of admissible solutions for this kind of PDE must include discontinuous solutions, therefore the concept of weak solution was introduced.

- A strategy to choose the right weak solution for a particular problem is established using the so called entropy conditions, which are justified by the vanishing viscosity method.

- Even though the vanishing viscosity method provide existence and uniqueness arguments for scalar conservation laws, the hope is that it also works for systems of conservation laws. However this is still considered an open problem.

- Finally, some particular systems of conservation laws were introduced, which will be treated in the following chapters.

The following chapter will introduce the basic concepts and characteristics of the strategy which will be implemented for the numerical solution of systems of conservation laws.

Chapter 2

The Discontinuous Galerkin method

Introduction

The fundamental theoretical aspects concerning scalar and systems of conservation laws were introduced in the previous chapter. In this chapter, the numerical method selected to numerically solve the scalar and systems of conservation laws will be presented.

Different characteristics of the Finite Element method (FEM) and Finite Volume method (FVM), which are the base of the discontinuous Galerkin (DG) method will be introduced in the present chapter, as well as domain discretization, the space of basis functions, and the solution of volume and surface integrals.

2.1 Domain discretization

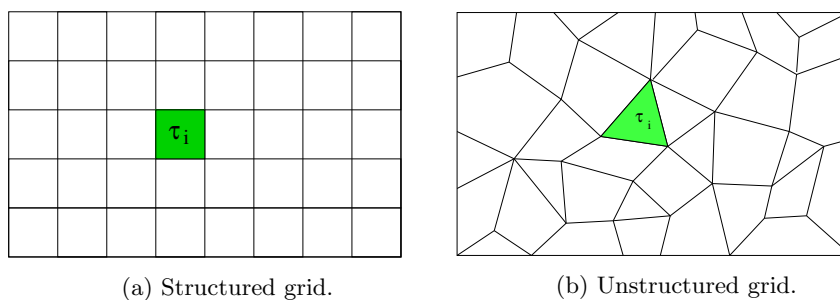<div align="center">(a) Structured grid.   (b) Unstructured grid.</div>

Figure 2.1: Discretization for a domain D. The cells in the domain can be in general any type of polygon

In order to compute approximate solutions for the conservation laws, the first step is to discretize the domain. This is done by defining a tessellation T over the set $\bar{D}$. This tessellation typically has the following characteristics

- $\bar{D} = \bigcup_{\ell=1}^{N} \tau_\ell,$

- each $\tau \in T$ is a closed set and the interior of τ^{o} is non-empty,

- $\tau_i^{\mathrm{o}} \cap \tau_j^{\mathrm{o}} = \varnothing,\ \forall i \neq j, \tau_i, \tau_j \in T.$

- for each $\tau \in T$ the boundary $\partial\tau$ is Lipschitz-continuous.

Figure (2.1) shows examples of tessellations, which can be typically classified as structured (2.1a) or unstructured (2.1b) meshes, depending on whether all the cells τ_ℓ have or not the same shape and size.

Assuming that $\mathbf{q}_\ell$ is the solution of the PDE

$$
\begin{cases}
\partial_t \mathbf{q} + \nabla_{\mathbf{x}} \cdot \mathbf{F}(\mathbf{q}) = 0, \\
\mathbf{q}(0, \mathbf{x}) = \mathbf{q}_0(\mathbf{x}),
\end{cases}
\tag{2.1}
$$

in the cell $\tau_\ell \subset D$, the solution $\mathbf{q}(t, \mathbf{x})$ to the PDE in the whole domain D can be written as the sum of the local solutions from each cell τ_ℓ of the triangulation T. That is

$$
\mathbf{q}(t, \mathbf{x}) = \sum_{\ell=0}^{N} \mathbf{q}_\ell(t, \mathbf{x}) \chi_\ell(\mathbf{x}),
$$

where $\chi_\ell(\mathbf{x})$ denote an indicator function of the form

$$
\chi_\ell(\mathbf{x}) = \begin{cases}
1 & \text{if}\ \ \mathbf{x} \in \tau_\ell \\
0 & \text{otherwise}
\end{cases}
$$

Physical to computational domain

In order to simplify the implementation, a mapping between the cells in the physical domain to a generic or reference cell in a computational domain can be defined. For one-dimensional problems, the variable $x \in \tau_i = [a, b]$ can be transformed by this simple operation [37]

$$
x(r) = a + \frac{1+r}{2} h_i, \quad h_i = b - a,
\tag{2.2}
$$

to the reference variable $r \in I = [-1, 1]$.

For higher dimensional problems, this can be easily implemented when the domain discretization is composed by a low variety of cell shapes (polygons in general). As this number increases the implementation can become tedious.

In the particular case of Cartesian grids, a trivial extension of the one-dimensional case is implemented. In two dimensions, for example, one can connect the points $\mathbf{x} \in [a, b] \times [c, d]$ and $\mathbf{r} \in [-1, 1] \times [-1, 1]$ by the mapping

$$
\mathbf{x} = \mathbf{J}\mathbf{r} + \bar{\mathbf{x}}_0, \quad \mathbf{J} = \frac{1}{2}\begin{pmatrix} b-a & 0 \\ 0 & d-c \end{pmatrix}, \quad \mathbf{x}_0 = \frac{1}{2}\begin{pmatrix} b+a \\ d+c \end{pmatrix}
\tag{2.3}
$$

The Jacobian matrix $\mathbf{J}$ has full rank and therefore it is invertible, so the mapping from the computational domain to the physical domain can also be achieved through $\mathbf{r} = \mathbf{J}^{-1}(\mathbf{x} - \bar{\mathbf{x}}_0)$.

This mapping definition is also useful since different families of polynomials, which are used as basis and trial functions, are defined on a reference set or interval. This is the case for example of the Legendre polynomials.

The weak representation

Consider the subspace $\mathcal{W}_h^k \subset L^2$ given by

$$\mathcal{W}_h^k := \left\{ \phi \in \mathcal{C}(D) : \ \phi|_{\tau_\ell} \in \mathcal{P}^k(\tau_\ell) \ \forall \ell = 1, ..., N \right\}. \tag{2.4}$$

In order to motivate the use of the weak representation, consider the case for which the solution of the conservation law is approximated by its L^2-projection $\mathbf{q}_h = P_h(\mathbf{q})$ in $\mathcal{W}_h^k$, such that it satisfies

$$\int_D (\mathbf{q} - P_h(\mathbf{q}))\phi d\mathbf{x} = 0. \tag{2.5}$$

Replacing the exact solution $\mathbf{q}$ by $\mathbf{q}_h$ in the PDE (2.1), the conservation law is not exactly satisfied and then a residual error $\mathbf{R}_h$ shows up.

$$\partial_t \mathbf{q}_h + \nabla_\mathbf{x} \cdot \mathbf{F}(\mathbf{q}_h) = \mathbf{R}_h, \tag{2.6}$$

Writing a weak representation of the PDE (2.6) by multiplying the equation with a test function, integrating over space and performing integration by parts as in subsection (1.1.2), follows the semi-discrete scheme

$$\int_D \left[\varphi \partial_t \mathbf{q}_h - \mathbf{F}(\mathbf{q}_h) \nabla_x \varphi \right] d\mathbf{x} = \int_D \mathbf{R}_h \varphi d\mathbf{x}. \tag{2.7}$$

Since the solution is supposed to be continuous at the cell interfaces through all the domain, the additional terms resulting after the integration by parts cancel out in a telescopic way as the local solutions are summed up.

When the flux function is a linear operator $\mathbf{F}(\mathbf{q}_h) = \mathbf{A}\mathbf{q}_h$, where $\mathbf{A}$ is a constant matrix, the right-hand-side of equation (2.7) can be vanished by selecting the test function φ as ϕ, a member of the basis function space $\mathcal{W}_h^k$, since the residual $\mathbf{R}_h \in \left(\mathcal{W}_h^k\right)^\perp$ which is orthogonal to $\mathcal{W}_h^k$. Then

$$\int_D \left[\phi \partial_t \mathbf{q}_h - \mathbf{A}\mathbf{q}_h \nabla_x \phi \right] d\mathbf{x} = 0. \tag{2.8}$$

Unfortunately this is not always the case, since $\mathbf{F}(\mathbf{q}_h)$ is in general a nonlinear function, therefore an approximation error is generated depending on the polynomial order k of the basis functions.

$$\int_D \left[\phi \partial_t \mathbf{q}_h - \mathbf{F}(\mathbf{q}_h) \nabla_x \phi \right] d\mathbf{x} = \mathcal{O}(h^{k+1}). \tag{2.9}$$

This procedure, where the test and basis functions belong to the same function space, is called the **Galerkin method**.

2.2 Finite element space

The finite element method is a procedure for solving PDE numerically. The main idea is to approximate the solution of a PDE, first by decomposing the domain in cells (like it was shown in section 2.1), and afterwards by approximating the solution in each cell as a linear combination of a set of functions called **elements**, **basis functions** or **trial functions**.

In order to define the finite element space, let now $\mathbf{q}_{h,\ell}(t, \mathbf{x})$ be the vector of approximate states $q_{h,\ell}(t, \mathbf{x})$ to the local solution at the cell τ_ℓ. Thus, each one of the approximated states will be defined as a linear combination of space dependent function $v_j^\ell(\mathbf{x})$

$$q_{h,\ell}^{(i)}(t, \mathbf{x}) = \sum_{j=1}^{N_p} \hat{q}_j^{(i)}(t) v_j^\ell(\mathbf{x}), \quad \forall i = 1, ..., m, \tag{2.10}$$

where N_p is the number of basis functions $v_j^\ell(\mathbf{x})$. In particular, the functions $v_j^\ell(\mathbf{x})$ are a finite set of basis functions for the space $\mathcal{P}^k(\tau_\ell)$. This motivates the definition of the following finite element space $\mathcal{V}_h^k$

$$\mathcal{V}_h^k = \{v \in L^2(D) : v|_{\tau_\ell} \in \mathcal{P}^k(\tau_\ell), \ \forall \ell = 1, ..., N\}. \tag{2.11}$$

$\mathcal{P}^k$ is restricted in this work to a family of polynomials like Legendre or Lagrange (cf. [39, 61]), however it is also possible to find approaches in the literature which use not-polynomial basis functions, for instances, as it was discussed in [82].

Defining $\forall i = 1, ..., m$

$$\hat{\mathbf{Q}}_i(t) := [\hat{q}_1^{(i)}(t), ...\hat{q}_{N_p}^{(i)}(t)]^T \quad \text{and} \quad \mathbf{v}^\ell(\mathbf{x}) := [v_1^\ell(\mathbf{x}), ..., v_{N_p}^\ell(\mathbf{x})]^T \tag{2.12}$$

as the vectors of time dependent coefficients and basis functions respectively, the equation (2.10) can be written in vectorial form as

$$q_{h,\ell}(t, \mathbf{x}) = \hat{\mathbf{Q}}(t) \cdot \mathbf{v}^\ell(\mathbf{x}). \tag{2.13}$$

Basis functions for higher dimensions

For higher dimensions, the space of polynomial basis can be build as a tensor product of one-dimensional basis. There are usually two different forms used, which are

1. The tensorial product with maximum combined order k called $\mathbb{P}^k$. This means that for each element of this basis the sum of the polynomial orders in each dimension has to be at most k ($\sum_{\ell=1}^d k_\ell \le k$).

   For example, let $p^{(j)}(x_\ell)$ be polynomial family of order j in the direction ℓ, then the tensorial polynomial basis of order $k = 2$ in two dimensions would be given by

$$\begin{aligned} \mathbb{P}^2 = \text{Span} \{ & p^{(0)}(x_1)p^{(0)}(x_2), \ p^{(0)}(x_1)p^{(1)}(x_2), \ p^{(0)}(x_1)p^{(2)}(x_2), \\ & p^{(1)}(x_1)p^{(0)}(x_2), \ p^{(1)}(x_1)p^{(1)}(x_2), \\ & p^{(2)}(x_1)p^{(0)}(x_2)\} \end{aligned}$$

2. The full tensorial product of k-order one-dimensional polynomials denoted $\mathbb{Q}^k$ ($k_\ell \le k, \ \forall \ell \le d$), meaning that the polynomials can have simultaneously the maximum order k in each dimension.

   Similar to the previous case, let $p^{(j)}(x_\ell)$ be polynomial family of order j in the direction ℓ, then the tensorial polynomial basis of order $k = 2$ in two dimensions would be given in this case by

$$\mathbb{Q}^2 = \text{Span} \{p^{(0)}(x_1)p^{(0)}(x_2), \ p^{(0)}(x_1)p^{(1)}(x_2), \ p^{(0)}(x_1)p^{(2)}(x_2),$$
$$p^{(1)}(x_1)p^{(0)}(x_2), \ p^{(1)}(x_1)p^{(1)}(x_2), \ p^{(1)}(x_1)p^{(2)}(x_2),$$
$$p^{(2)}(x_1)p^{(0)}(x_2), \ p^{(2)}(x_1)p^{(1)}(x_2), \ p^{(2)}(x_1)p^{(2)}(x_2)\}$$

2.3 Discontinuous Galerkin semi-discrete formulation

A discontinuous Galerkin formulation considers the case in which the continuity condition at the cell interfaces inside the domain is relaxed, therefore the interface terms (which result from the integration by parts in the weak representation) do not vanish. Then, the semi-discrete scheme (2.9) for the continuous case would be modified as

$$\sum_{\ell=1}^{N} \left[\int_{\tau_\ell} \left[v \partial_t \mathbf{q}_h - \mathbf{F}(\mathbf{q}_h) \nabla_x v \right] d\mathbf{x} - \sum_{\alpha \subseteq \partial \tau_\ell} \int_\alpha \left[\mathbf{F}(\mathbf{q}^+)v^+ - \mathbf{F}(\mathbf{q}^-)v^- \right] \cdot \mathbf{n}_\alpha d\sigma \right] = 0,$$
(2.14)

where $v \in \mathcal{V}_h^k$, the sum of the integrals over the cells τ_ℓ represents the integration over the whole domain, and the sum over α refers to the surface integral over all the interfaces of each cell. The super-index $+$ and $-$ in the functions $\mathbf{q}^+$, v_h^+ $\mathbf{q}^-$ and v_h^- of the surface integrals are defined such that

$$\boldsymbol{u}^\pm = \lim_{\varepsilon \to 0} \boldsymbol{u}(t, \mathbf{x} - \varepsilon \mathbf{n}_\alpha^\pm)$$

in which the signs $+$ and $-$ in the normal vector $\mathbf{n}_\alpha^\pm$ stand for the outward and the inward of the cells respectively with respect to the edge α. Several different aspects have to be considered in the formulation of a scheme for equation (2.14).

- how to solve the volume integrals,

- how to treat the values at the cell interfaces,

- which basis function space should be selected,

- and how to perform the time integration.

These different aspects will be explained in the following sections.

2.3.1 The volume integrals

Following the Galerkin approach, the test functions are taken from the same space of basis functions, and by using the vector notation introduced in (2.12) and (2.13) the first integrand of equation (2.14) can be written for every state of the vector $\mathbf{q}_h$ as

$$\int_{\tau_\ell} \left(\partial_t \hat{\mathbf{Q}}_i(t) \cdot \mathbf{v}^\ell(\mathbf{x}) \right) \mathbf{v}^\ell(\mathbf{x}) d\mathbf{x} = \mathbf{M}^\ell \left(\partial_t \hat{\mathbf{Q}}_i(t) \right), \quad \forall i = 1, ..., m, \qquad (2.15)$$

where the entries of the matrix $\mathbf{M}^\ell$ are given by

$$\mathbf{M}_{\alpha,\beta}^\ell = \int_{\tau_\ell} v_\alpha^\ell(\mathbf{x}) v_\beta^\ell(\mathbf{x}) d\mathbf{x}, \quad \forall 1 \le \alpha, \beta \le N_p$$

and $\mathbf{M}^\ell$ is known as the **mass matrix**.

In continuous finite elements the mass matrix tends to be quite large, since it includes all the basis functions of all the cells in the domain, due to the continuity condition that has to be satisfied at the cell interfaces. For the DG method the story is quite different. Since there is no continuity condition, the mass matrices are local operators for every cell, which makes them independent from the neighboring cells.

Moreover, for structured Cartesian meshes using the mappings defined in (2.2) and (2.3), a single mass matrix can be defined in the reference domain. This means that the mass matrix is then unique when the polynomials basis have the same degree over the whole domain.

Considering now the second integrand of equation (2.14), and note that for systems of conservation laws

$$\mathbf{F}(\mathbf{q}) = \left[f_{\xi,\eta}(\mathbf{q}) \right], \quad 1 \le \xi \le m, \quad 0 \le \eta \le d,$$

is a matrix of functions, where m is the dimension of the state vector $\mathbf{q}$. For simplicity in the notation, the sub-indices of a single component $f_{\xi,\eta}(\mathbf{q})$ of $\mathbf{F}(\mathbf{q})$ will be omitted.

Due to the non-linear dependency of the components $f(\mathbf{q})$ of the flux function on the states $\mathbf{q}$, the procedure applied in the previous case cannot be used here. However, two different ways can be followed in order to integrate this term.

The first method is proposed in [37], and it assumes that the numerical approximation $f_h(\mathbf{q}_h)$ to the functions $f(\mathbf{q})$ can be also written as a linear combination of the same basis function used for $\mathbf{q}$ as follows

$$f_h(\mathbf{q}_h) = \sum_{j=1}^{N_p} \hat{f}_j(\hat{\mathbf{Q}}(t)) v_j^\ell(\mathbf{x}), \tag{2.16}$$

where $\hat{f}_j(\hat{\mathbf{Q}}(t)) := \hat{f}_j\left(\hat{\mathbf{Q}}_1(t), ..., \hat{\mathbf{Q}}_m(t) \right)$ are time dependent coefficients. A vector representation can also be defined here as

$$f_h(\mathbf{q}_h) = \hat{\mathbf{F}}(\hat{\mathbf{Q}}(t)) \cdot \mathbf{v}^\ell(\mathbf{x}), \tag{2.17}$$

where $\hat{\mathbf{F}}(\hat{\mathbf{Q}}(t)) = \left[\hat{f}_1(\hat{\mathbf{Q}}(t)), ..., \hat{f}_{N_p}(\hat{\mathbf{Q}}(t)) \right]^T$ is the vector of time dependent coefficients of the polynomial approximation. Then, in a similar fashion as before one could write the second integral of equation (2.14) for all as

$$\int_{\tau_\ell} \left(\hat{\mathbf{F}}(\hat{\mathbf{Q}}(t)) \cdot \mathbf{v}^\ell(\mathbf{x}) \right) \partial_{x_\kappa} \mathbf{v}^\ell(\mathbf{x}) d\mathbf{x} = \mathbf{S}^{\ell,\kappa} \hat{\mathbf{F}}(\hat{\mathbf{Q}}(t)), \quad \forall 1 \le \kappa \le d, \tag{2.18}$$

where $\mathbf{S}^{\ell,\kappa}$ is known as the **stiffness matrix**, and its components are given by

$$\mathbf{S}^{\ell,\kappa}_{\alpha,\beta} = \int_{\tau_\ell} \mathbf{v}_\alpha(\mathbf{x}) \partial_{x_\kappa} \mathbf{v}_\beta(\mathbf{x}) d\mathbf{x}$$

A second method that can be used, guarantees higher order accuracy in the solution. It consists of a numerical integration over the whole expression using

a Gauss-type quadrature rule [61]. The integral of each component of the flux function is then replaced by

$$\int_{\tau_\ell} f(\mathbf{q}) \partial_{x_\kappa} v_j(\mathbf{x}) d\mathbf{x} = \sum_{r=1}^{p} w_r f(\mathbf{q}(t, \mathbf{x}_r)) \partial_{x_\kappa} v_j(\mathbf{x}_r) \qquad (2.19)$$

where $\mathbf{x}_r$ are the quadrature nodes and w_r the associated quadrature weights. Here, p is selected such that the quadrature is exact for polynomials of order $2k$ over the cell. According to [12] this should be enough to achieve $k + 1$ order of accuracy.

This last method (using numerical integration) will be used in the present work for the implementation of test cases in later chapters.

2.3.2 Surface integrals

Consider the case in which the test function is taken as $v = 1$. Then, the discontinuous Galerkin scheme in 2.14 will be reduced to

$$\sum_{\ell=1}^{N} \left[\int_{\tau_\ell} [\partial_t \mathbf{q}_h] \, d\mathbf{x} - \sum_{\alpha \subseteq \partial \tau_\ell} \int_\alpha [\mathbf{F}(\mathbf{q}_h^+) - \mathbf{F}(\mathbf{q}_h^-)] \cdot \mathbf{n}_\alpha d\sigma \right] = 0, \qquad (2.20)$$

which resembles a FD scheme. This equation is not in conservation form since in general the surface integrals at the interfaces of the cells do not cancel out as in the continuous case. It is well known that in order to converge to a meaningful weak solution, a FD scheme has to be in conservation form according to the Lax-Wendroff theorem ([30] pag. 100).

Thus, it becomes necessary to modify the flux functions $\mathbf{F}(\mathbf{q}^+)$ and $\mathbf{F}(\mathbf{q}^-)$ in equation (2.20) in order to obtain a conservative scheme. The easiest way to do this is by using the so called central flux

$$\hat{\mathbf{h}}(\mathbf{q}_h^+, \mathbf{q}_h^-) = \frac{1}{2}[\mathbf{F}(\mathbf{q}_h^+) + \mathbf{F}(\mathbf{q}_h^-)].$$

Then the surfaces integrals in (2.20) will look like

$$\sum_{\alpha \subseteq \partial \tau_\ell} \int_\alpha \frac{1}{2}[\mathbf{F}(\mathbf{q}_h^+) + \mathbf{F}(\mathbf{q}_h^-)] \cdot \mathbf{n}_\alpha d\sigma,$$

which vanish when the sum over the whole domain is performed. However, it turns out that this is also not enough, since it is also well known that the central scheme may converge to a non-physically relevant weak solution (ca. [47] Remark 2.3.2).

If the set of basis functions in the DG scheme 2.20 is also chosen to be $V_h^0 = \{1\}$, it degenerates to a monotone scheme if the flux functions $\mathbf{F}(\mathbf{q}_h^+)$ and $\mathbf{F}(\mathbf{q}_h^-)$ are replaced by a monotone flux $\hat{\mathbf{h}}(\mathbf{q}_h^+, \mathbf{q}_h^-)$.

This motivates the use of monotone schemes, which converge to the unique entropy solution for scalar conservation laws ([47] Theorem 2.3.19) when they satisfy the following characteristics

- *consistency*:
  $\hat{h}(q^-, q^+) = f(q)$, when $q^- = q^+ = q$;

- *monotonicity*:

  $\hat{h}$ is non-decreasing with respect to the first argument, and a non-increasing with respect to the second argument $\hat{h}(\uparrow,\downarrow)$, and

- *Lipschitz continuity*:

  meaning that for all q_1^-, q_2^-

  $$\left|\hat{h}(q_1^-,\cdot) - \hat{h}(q_2^-,\cdot)\right| \leq L_- \left|q_1^- - q_2^-\right|,$$

  and for all q_1^+, q_2^+

  $$\left|\hat{h}(\cdot,q_1^+) - \hat{h}(\cdot,q_2^+)\right| \leq L_+ \left|q_1^+ - q_2^+\right|.$$

- *Normal criteria*:

  in addition according to [18] (page 206), for higher dimensional problems it has to be satisfied on a cell interface, that the numerical flux should have the same magnitude when it is evaluated from both sides, and they only differ in their normal direction, meaning

  $$\hat{\mathbf{h}}(\mathbf{q}_h^+,\mathbf{q}_h^-)\mathbf{n}_- + \hat{\mathbf{h}}(\mathbf{q}_h^-,\mathbf{q}_h^+)\mathbf{n}_+ = 0.$$

These kind of schemes can in general be written as a sum of a central scheme (which alone is unstable, as stated before) with a dissipation (or stabilization) term which mimics a second order derivative, as follows

$$\hat{h}(q^+,q^-) = \frac{(f^+ + f^-)}{2} - D\frac{(q^+ - q^-)}{2}. \tag{2.21}$$

Here, the dissipation coefficient (or matrix) D is a design parameter, which comes from the characteristic values, and moreover it will also depend on the test case and the properties that would like to be preserved.

   One example of a monotone scheme is the Godunov scheme, which uses the exact solution of the Riemann problem at the cell interfaces. The drawback in the use of this particular type of flux is that, as soon as systems of conservation laws are considered, the complexity of the Riemann problem increases. Therefore, in systems of conservation laws, it is more common to use approximate Riemann solvers. Some examples can be found in [77].

Remark: Strictly speaking, it would be necessary to solve the so called **generalized Riemann problem**, which can be roughly defined, for the scalar case, as the initial value problem

$$\begin{cases} \partial_t q + \partial_x f(q) = 0, \\ q(0,x) = q_0(x), \end{cases} \tag{2.22}$$

subject to the initial condition

$$q_0(x) = \begin{cases} P(x^-) & \text{if } x < b, \\ P(x^+) & \text{if } x > b, \end{cases} \tag{2.23}$$

where $P_{\pm}(x)$ are the smooth polynomial approximations inside the cells, and b represents the value of x where the interface between cells is located. The structure of the solution is determined by the **associated Riemann problem**:

$$\begin{cases} \partial_t q^A + \partial_x f\left(q^A\right) = 0, \\ q^A(0,x) = q_{\pm}, \quad \pm x > 0. \end{cases} \tag{2.24}$$

where $q_{\pm}$ are the limit values of $P_{\pm}(x)$ at $x = b$, i.e. $q_{\pm} = P_{\pm}(b^{\pm})$. According to [64], the wave configuration of the generalized Riemann problem (2.22) and (2.23) is the same as for its associated Riemann problem (2.24).

In any case, there is also no interest in this work in exactly solving the generalized Riemann problem or its associated Riemann problem at the interface of the cells. Instead, a so called approximate Riemann solver or a numerical flux are used.

2.3.3 Orthogonal basis functions

There is a lot of freedom in the way the set of basis/test functions can be specified for the numerical scheme. The way in which this function space is chosen might affect different aspects of the implementation and performance of the numerical scheme.

An easy way to set this function space would be to simply choose the space of monomial polynomials

$$\mathcal{V}_h^0 = \{1, x, x^2, ..., x^{k-1}\}, \tag{2.25}$$

which is very easy to implement. However, the condition of the mass matrix will be affected, since in this case

$$\mathbf{M}_{i,j} = \frac{1}{i+j+1}\left[1 + (-1)^{i+j}\right] \tag{2.26}$$

it will resemble a Hilbert matrix, which is know to be poorly conditioned [37]. As the order of the polynomial k increases, the mass matrix get closer to be ill-conditioned. Therefore a more convenient choice of the basis functions would be a set of orthogonal functions, which guarantees a better condition of the mass matrix [37]. These polynomial functions are called orthogonal when they satisfy

$$\int_{\tau} \phi_i(\mathbf{x})\phi_j(\mathbf{x})d\mathbf{x} = \begin{cases} c_i^2 & \text{if } i = j \\ 0 & \text{otherwise} \end{cases} \tag{2.27}$$

that is, the L^2 inner product between two different basis functions is zero, otherwise it is the square of its magnitude c_j. This gives as a result that the mass matrix is diagonal, or even the identity matrix in the case in which normalized polynomials are used.

2.3.4 Nodal vs modal approach

Two different approaches can be used in order to define a basis function for the DG scheme, either a **modal** or a **nodal**. The difference is the following

- A **modal** approach uses a family of polynomials (for example the Legendre polynomials) and the degrees of freedom are given by the amount of terms that are used to approximate the solution. The values of the state q in a certain point $\mathbf{x}$ of the domain can be computed by evaluating the linear combination of basis function at the specified point with the time depending coefficients $\hat{q}(t)$ at time t

$$q(t, \mathbf{x}_p) = \sum_{i=1}^{N} \hat{q}(t)\phi(\mathbf{x}_p).$$

- The **nodal** approach, also called collocation method, uses the Lagrange polynomials associated to a set of **nodes** ξ_i which define the degrees of freedom of the approximation. Even though these polynomials are not orthonormal, by a clever selection of the nodes ξ_i (for example, the use of Gauss-Lobatto nodes in like in [37]) the condition of the mass matrix can be improved,

  Another characteristic of this approach is that the vector of time dependent coefficients $\hat{Q}(t)$ contains the values of the state approximation at these nodes, which makes it very useful in case the values at these nodes are needed. The solution at a point $\mathbf{x}_r$ is then computed as

$$q(t, \mathbf{x}_r) = \sum_{i=1}^{N} \hat{q}(\xi_i)\ell_i(\mathbf{x}_r).$$

  where

$$\ell_i(\mathbf{r}) = \prod_{j=1, j \neq i}^{N} \frac{r - \xi_i}{\xi_i - \xi_j}$$

  is the one-dimensional Lagrange polynomial associated with the node ξ_i.

Both representations are equivalent [37], and moreover, a transformation between these two forms is defined through a simple matrix operation by a Vandermode type matrix. This relation might be useful in case certain calculations are easier to compute in one or the other form

2.4   Time integration

So far only the space discretization has been considered in this work, and the semi-discrete scheme can be written as

$$\partial_t \hat{Q} = \mathcal{L}(\hat{Q}) \tag{2.28}$$

where $\hat{Q}$ is the vector of time dependent coefficients of the state variable q_i, $\mathcal{L}(\hat{Q})$ refers to the DG space discretization described before. The aim is to use the method of lines in this work, which solves the space integration for one specific "line" in time converting the PDE (or system of PDE) into an ODE (or system of ODE), so a classical numerical method for ODE can be used.

The simplest explicit time integration method to perform is the forward Euler method, where the state variables are updated as

$$\hat{Q}^{n+1} = \hat{Q}^n + \Delta t \cdot \mathcal{L}(\hat{Q}) \tag{2.29}$$

meaning that the time step $n + 1$ is the state at time step n plus and updated which consists of the space integration $\mathcal{L}(\hat{Q})$ times the length of the time-step Δt.

2.4.1 The Runge-Kutta method

If the full discrete numerical scheme has the property of being total variation bounded (TVB) using the forward Euler method, and the set of admissible solutions to the PDE is a convex set, then it can be proven that a Runge-Kutta (RK) type time integration method can be used and the numerical scheme will also be TVB [70].

This fact comes from the definition of the strong stability preserving (SSP) schemes (Some analysis about this topic can be found in [32, 33, 31]), since the SSP RK method is constructed as a convex combination of forward Euler time steps. The RK method is achieved through the following procedure [11]

1. Set $q_h^{(0)} = q_h^n$.

2. For $i = 1, ..., \mathcal{K}$ compute the intermediate functions

$$q_h^{(i)} = \sum_{l=0}^{i-1} \alpha_{i,l} q_h^{(l)} + \beta_{i,l} \Delta t^n \mathcal{L}\left(\mathbf{q}_h^{(l)}\right),$$

3. Set $q_h^{(n+1)} = q_h^{\mathcal{K}}$.

Moreover, the RK method considered in this work is required to satisfy the following conditions [11]

- If $\beta_{i,l} \neq 0$ then $\alpha_{i,l} \neq 0$,

- $\alpha_{i,l} \geq 0$,

- $\sum_{l=0}^{i-1} \alpha_{i,l} = 1$.

2.4.2 Courant-Friedrichs-Lewy (CFL) condition

Due to the adoption of the method of lines to solve conservation laws through discontinuous Galerkin, it is possible to implement an explicit scheme like Runge-Kutta as it was explained in this subsection.

Also, one of the properties of the conservation laws PDE is its finite speed of propagation of information, and this speed is given by the eigenvalues of the Jacobian matrix. Thus, the solution in every point of the domain will have a domain of dependence (Figure 2.2a) at previous times and a range of influence (Figure 2.2b) in the future.

A condition for the stability of the numerical scheme emerges, since it has to work faster than the speed of propagation of information. This translates to a restriction that has to be imposed to the time step size, which is called the **Courant-Friedrichs-Lewy condition** (or **CFL-condition**).

The CFL-condition has been already used in FVM and FDM. For discontinuous Galerkin methods, a slightly modification of the condition has to be taken into account, adding also a dependence on the polynomial degree k used

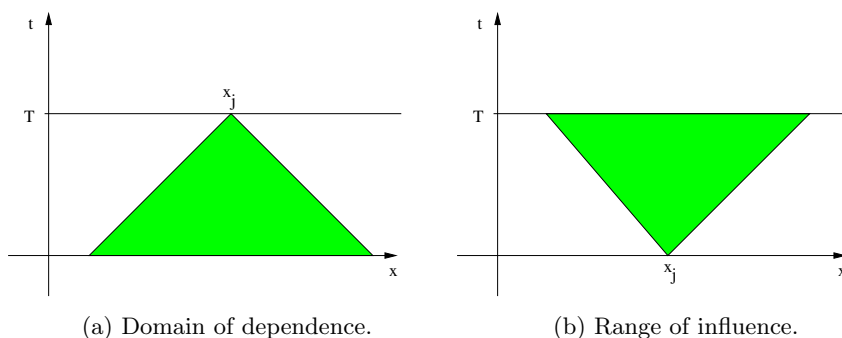(a) Domain of dependence.   (b) Range of influence.

Figure 2.2: 1D illustration of the finite speed of propagation of information in conservation laws. It depends on the maximum eigenvalue of the Jacobian matrix $f'(q)$.

to approximate the solution. The modified version of the CFL-condition is then ([11, 37])

$$\Delta t < \frac{\text{CFL}}{(2k + 1) \sum_i \left| \lambda_{x_i}^{max} \right|} \left| \tau \right|, \tag{2.30}$$

where $|\tau|$ is the spacial volume of the cell, $0 < \text{CFL} < 1$ is a parameter to be selected and $\lambda_{x_i}^{max}$ is the maximum eigenvalue of the Jacobian matrix in direction x_i. This definition is restricted to Cartesian regular grids, so in case of using an unstructured grid a more efficient CFL-condition can be found ([13, 12, 17]).

Summary

- A detailed description of the discontinuous Galerkin discretization in space was introduced in this chapter, including the solution of the different integrals which arise due to the construction of the numerical scheme through the weak representation.

- Some technical details were addressed in the selection of the test functions and a description of the methods used to approximate the generalized Riemann problem at the cell interfaces.

- The Runge-Kutta discretization in time was presented in this chapter, as well as the minimum requirements that has to be satisfied in order to guarantee stability of the numerical scheme.

As it will be seen in the next chapter, some problems arise in the attempt of approximate discontinuous solution with higher order methods like DG. This give place to spurious non-physical oscillations, which have to be controlled.

The following chapter describes some of the techniques that have been traditionally used to stabilize the polynomial approximation around discontinuities and eliminate the spurious oscillations that may appear.

Chapter 3

Dealing with discontinuities

Introduction

As it was described in previous chapters, one characteristic of systems and scalar conservation laws is the possibility of developing discontinuous solutions from continuous initial data. This leads to a huge challenge when these solutions are meant to be approximated numerically.

When a scalar conservation law is approximated via first order numerical schemes, it will converge to the entropy solution but very slowly, since in order to obtain a non-dissipative approximate solution, the grid-size needs to be chosen very small. Figure (3.1) shows an example of how much the size of the elements would have to be reduced in order to have a sharp approximation for the density ρ in the Sod shock-tube test case.

Higher order methods are preferred since they should introduce less dissipation in the numerical solution than the first order method would, thus sharper profiles can be achieved and therefore more features in the numerical solutions can be seen.

But perhaps the viscosity introduced by higher order methods is actually too small. The problem arises when a discontinuous solution has to be resolved, and instead of being smeared out through several cells of the grid, non-physical oscillations like Gibbs phenomena, which are known from Fourier analysis (cf. Hewitt and Hewitt [38]), appear in the place where the discontinuity should be located. An illustration of this phenomena is shown in Figure 3.2

Some of the techniques that have been traditionally applied in order to deal with discontinuities in DG methods will be described in this chapter, including limiting procedures and numerical viscosity. Moreover, some of these techniques will be applied in later chapters in numerical test cases.

3.1 Limiting procedures

Limiting, which will be referred to as the operator $\Lambda\Pi(\cdot)$, is a post-processing or projection procedure and can be applied on the numerical approximation $\mathbf{q}_h$ after certain time level is computed. It works in two steps:

1. Each state of the numerical approximation in every cell is analyzed, in order to determine whether it shows non-physical oscillations. This is
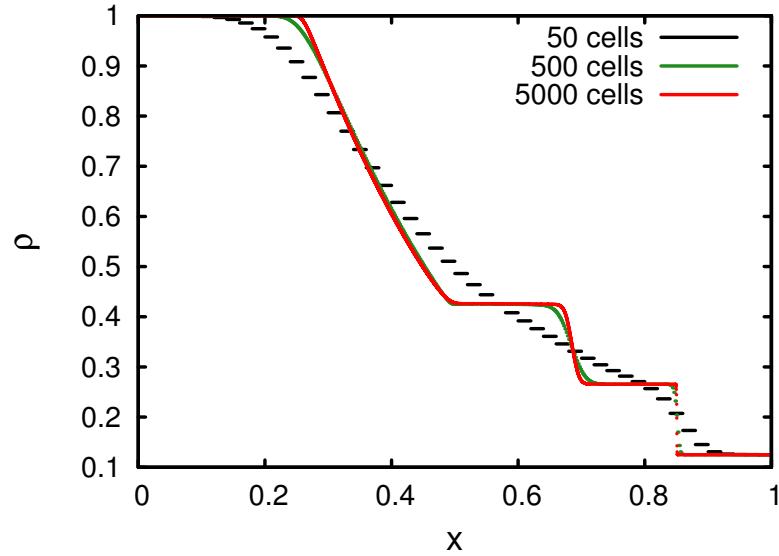
Figure 3.1: First order approximation of the density in the shock tube test case for different cell sizes for the grid.
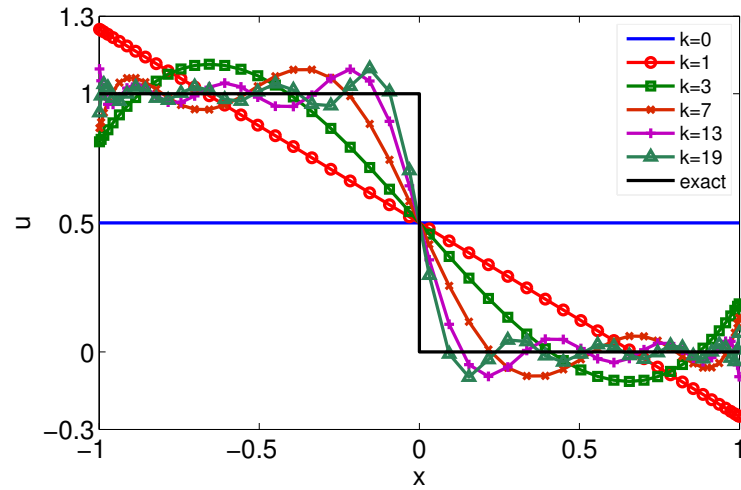

Figure 3.2: Illustration of the Gibbs phenomena at $x = 0$ as the function $u(x) = -sign(x)$ is approximated through a linear combination of Legendre polynomials in the interval $[-1, 1]$. The maximum order of the polynomial approximation is given by k.

done through an indicator function, which will flag the cell as a "trouble cell" in case it is triggered.

2. If the indicator function is triggered, the projection procedure is applied on the "trouble cell", from the original numerical solution $\mathbf{q}_h$ to a non-oscillatory modified solution, defined as $\mathbf{q}_h^{mod} := \Lambda\Pi(\mathbf{q}_h)$.

All the limiting procedures presented in this section will be explained in the one dimensional case, since their extension to higher dimensions is fairly simple when Cartesian grids are used.

As it was pointed out before, the states of the approximated solution $\mathbf{q}_h(t, \mathbf{x})$ in a cell are constructed as a linear combination of polynomial basis functions. Therefore, they can also be written as

$$q_{h,i}(t, x) = \sum_{l=0}^{k} q_i^{(l)}(t)\phi_l^i(x), \tag{3.1}$$

in τ_i, where k is the number of basis functions. Here it is assumed that $\mathbb{P}^k$ polynomial basis are being used, therefore (3.1) can be read as a linear combination of different modes, where $l = 0$ is the mean value of the state, $l = 1$ represent the linear component, $l = 2$ the quadratic component, and so on. Then, the approximated solutions at the interfaces from the interior of de cell are given by

$$q_{i+\frac{1}{2}}^- = q_i^{(0)} + \tilde{q}_i, \quad q_{i-\frac{1}{2}}^+ = q_i^{(0)} - \tilde{\tilde{q}}_i,$$

where, in one dimension, the subindices $i-\frac{1}{2}$ and $i+\frac{1}{2}$ refer respectively to the left and right interface points of the cell τ_i, and the indices − and + refer respectively to the limits from the left or from the right of each interface, meaning that for all $i = 1, ..., N$

$$q_a^{\pm} := \lim_{\varepsilon \to 0} q_{h,i}(t, x_a \pm \varepsilon),$$

for some $x_a \in \tau_i$. The values $\tilde{q}_i$ and $\tilde{\tilde{q}}_i$ are given by

$$\tilde{q}_i = \sum_{l=1}^{N_p} q_i^{(l)}\phi_l^i(x_{i+\frac{1}{2}}), \quad \tilde{\tilde{q}}_i = -\sum_{l=1}^{N_p} q_i^{(l)}\phi_l^i(x_{i-\frac{1}{2}}).$$

This means that $\tilde{q}_i$ and $\tilde{\tilde{q}}_i$ are the linear combination of the higher order modes at the interfaces of the cell, which represent the distances between the mean value of the cell and the approximated solution at the cell interfaces. Figure (3.3) illustrates the meaning of these quantities.

3.1.1 TVB type limiter

This type of limiting was proposed throughout the work of Cockburn and Shu in the series of paper [15], [13], [12] and the review paper [18]. This limiter revises the approximated solution such that its cell average values become TVB stable, and it reduces the high order polynomials to linear polynomials in "trouble" cells where artificial oscillations appear.

A basic criteria is implemented such that the high order polynomial is not modified in case the cell is not flagged as a "trouble" cell, thus the numerical scheme does not lose its high order accuracy by the limiter. The limiting criteria
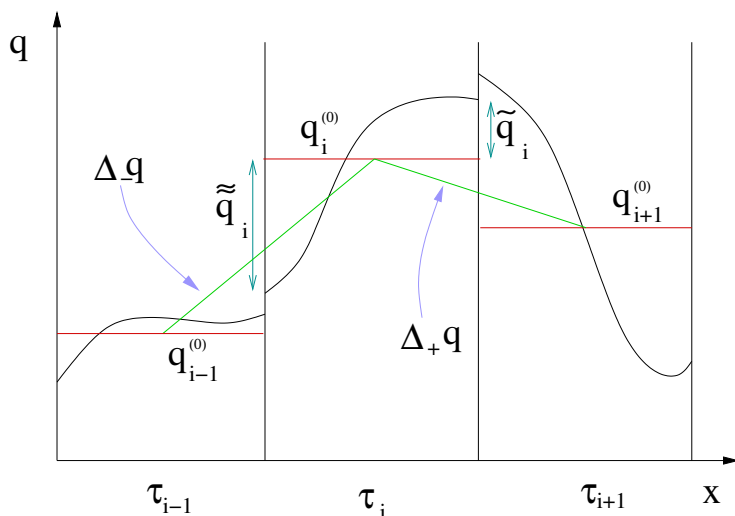
Figure 3.3: Illustration of some quantities used for the limiting procedures, in particular for the TVB limiting.

relies entirely on the mean values of the states $q^{(0)}$, both inside the cell τ_i and at the neighboring cells τ_{i+1} and τ_{i-1}, for a specific direction. To explain this procedure, let the forward and backward differences of the mean values be respectively defined as

$$\Delta_+ q_i^{(0)} = q_{i+1}^{(0)} - q_i^{(0)}, \quad \Delta_- q_i^{(0)} = q_i^{(0)} - q_{i-1}^{(0)}.$$

The boundary values are then changed by for

$$\tilde{q}^{mod} = \tilde{m}\{\tilde{q}, \Delta_+ q_i^{(0)}, \Delta_- q_i^{(0)}\}, \quad \tilde{\tilde{q}}^{mod} = \tilde{m}\{\tilde{\tilde{q}}, \Delta_+ q_i^{(0)}, \Delta_- q_i^{(0)}\}.$$

The function $\tilde{m}$ is the modified minmod function given by

$$\tilde{m}(a_1, ..., a_n) = \begin{cases} a_1, & \text{if } |a_1| \le Mh^2 \\ m(a_1, ..., a_n), & \text{otherwise,} \end{cases} \tag{3.2}$$

where the parameter $M \ge 0$ has to be determined, and $m(a_1, ..., a_n)$ is the minmod function defined as

$$m(a_1, ..., a_n) = \begin{cases} s \min_{1 \le j \le n} |a_j|, & \text{if } sign(a_1) = ... = sign(a_n) = s, \\ 0, & \text{otherwise,} \end{cases} \tag{3.3}$$

The indicator criteria of the modified minmod function is given by the comparison $|a_1| \le Mh^2$, therefore the selection of the value for the only parameter M has a huge impact on the performance of this limiter. The lower the value of M, the more restrictive the limiter becomes.

In the context of scalar conservation laws, some selection options of the parameter M where presented in [15]. This selection options use the fact that the solution of a scalar conservation law satisfies a maximum principle. But this is not the case for arbitrary systems of conservation laws, in which situation M has to be chosen carefully by intuition.

One could simply select the parameter M as zero, but this will certainly destroy the accuracy of the numerical solution in regions where the solution is smooth but a local extrema is located.

3.1.2 $\mathbb{Q}^k$ polynomial basis limiter

When $\mathbb{P}^k$ polynomial basis functions are used, the zero order mode $q^{(0)}$ is the mean value of the state in the cell. Therefore, the first order approximation can be obtained just by eliminating the higher order terms of the polynomial.

For a $\mathbb{Q}^k$ polynomial basis one must follow a different procedure. The main difference of this limiting procedure is that the average gradients of the states, calculated as

$$\overline{\nabla q_i} = \frac{1}{|\tau_\ell|} \int_{\tau_\ell} \nabla q_i dx$$

are compared component wise with the averaged values of the state variables $\overline{\mathbf{q}}$ in the element τ_ℓ. The modified average gradient $\nabla q_i^{(m)} = \left[\partial_x q_i^{(m)}\right]^T$ is computed as

$$\partial_x q_i^{(m)} = \tilde{m}\left(\overline{\partial_x q_i}, \beta \frac{\Delta_- \overline{q}_i}{\Delta x}, \beta \frac{\Delta_+ \overline{q}_i}{\Delta x}\right)$$

where $\beta \in [1, 2]$, Δx and $\tilde{m}(a_1, ..., a_d)$ is again the modified minmod function defined in equation (3.2) introduced by Cockburn and Shu [17]. If $\partial_x q_i^{(m)} \neq \overline{\partial_x q_i}$, then the solution q_h must be reduced to the following linear polynomial $\Lambda \Pi_h(q_h)(x)$

$$\Lambda \Pi_h(q_h)(x) = \overline{q_i} + (x - x_i)\, \partial_x q_i^{(m)} \tag{3.4}$$

Some numerical simulations were published in [25] by the author of this work and some collaborators, using this gradient based limiter with competitive results.

3.1.3 Moment coefficients limiter

This limiter is also known as BDF limiter and it was introduced by Biswas, Devine and Flaherty in [5]. It is a limiter that works by first reducing the magnitude of the higher order terms (or moments) where the solution is thought to show spurious oscillations. The coefficient of the moment l is modified as follows

$$q_i^{(l),mod} = \frac{1}{2l-1} m\left((2l-1)q_i^{(l)}, \Delta_+ q_i^{(l)}, \Delta_- q_i^{(l)}\right), \quad 1 \leq l \leq k \tag{3.5}$$

where the index (l) refers to the order of the polynomial term, and k is the maximum order of the basis functions, and once again m is the minmod function defined in equation (3.3). This method is applied adaptively, starting from the highest to the lowest moment in each cell, so the difference with the TVB limiter is that (in theory) it should preserve the higher order accuracy.

An extension of this limiter for two dimension problems was introduced by Krivodonova in [45], and later Kuzmin in [49] presented an extension for implicit time integration and general geometries using Taylor polynomial basis.

Also, a modification of this limiter was presented in [7] by Burdeau, Sagaut and Bruneau, where the coefficients of the moments are limited by the coefficients of the moments of one lower order, as follows

$$q_i^{(l),m} = \frac{1}{2l-1} m\left((2l-1)q_i^{(l)}, \Delta_+ q_i^{(l-1)}, \Delta_- q_i^{(l-1)}\right), \quad 1 \le l \le k \qquad (3.6)$$

If $q_i^{(l),m} \ne q_i^{(l)}$ the moment $q_i^{(l)}$ is replaced by

$$q_i^{(l),mod} = maxmod\left(q_i^{(l),m}, q_i^{(l),max}\right),$$

and the $maxmod$ function is defined as

$$maxmod\,(a_1,...,a_N) = \begin{cases} s \cdot max_{1 \le j \le n} |a_j|, & \text{if } sign(a_1) = ... = sign(a_n) = s, \\ 0, & \text{otherwise,} \end{cases}$$

where

$$q_i^{(l),max} = \frac{1}{2l-1} m\left((2l-1)q_i^{(l)}, q_{i+\frac{1}{2}}^{(l-1)+} - q_i^{(l-1)}, q_i^{(l-1)} - q_{i-\frac{1}{2}}^{(l-1)-}\right),$$

$$q_{i+\frac{1}{2}}^{(l-1)+} = q_{i+1}^{(l-1)} - (2l-1)q_{i+1}^{(l)},$$

$$q_{i-\frac{1}{2}}^{(l-1)-} = q_{i-1}^{(l-1)} - (2l-1)q_{i-1}^{(l)},$$

which means that, the impact or size of a mode (l) will be also bounded by the difference between the coefficients of the two different modes (l) and $(l-1)$ in the neighboring cells.

3.1.4   Monotonicity-preserving (MP) type limiter

The MP limiter was introduced by Seresh and Huynh in [76] as a monotone forth order reconstruction for a finite volume scheme. It is based on the median values calculated as

$$median(x,y,z) = x + m(y-x, z-x),$$

where m is the minmod function. The modified values at the boundaries are then

$$q_{i+\frac{1}{2}}^{-,mod} = median(q_{i+\frac{1}{2}}^-, q_{i+\frac{1}{2}}^{min}, q_{i+\frac{1}{2}}^{max}),$$

where

$$q_{i+\frac{1}{2}}^{max} = max\left[min\left(q_i^{(0)}, q_{i+1}^{(0)}, q_{i+\frac{1}{2}}^{MD}\right), min\left(q_i^{(0)}, q_{i+1}^{UL}, q_{i+\frac{1}{2}}^{LC}\right)\right],$$

$$q_{i+\frac{1}{2}}^{min} = min\left[max\left(q_i^{(0)}, q_{i+1}^{(0)}, q_{i+\frac{1}{2}}^{MD}\right), max\left(q_i^{(0)}, q_{i+1}^{UL}, q_{i+\frac{1}{2}}^{LC}\right)\right]$$

and

$$d_i = q_{i+1}^{(0)} - 2q_i^{(0)} + q_{i-1}^{(0)},$$

$$d_{I+\frac{1}{2}}^{M4X} = m\left(4d_i - d_{i+1}, 4d_{i+1} - d_i, d_i, d_{i+1}, d_{i-1}, d_{1+2}\right),$$

$$q_{i+\frac{1}{2}}^{MD} = \frac{1}{2}\left(u_i^{(0)} + u_{i+1}^{(0)} - d_{i+\frac{1}{2}}^{M4X}\right),$$

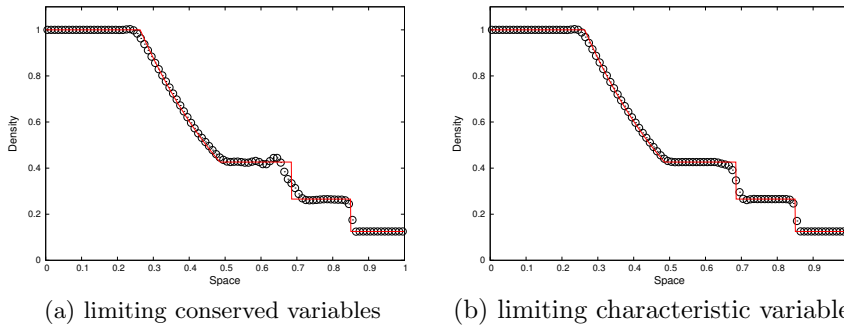(a) limiting conserved variables   (b) limiting characteristic variables

Figure 3.4: Comparison between the limiting procedure over (a) the characteristic variables and (b) the conserved variables for the one dimensional shock tube test case.

$$q_{i+\frac{1}{2}}^{UL} = u_i^{(0)} + \alpha \left(u_i^{(0)} - u_{i-1}^{(0)} \right),$$

$$q_{i+\frac{1}{2}}^{LC} = u_i^{(0)} + \frac{1}{2} \left(u_i^{(0)} - u_{i-1}^{(0)} + \frac{\beta}{3} - d_{i-\frac{1}{2}}^{M4X} \right),$$

where UL stands for upper limit, MD stands for median and LC stands for large curvature. All these names come from geometrical relations used to develop the limiter. The $q_{i-\frac{1}{2}}^{+,mod}$ value can be found in a similar fashion as the one just presented.

A modification of this limiter is shown in [66], where the moments are limited to

$$q_i^{(l),mod} = \varphi_i q_i^{(l)}, \quad 1 \le l \le k,$$

where

$$\varphi_i = min \left(1, \frac{1}{\Delta_{min} q_i} \left| q_i^{(0)} \right| \right), \quad \Delta_{min} q_i = q_i^{(0)} - min_{\mathbf{x} \in \tau_i} q(t, \mathbf{x})$$

which relaxes the monotonicity preserving constraint, and enforces the sign preservation.

3.2 Characteristic limiting

The post-processing limiting procedure can be directly applied on the conserved variables $\mathbf{q}$. However, some experiments had led to the conclusion that it would produce more accurate solutions when it is applied on the characteristic variables of the system (see Figure (3.4)).

Now, in order to apply the limiting procedure to the characteristic variables $\mathbf{z}$, the system conservation law has to be written first in quasilinear form, as explained in Section 1.1, obtaining then the Jacobian matrix $\mathbf{A}$. Then, the right and left eigenvector matrices, $\mathcal{R}$ and $\mathcal{L} = \mathcal{R}^{-1}$ respectively, must be computed from the Jacobian matrix $\mathbf{A}$. As it was shown in section 1.1.1, a decomposition $\mathbf{A} = \mathcal{R} \mathbf{\Lambda} \mathcal{L}$ can be found, reaching an equation similar to (1.8), which is

$$\begin{cases} \partial_t \mathbf{z} + \mathbf{\Lambda}(\mathbf{q}) \partial_x \mathbf{z} = 0, \\ \mathbf{z}(0, \mathbf{x}) = \mathbf{z}_0(\mathbf{x}), \end{cases} \tag{3.7}$$

note the dependency of the matrix $\mathbf{\Lambda}$ on the vector of states $\mathbf{q}$.

Several important remarks have to be pointed out here, the first one being the fact that a method to diagonalize linear systems was used in a framework in which there is no hint about the linearity of the conservation laws system. The second remark is that in several dimensions more than one Jacobian matrix appear. Hence, in order to transform the system in the form (3.7), it is necessary that the Jacobi matrices have the same basis of Eigenvectors which span the $\mathbb{R}^d$, which is the case if and only if the Jacobian matrices commute.

For the first aspect, the Jacobian matrix, the eigenvalues and the matrix of left and right eigenvectors are computed analytically, and then if the system is non-linear, all these matrices ($\mathbf{A}$, $\mathbf{\Lambda}$, $\mathcal{L}$, $\mathcal{R}$) will have a direct dependency on the vector of states $\mathbf{q}$. To make these matrices linear, they are usually evaluated in some local averaging quantities for each cell, for example, the mean values $\bar{\mathbf{q}}$ of the states inside each cell. Therefore, the eigensystem used in the transformation of course has the same structure, but it is locally different, since the values at which it is evaluated may differ from cell to cell.

The second aspect has an intuitive motivation, that is, the limiting techniques applied on Cartesian grids are performed component-wise. This means that the limiting is applied sequentially direction by direction. Even thought the diagonalization should be in general unique, not all the systems under study will commute, and therefore for simplicity in the expansion to higher dimensions, each one of the Jacobian matrices $\boldsymbol{A}_\kappa$ are diagonalized independently, for $\kappa \in \{1, ..., d\}$.

In practice the diagonalized system (3.7) is neither assembled, nor used to solve the spacial integration or in the time evolution. In the multi-dimensional case, the $\mathcal{L}_\kappa$ and $\mathcal{R}_\kappa$ matrices are used just to transform to and from the characteristic variables $\mathbf{z}$, as

$$\mathbf{z} = \mathcal{L}_\kappa(\bar{\mathbf{q}})\mathbf{q} \quad \text{and} \quad \mathbf{q} = \mathcal{R}_\kappa(\bar{\mathbf{q}})\mathbf{z}$$

for every space dimension $\kappa \in \{1, ..., d\}$ and for each cell of the tessellation, so that the limiting procedure to correct spurious oscillations can be applied directly on the information transported along some average direction of the characteristics. The limiting procedure over the characteristic variables can then be summarized in the following steps

- Compute analytically the $\mathcal{L}_\kappa$ and $\mathcal{R}_\kappa$ matrices for every space dimension κ.

- Loop over the "trouble" cells.

  - evaluate the $\mathcal{L}_\kappa$ and $\mathcal{R}_\kappa$ matrices with the local averaged values $\bar{\mathbf{q}}$.
  - for every dimension κ.
    * Compute the characteristic variables as $\mathbf{z} = \mathcal{L}_\kappa(\bar{\mathbf{q}})\mathbf{q}$.
    * Apply the selected limiting procedure over the characteristic variables, and obtain the modified solution $\mathbf{z}^{mod} = \mathbf{\Lambda}\Pi(\mathbf{z})$.
    * If $\mathbf{z} \neq \mathbf{z}^{mod}$, replace $\mathbf{z}$ by $\mathbf{z}^{mod}$.
    * Transform back to conserved variables $\mathbf{q}^{mod} = \mathcal{R}_\kappa(\bar{\mathbf{q}})\mathbf{z}$.

Figure (3.4) shows the solution of a 1-D shock tube problem using limiter on conserved and characteristic variables. As it can be noticed, the characteristic

limiter produces better resolution of discontinuities and also avoids unphysical oscillations.

3.3 Positivity preserving limiter

So far all methods presented in this chapter to control oscillations around discontinuous solutions do not mention specific applications. In this particular case it will be different, since this kind of limiter is important for the numerical implementations of certain systems of conservation laws in later chapters.

It could also be called a bound preserving limiter. It was first introduced by X. Zhang and S.-W. Shu in [83, 84, 85], and it is based entirely on the supposition that the DG scheme satisfy a total variation bounded (TVB) property in the mean values. Essentially, this property ensures that if the mean value of a state $\bar{q}_j$ is between certain values $\left[q^{\min}, q^{\mathrm{Max}}\right]$ at the beginning of the time-step n, it will remain within the same bounds for the following time-step $n+1$. The bound preserving limiter is designed to modify the polynomial around the mean value to stay within the desired bounds, but it does not modify the mean value itself.

It is important to state that the aim of this work is finding physically relevant solutions, namely solutions which belong to an admissible set $\mathcal{U}_{\mathrm{ad}}^{\epsilon}$. In the case of the compressible Euler equations of gas dynamics, or the MHD equations, the values of the pressure and density must be non negative. In case this is not satisfied, the numerical solution $\mathbf{q}_h$ has to be modified such that $\mathbf{q}_h(t,\xi) \in \mathcal{U}_{\mathrm{ad}}^{\epsilon}$, $\forall \xi \in Q$, where Q is a set of quadrature nodes inside the cell and

$$\mathcal{U}_{\mathrm{ad}}^{\epsilon} = \{\mathbf{q} \in \mathbb{R}^m : \rho(\mathbf{q}) \geq \epsilon, \quad p(\mathbf{q}) \geq \epsilon\},$$

here m is the number of states variables of the vector $\mathbf{q}$, ρ stands for the density and p for pressure. Another example is the the shallow water system of equations, where an admissible set would ensure that the water level is non-negative. In fact the bounded preserving limiter is especially designed for systems with convex sets of admissible solutions. Some applications also for non isentropic problems can be found in [84], [85] and [65]

It may happen in some cases that the time-step restriction or the post-processing procedures introduced in the previous sections are not enough to prevent a numerical implementation from crashing due to the presence of negative values in certain regions of the numerical approximation $\mathbf{q}_h$. Therefore, the bound preserving limiter can be implemented to ensure a minimum lower value $\epsilon > 0$ for the state variables which need it. An illustration can be seen in Figure (3.5).

In the following, an example of how the limiter ensures that the modified solution belongs to the set $\mathcal{U}_{ad}^{\epsilon}$ for the Euler equations. As it has been mentioned before, the limiter can only be applied when the scheme preserves the bounds of the the cell average of the DG solution. In [84], it has been proven that this can be ensured by a suitable CFL restriction. If this is satisfied, but some values of the density ρ_h are negative, the polynomial solution has to be corrected by $\hat{\rho}_h$, as follows

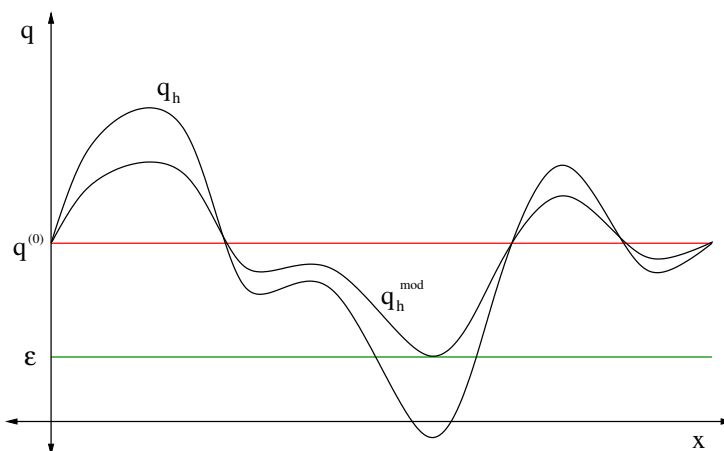$$\hat{\rho}_h(t,\xi) \geq \epsilon, \quad \forall \xi \in Q,$$

Figure 3.5: Illustration of how the positivity preserving limiter works. The polynomial solution q_h is shrank around the mean value $q^{(0)}$ until its minimum value is set to ϵ. The high order accuracy is preserved.

by finding a modified density as follows

$$\hat{\rho}_h = \theta_1(\rho_h - \overline{\rho}) + \overline{\rho}_n,$$

where

$$\theta_1 = \min\left\{\frac{|\overline{\rho} - \epsilon|}{|\overline{\rho} - \rho^{\min}|}, 1\right\} \quad \text{and} \quad \rho^{\min} = \min_{\xi \in Q} \rho_h(\xi).$$

This basically shrinks the polynomial solution around the mean value, as it is illustrated in Figure 3.5, ensuring that every value of the density at the quadrature points $\xi \in Q$ is over the ϵ–tolerance.

After ensuring the positivity of the density, it is necessary also to control the values of the pressure. Denote $\hat{\mathbf{q}}_h$, the new state vector such that the density $\hat{\rho}_h$ is already positive. If $p(\hat{\mathbf{q}}_h(\xi)) < \epsilon$, $\forall \xi \in Q$, a state between $\overline{\mathbf{q}}$ and $\hat{\mathbf{q}}_h(\xi)$ has to be found such that

$$p((1 - t_\xi)\overline{\mathbf{q}} + t_\xi \hat{\mathbf{q}}_h(\xi)) = \epsilon,$$

where t_ξ is the unique solution $t_\xi \in (0, 1)$ of a quadratic equation. Finally the modified solution will be given by

$$\tilde{\mathbf{q}}_h = \theta_2(\hat{\mathbf{q}}_h(t, \mathbf{x}) - \overline{\mathbf{q}}) + \overline{\mathbf{q}}, \qquad \theta_2 = \min_\xi t_\xi.$$

It is important to mention that as it was proven in [84], this limiting procedure does not affect the accuracy of the method. And finally, there are alternative approaches to obtain a positive preserving high order schemes, for example the work of Kuzmin and Turek presented in [50].

3.4   Artificial viscosity approach

A different methodology, which can be followed in order to avoid oscillatory solutions, is to modify the hyperbolic nature of the problem transforming it

into a parabolic problem. This is achieved by including an additional term in the equation, which brings regularity to the solutions by adding artificial viscosity. Then the new system is of the type

$$\partial_t \mathbf{q} + \nabla_{\mathbf{x}} \cdot \mathbf{F}(\mathbf{q}) - \nabla_{\mathbf{x}} \cdot (\varepsilon \nabla_{\mathbf{x}} \mathbf{q}) = 0 \tag{3.8}$$

where ε is the artificial viscosity parameter which has to be given. This idea, which was firstly presented by Von Neuman and Richtmyer in [79], in principle makes sense since the proof of existence of solutions for scalar and certain systems of conservation laws are based on viscous approaches (see section 1.1.4) at the limit $\varepsilon \to 0$.

The big difference in this procedure lays on the necessity of discretizing a second order derivative. This can be handled in different ways, like for example following the Local discontinuous Galerkin method (LDG) introduce in [16], in which the second order derivative is split into two equations as follows

$$\partial_t \mathbf{q} + \nabla_{\mathbf{x}} \cdot \mathbf{F}(\mathbf{q}) - \nabla_{\mathbf{x}} \cdot \mathbf{y} = 0 \tag{3.9}$$

$$\mathbf{y} - \varepsilon \nabla_{\mathbf{x}} \mathbf{q} = 0 \tag{3.10}$$

The use of this approach is shown, for example, in [60] where a LDG scheme is proposed to solve systems of conservation laws, and some test cases are presented solving the compressible Euler equations. A different approach would be to directly write the weak representation as in the case of the conservation laws as follows

$$\sum_{\ell=1}^{N} \left[\int_{\tau_\ell} \left(\partial_t \mathbf{q} + \nabla_x \cdot \mathbf{F}(\mathbf{q}) - \nabla_x \cdot (\varepsilon \nabla_x \mathbf{q}) \right) v(\mathbf{x}) d\mathbf{x} \right] = 0, \tag{3.11}$$

and integrating by parts it follows that

$$\sum_{\ell=1}^{N} \left[\int_{\tau} \left[v \partial_t \mathbf{q}_h - \left(\mathbf{F}(\mathbf{q}_h) - \varepsilon \nabla_x \mathbf{q}_h \right) \nabla_x \cdot v \right] d\mathbf{x} \right.$$
$$\left. + \sum_{\alpha \subseteq \partial \tau_\ell} \int_{\alpha} \llbracket v\mathbf{F} - v\varepsilon \nabla_x \mathbf{q}_h \rrbracket \cdot \mathbf{n}_\alpha d\sigma \right] = 0, \tag{3.12}$$

where $\llbracket \cdot \rrbracket$ is called the jump operator given by $\llbracket \star \rrbracket = \star^+ - \star^-$. This equation involves two additional integrals which are not present in equation (2.14), and which come from the viscous term. The first one is a volume integral over the gradient of the state variable approximation $\mathbf{q}_h$, and the second one is a boundary term, which has to be replaced by an additional numerical flux.

Since the numerical solution $\mathbf{q}_h$ is a linear combination of polynomials, it is not a problem to approximate the first additional integral. The derivatives of the numerical approximation $\mathbf{q}_h$ can be calculated exactly. In the case of the boundary integral, an additional numerical flux will have to be used to approximate these terms.

As a remark, it is worth noting that since $\varepsilon \neq 0$, the space of admissible solutions for $\mathbf{q}(t, \mathbf{x})$ in this case is different to the set of admissible solutions of the conservation laws, since more regularity is needed for the weak representation to make sense.

After choosing a method to solve the new system of equations, the viscosity coefficient ε has to be chosen. The election of this parameter will depend, as in the limiting procedures, on the system of conservation laws that is being

(a) Shock tube viscous approach   (b) Viscous coefficient used
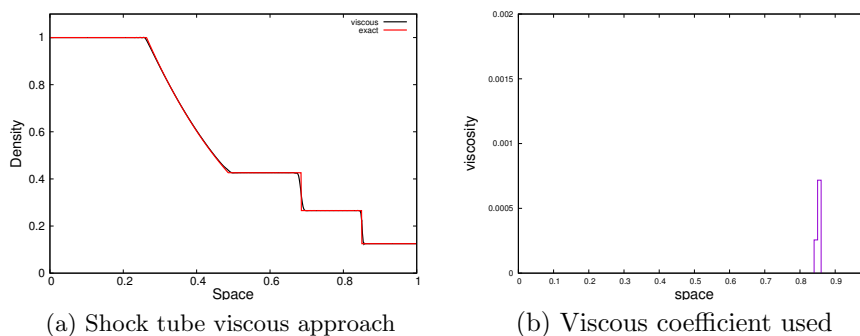
Figure 3.6: Sod shock tube problem using the a DG scheme from equation (3.12), and calculating the viscous coefficient as in [60]. The viscosity is added only at the shock wave.

considered. Different ideas have been proposed in this topic, for example in [60], a LDG scheme is presented in order to solve the Euler equations of gas dynamics. The viscous terms are added in two different ways:

1. either as Laplacian viscosity, which means that it will be added to all equations of the system,

2. or by following a physical fluid model, in which the viscosity is added just in the momentum and energy equations.

The coefficient ε is then modulated through an indicator smooth function, which prevents viscous effects away from discontinuities. This means that in smooth regions, the viscous term is zero, and therefore the solution is calculated over the initial conservation law system.

Another example of how to choose the viscosity coefficient is shown in [86], where a DG scheme like the one shown in equation (3.11) is used. The value of ε is a piecewise constant function over the domain, which is proportional to the entropy residual. This entropy residual is integrated along with the system using a finite difference scheme. Additional studies in this topic can be seen in [81, 3, 34, 22]

For low order schemes the viscous approach can be seen as rather complicated and not very efficient when it is compared to slope limiting procedures. However, in the case of higher order DG schemes the story might be different as presented in [60]. This is due to the fact that the accuracy in the neighborhood of the discontinuities becomes $\mathcal{O}(h/k)$, whereas by using slope limiters it is only $\mathcal{O}(h)$, where h is the size of the cell and k the order of the polynomial.

3.5   Shock indicator

Each one of the limiting procedures and the viscous approach have their own way to decide which cell should be flagged as trouble cell, in order to eliminate oscillations. This indicator can strongly influence the performance of the limiting procedure or viscous approach.

Krivodonova et al. [46] introduced a way to detect trouble cells where discontinuities may occur. It is known as the KXRCF indicator, as an acronym for the names of the authors, and it is based on the fact that the DG approximation shows strong super convergence at the outflow boundary of each element only when smooth solutions are approximated. This was firstly shown by Flaherty et al. in [24].

To explain the underlying concept here, let the boundaries of each single cell τ_ℓ be divided into inflow $(\partial\tau_\ell^-)$ and outflow $(\partial\tau_\ell^+)$ portions. The super-convergence result translates into

$$\frac{1}{\left|\partial\tau_\ell^+\right|} \int_{\partial\tau_\ell^+} (y_h - y)\, d\sigma = \mathcal{O}(h^{2k+1}), \tag{3.13}$$

where the variable y can be either a component of the state vector $\mathbf{q}$, a component of the vector of characteristic variables $\mathbf{z}$ or a function of the states, for example an entropy function of the system of conservation laws. Equation (3.13) means that, at the outflow boundary points, the rate of approximation y_h to a variable y is almost twice the optimal rate of approximation.

The super-convergence for DG methods was proven by S. Adjerid and T. C. Massey in [1] to be satisfied in the so-called Radau points due to the techniques used in the proof. However, as it was studied by Y. Cheng and C-.W. Shu in [10] and by B. Cockburn et al. in [14], there are many other points in which this property is satisfied.

In contrast to the previous case, at inflow boundaries the rate of approximation would be given by

$$\frac{1}{\left|\partial\tau_\ell^-\right|} \int_{\partial\tau_\ell^-} (y_h - y)\, d\sigma = \mathcal{O}(h^{k+1}).$$

Hence, analyzing the jump in the inflow edge $\alpha \subseteq \partial\tau_\ell^-$

$$I_\alpha = \int_\alpha \left(y_h^- - y_h^+\right) d\sigma = \int_\alpha \left(y_h^- - y\right) d\sigma + \int_{\alpha_{nb}} \left(y - y_h^+\right) d\sigma. \tag{3.14}$$

Here y^- and y^+ represent the values at the boundary from inside and outside the cell τ_ℓ respectively, and $\alpha_{nb} = |\alpha|\,\mathbf{n}_{nb} = -|\alpha|\,\mathbf{n}$ is the corresponding outflow boundary of the neighboring cell. The fist integral is then of order $\mathcal{O}(h^{k+2})$, while the second one is of order $\mathcal{O}(h^{2(k+1)})$. This means that in the case of smooth solutions, I_α is of order $\mathcal{O}(h^{k+2})$ across the interface α. If the variable y is discontinuous in a cell, then at least one of the integrals in (3.14) will be of order $\mathcal{O}(h)$. Therefore,

$$I_\alpha = \begin{cases} \mathcal{O}\left(h^{k+2}\right), & \text{if } y\,|_{\partial\tau_\ell} \text{ is smooth,} \\ \mathcal{O}(h), & \text{if } y\,|_{\partial\tau_\ell} \text{ is discontinuous.} \end{cases}$$

As it is shown in [46], the discontinuity detector is then constructed by normalizing the quantity I_α with an a kind of average convergence rate of order $\mathcal{O}\left(h^{(k+1)/2}\right)$ as follows

$$\mathcal{I}_\alpha = \frac{\left|\int_\alpha \left(y_h^- - y_h^+\right) d\sigma\right|}{h^{\frac{k+1}{2}} |\alpha|\, \|y_h\|_{\tau_\ell}}.$$

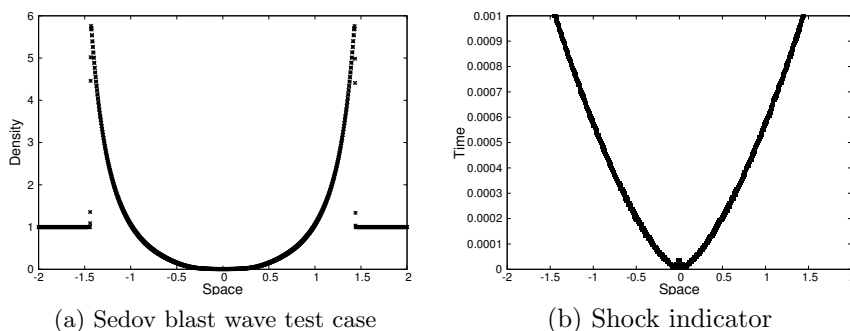(a) Sedov blast wave test case   (b) Shock indicator

Figure 3.7: Sedov blast wave test case in one space dimension at the final time $t = 0.001s$ (a). Shock indicator along all the time evolution (b), the indicator follows the shock fronts accurately.

for $\alpha \subseteq \partial \tau_\ell^-$. This indicator function will go to zero as the grid size is reduced or the polynomial order is increased ($\mathcal{I}_\alpha \to 0$ as $h \to 0$ or $k \to \infty$), and it will go to infinity ($\mathcal{I}_\alpha \to \infty$) near a discontinuity. Thus, the quantity can be interpreted as follows

- If $\mathcal{I}_\alpha < 1$, y is continuous

- If $\mathcal{I}_\alpha > 1$, y is discontinuous

In the case of the Euler equations or the MHD equations, common choices of variables to use for the indicator function are the density, the energy and the physical entropy.

Some experiments were carried out in one-dimensional problems in order to test the performance of the shock indicator, and decide whether it is worth to be implemented for multi-dimensional problems. Figure (3.7) shows the case of the Sedov blast wave test case in one-dimension, and how the shock indicator follows the shock front along the time.

Summary

- Some methods to control spurious oscillations have been introduced in this chapter. These have been traditionally divided in two groups, which are limiting post-processing procedures and numerical viscosity approaches. Both have been widely used in the literature, but each one of them imposes different modifications to the numerical scheme.

- Even if limiting procedures are post-processing steps, it should be noted that they can also be seen as addition of viscosity to the numerical scheme, since it is basically the result of suppressing, or simply reducing the impact of the the higher order terms of the polynomial approximation.

- A limiting procedure on the characteristic variables was presented, where the Jacobian matrix has to be evaluated in some local average values inside each cell.

- Each one of the limiting procedures and viscous approaches have a indicator function which trigger the implementation of the method. An additional shock indicator was introduced based on the lost of superconvergence at the outflow boundaries of the cells.

- A positivity preserving limiter was introduced keeping in mind on the implementations that will be carried out in the following chapters. Its use may be restricted to problems with near vacuum conditions, where the normal limiting procedures do not guarantee the positivity of the polynomial approximation. It may be applied just when the positivity of the mean value of the variable is guaranteed in order to operate correctly. For the RK-DG method this can be achieved by a slight modification of the CFL restriction. The values for the CFL restriction can be found in [84], Table 2.1.

In the following chapter, the details related to the implementation of the numerical scheme will be shown, plus different test cases for both systems of conservation laws under study.

Chapter 4

Numerical experiments

Introduction

In the previous chapters, a general framework to find numerical solutions for hyperbolic systems of conservation laws using the discontinuous Galerkin method was introduced. Additionally some techniques commonly used in the literature to deal with oscillations around discontinuities of the solutions were presented.

This chapter shows, through different two-dimensional test cases, the application of some of the concepts previously introduced on two specific systems of equations which are relevant for astrophysical applications. First, the Euler equations presented in Section 1.2.1, which describe an inviscid hydrodynamical compressible flow, will be considered.

The second system under study can be considered as an extension of the Euler system, since in addition to the conservation of mass, momentum and energy, it considers both the conservation of the magnetic field and its divergence free condition. This second PDE system is the ideal Magnetohydrodynamical (MHD) model, which was already introduced in Section 1.2.2.

In the following sections the test cases will be explained together with the numerical results obtained through a C++ implementation developed using the `deal.II` C++ libraries [2]. Some additional details related to the implementation procedure will be presented, particularly for the MHD test cases, since additional considerations have to be taken into account.

4.1   Implementation details

4.1.1   The `deal.II` libraries

`deal.II` is a set of libraries programmed in `C++` (this work uses the `deal.II` 8.4 version introduced in [2]), which aids the implementation of grid based numerical methods. Its structure is based on object oriented programming, it is specialized in Cartesian grids and supports the necessary tools to implement FEM, FVM and DG methods.

The major advantages that it provides with respect to other open-source codes reside on the continuous development of the code and the community working around it. Therefore it has an up-to-date and very complete docu-

mentation, plus a very active users group, where questions are posted almost every day. Nevertheless, it is very important to have a good knowledge of object oriented programming in order to use the libraries.

There are also some withdraws that come from using open source codes. For example, when changes are made in commands from the core libraries, it is sometimes necessary to modify the application code in order to be compatible again. It happens more often with commands from the features being under development.

Another example is that, users are limited to the availability to certain developments in the libraries, and new functionalities that may be very important for certain users are either under development, or not even in the plans of the libraries. So, if the knowledge of the user in high-level programming is good enough, then the user would be able to implement this things in the core code, otherwise the user can post the limitation in the forum and wait until it is implemented by somebody else.

In this matter, just to cite one example, it was until very late in the development of this work that it was possible to implement periodic boundary conditions in combination with a parallel grid.

4.1.2 Characteristics of the `dflo` code

This is a `C++` code developed to compute solutions for systems of conservation laws, which uses the `deal.II` libraries previously introduced. It was initially written by Prof. Dr. Praveen Chandrashekar, and afterwards some users of the code have contribute to expand the functionalities of it, including the author of this work. All of the numerical results presented in this chapter were obtained from simulations implemented in the `dflo` code. As in the case of `deal.II,` it is an open source code, so it is available to public use.

The code has the following characteristics

- works on a Cartesian grid for two dimensional problems,

- uses explicit time integration, a second or third order accuracy can be selected,

- $\mathbb{P}^k$ and $\mathbb{Q}^k$ Legendre polynomial basis can be used,

- can use TVD limiters TVB (average gradient limiter in case of $\mathbb{Q}^k$ polynomial basis),

- the limiting can be applied either over the conserved or the characteristic variables,

- can use the KXRCF shock indicator based on super-convergence, introduced in [24] and [46],

- for the MHD equations, either a local Lax-Friedrichs (LXF) or an entropy stable (ES) numerical flux described in Appendix A.4 can be used.

The `dflo` code is available for public use in the following repository:

$$\texttt{https://github.com/juanpablogallego/dflo}$$

4.1.3   General structure of the `dflo` code

Require: *input.file* and *grid.file*

1: *parameters* $\Leftarrow$ *read.parameters*[†]

2: *grid* $\Leftarrow$ *grid.file*

3: *initial.condition* $\Leftarrow$ *test.case*[†]

4: **if** *periodic.boundary* = true **then**

5: │   create periodic boundary map for all subdomains[‡]

6: **end if**

7: configure FE space[1]

8: **for** all cells **do**

9: │   *current.solution* $\Leftarrow$ *initial.condition*

10: **end for**

11: **if** *shock.indicator* = true **then**

12: │   **for** all cells **do**

13: │   │   compute *shock.indicator*[†]

14: │   **end for**

15: **end if**

16: **if** *apply.limiter* = true **then**

17: │   **for** all flagged cells **do**

18: │   │   apply limiting procedure[†]

19: │   **end for**

20: **end if**

21: *old.solution* $\Leftarrow$ *current.solution*

22: **while** *time* < *final.time* **do**

23: │   compute $\Delta time$[2]

24: │   **while** *stage* $\leq$ *number.of.stages* **do**[†] [3]

25: │   │   **for** all cells **do**[†] [4]

26: │   │   │   *rhs* $\Leftarrow$ (*volume.integrals*, *interface.integrals*)

27: │   │   │   *stage.update* $\Leftarrow$ *rhs* * *mass.matix*

28: │   │   **end for**

29: │   │   *current.solution* $\Leftarrow$ *current.solution* + *stage.update*

30: │   │   **if** *shock.indicator* = true **then**

31: │   │   │   **for** all cells **do**

32: │   │   │   │   compute *shock.indicator*[5]

33: │   │   │   **end for**

34: │   │   **end if**

35: │   │   **if** *apply.limiter* = true **then**

[†]This has been modified by the author for a parallel implementation in his visit to the TIFR (Bangalore, India) in 2014. The initial state of the code was a sequential implementation of the Euler equations. The parallel code is in the folder `src_mpi` of the repository.

[‡]This has been implemented by the author together with the MHD version of the code which includes the MHD model, the eigensystem, the correction of the $\nabla \cdot \mathbf{B} = 0$ condition, the numerical fluxes used and the set up of the test cases simulated later in this chapter. The MHD version of the code can be found in the `src_mhd` folder of the repository. The test cases implemented can be found in the `examples` folder of the repository.

[1]The Finite Element (FE) space refers to the degrees of freedom per cell, using either the $\mathbb{Q}^k$ or the $\mathbb{P}^k$ Legendre polynomial basis.

[2]The timestep calculation considers the CFL condition for the DG scheme.

[3]Here the explicit Runge-Kutta time integration mehtod is used.

[4]Here the Discontinuous Galerkin scheme integration is used.

[5] When the shock indicator and the limiting procedure are activated, they are run at every stage of the expicit Runge-Kutta method.

```
36:  │  │  │      for all flagged cells do
37:  │  │  │      │  apply limiting procedure⁵
38:  │  │  │      end for
39:  │  │  end if
40:  │  end while
41:  │  time ⟸ time + Δtime
42:  │  old.solution ⟸ current.solution
43:  │  if output results then
44:  │  │  write current.solution† ⁶
45:  │  end if
46:  end while
47:  write current.solution
48:  if compute error then
49:  │  compute the L²-error‡
50:  end if
```

4.2 Compressible inviscid Euler equations - Test Cases

A set of well known test cases were selected to show the performance of the code implementation. The main interest was focused on the use of the KXRCF shock indicator, which was contrasted to the indicator due to the limiting strategy.

It is important to highlight that all the results shown for the both system of equations were simulated using the $\mathbb{Q}^2$ Legendre polynomial basis, and therefore, also the $\mathbb{Q}^k$ polynomial basis limiter was used with $M = 0$. In fact, no significant difference were noted when a different limiter strategy was used.

This decision was made since for one dimensional test cases all limiting procedures have a good performance for different test cases, plus all of them depend on parameters that have to be adjusted. There is none outstanding difference from one strategy to another.

A second important aspect is the selection of the numerical flux which was used in these test cases. The results presented here were only those computed using the Local Lax-Friedrichs flux, which is well know for being one of the most simple stable numerical fluxes for being implemented.

Other kind of numerical fluxes like the HLLC or a Roe type Riemann solver could have been used, which would have added less numerical dissipation to the solution. The use of a different numerical flux would have required a different setup of the parameters in the limiting procedure.

Each test case has different grid-size, so they are as similar as possible to the common test cases in the literature.

4.2.1 Forward facing step

It is a classical two-dimensional test case introduced for the first time by Emery in [23], and later shown by Woodward and Corela in [80]. A fluid moving at three Mach enters a wind tunnel form the left. The tunnel has a reduction of its height due to a step in the lower wall which opposes the direction of the fluid.

---

[6]The output of results can be configured using either timesteps or number of iterations.
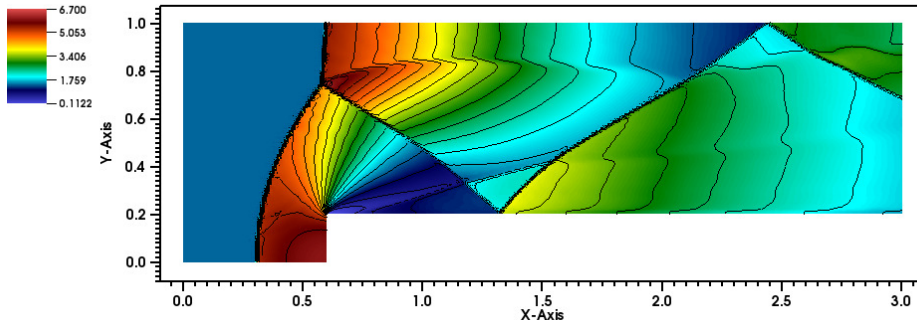
Figure 4.1: Forward facing step test case with no shock indicator after $t = 4$ seconds, for grid-size $\Delta x_1 = \Delta x_2 = \frac{1}{100}$, polynomial order $k = 2$, and a Lax-Friedrichs numerical flux.
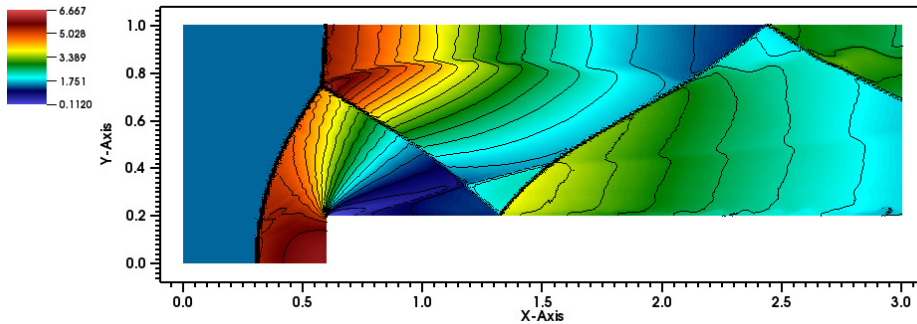

Figure 4.2: Forward facing step test case with the energy as shock indicator after $t = 4$ seconds, for grid-size $\Delta x_1 = \Delta x_2 = \frac{1}{100}$, polynomial order $k = 2$, and a Lax-Friedrichs numerical flux.

A shock-wave is then generated being reflected by the upper and lower walls of the tunnel.

For the initial state the gas constant is $\gamma = 1, 4$, the density and the energy are also constant over the whole domain with values $\rho = 1.4$ and $E = 8.8$ respectively. Finally, the speed of the gas is only horizontal with values $\mathbf{u} = [u_1, u_2]^T = [3, 0]^T$.

Figures 4.1, 4.2 and 4.4 show the results at time $t = 4.0$ of the simulation without shock indicator, and using the energy and the density as measure variables for the KXRCF shock indicator respectively. Figures 4.3 and 4.5 show the cells flagged as "trouble using the energy and the density in the shock indicator. Contour plots are also shown in order to highlight the location for the discontinuities in the solution. It can be noticed that the indicator is able to identify shock regions reliably. Moreover it does not flag the contact discontinuity in the density, which means that the limiter is not applied there.

Finally, Figure 4.6 shows a comparison of the contour curves from the three cases cited before together with the results shown in [17]. The numerical results have a bigger grid element size than the reference solution (Figure 4.6 plot (d)), however the results are comparable. The good behavior around the reflection points at the bottom wall are due to a refinement patch around the corner of the step. This is also explained in [17], where it is shown how a refinement in
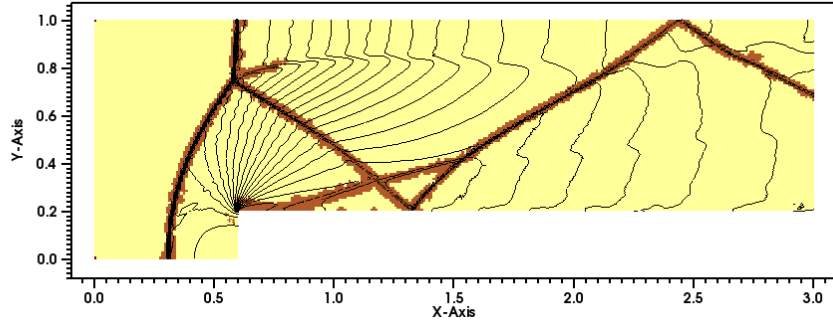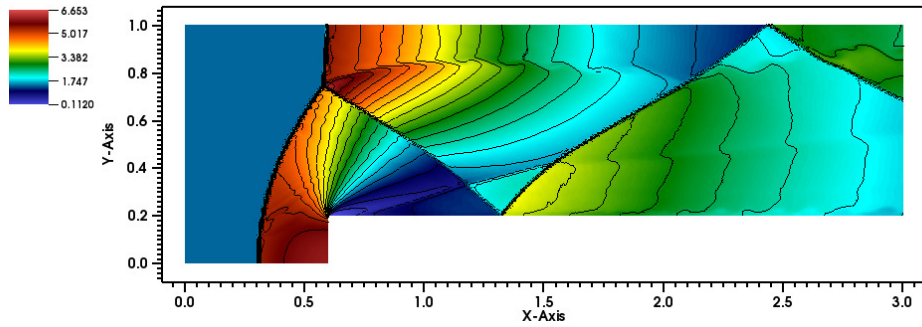
Figure 4.3: Energy shock indicator for the forward facing step test case after $t = 4$ seconds, for grid-size $\Delta x_1 = \Delta x_2 = \frac{1}{100}$, polynomial order $k = 2$, and a Lax-Friedrichs numerical flux.
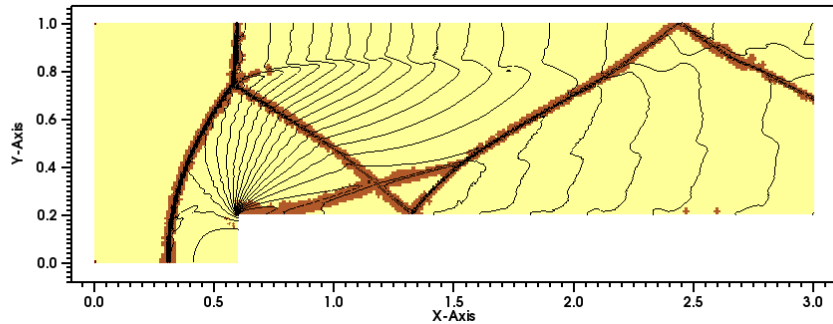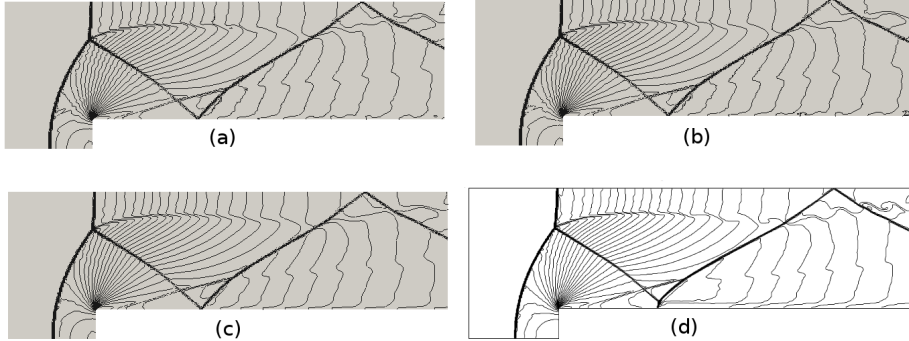

Figure 4.4: Forward facing step test case with the density as shock indicator after $t = 4$ seconds, for grid-size $\Delta x_1 = \Delta x_2 = \frac{1}{100}$, polynomial order $k = 2$, and a Lax-Friedrichs numerical flux.

Figure 4.5: Density shock indicator for the forward facing step test case after $t = 4$ seconds, for grid-size $\Delta x_1 = \Delta x_2 = \frac{1}{100}$, polynomial order $k = 2$, and a Lax-Friedrichs numerical flux.

Figure 4.6: Comparison of the contour plots for the forward facing step simulated with the density (a) and the energy (b) as indicator variables, without indicator (c) and contrasted with the results presented in [17] (d).
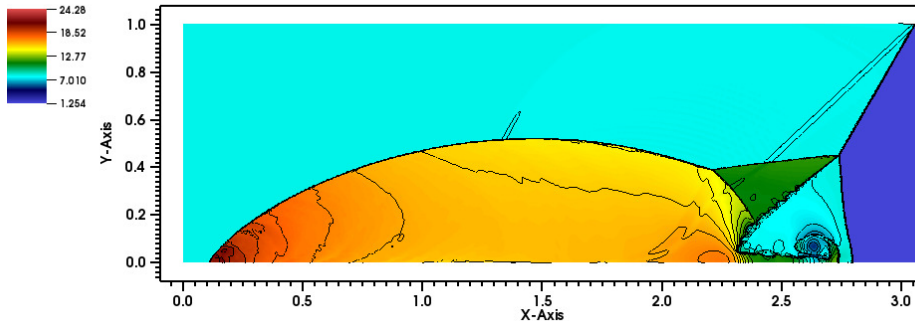

Figure 4.7: Double mach reflection test case with a grid-size of $\Delta x_1 = \Delta x_2 = \frac{1}{360}$, $\mathbb{Q}^2$ Legendre polynomial basis and without shock indicator.

the step corner improves the solution around the reflexion point at the bottom.

4.2.2 Double mach reflection

This test problem was presented by Woodward and Colella in [80] and consists of a planar shock wave at 10-Mach in a gas with $\gamma = 1, 4$ which meets a reflecting wall in an angle of 60 degrees. This test case was simulated, both without shock indicator (Figure 4.7) and also using the KXRCF shock indicator using the density (Figure 4.10) and the energy (Figure 4.8) as measure variables.

For the initial set up, at the pre-shock region the values of the density and the pressure are $\rho = 1.4$ and $p = 1$, and both components of the velocity are zero. At the post-shock region, the values of the density and the pressure are $\rho = 8$ and $p = 116.5$ respectively, and the speed has a magnitude of $|\mathbf{u}| = 8.25$ forming an angle of 30 degrees with the horizontal, then the component values are $\mathbf{u} = [7.145, -4.125]^T$.

The KXRCF shock indicator is able to identify regions where the solution is discontinuous so the limiter is only applied in those regions, as it is shown in Figure 4.11 using the density as indicator variable and 4.9 using the energy as
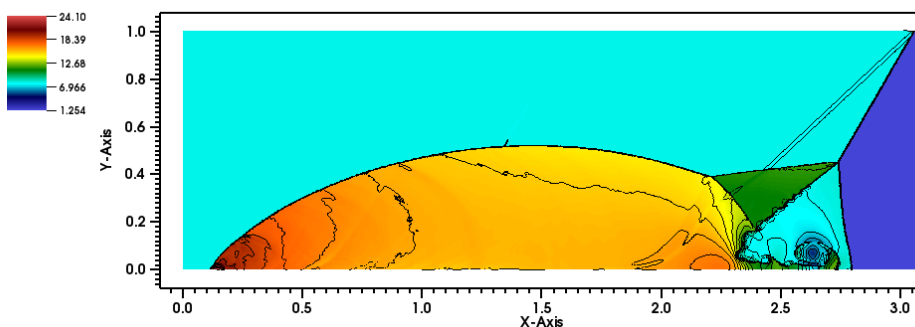
Figure 4.8: Double mach reflection test case with a grid-size of $\Delta x_1 = \Delta x_2 = \frac{1}{360}$, $\mathbb{Q}^2$ Legendre polynomial basis and with the energy as variable for the KXRCF shock indicator.
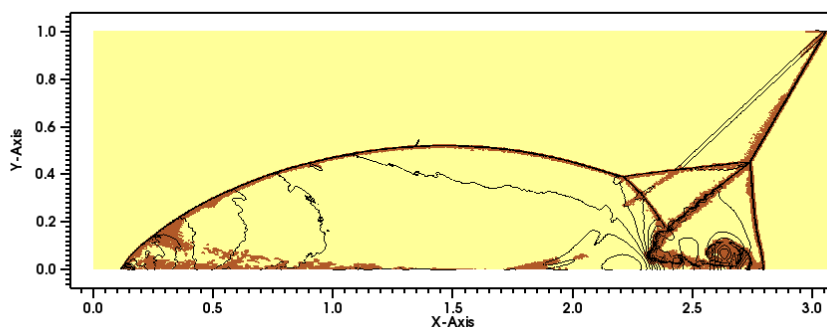

Figure 4.9: Flagged trouble cells using the KXRCF shock indicator with the energy as indicator variable.

indicator variable.

In Figure 4.12 a zoom around the double Mach reflection is shown for different three cases. Figure 4.13 shows only the contour plots to contrast the numerical results with one of the results in [17].

It should be noted that the numerical results presented in this work have less resolution than the reference case (showed in the lower right corner of Figure 4.13). Nevertheless, the results with and without shock indicator compare favorably with the reference case.

4.2.3   Sedov blast wave

This test case is described in [84], and consists of the following setup: a gas at rest (initial velocity is zero and $\gamma = 1.4$) has a uniform density distribution $\rho = 1$ and a highly concentrated energy at the center of the domain, which is given by

$$E = \begin{cases} \frac{0.979264}{|\tau_c|} & \tau_c \subset D_h, \\ 10^{-12} & \text{elsewhere,} \end{cases}$$

where τ_c is the cell at the center of the domain. Note that the initial energy is concentrated in the center cell to emulate a delta singularity, The concentration of energy generates a shock wave that propagates radially with time. The initial
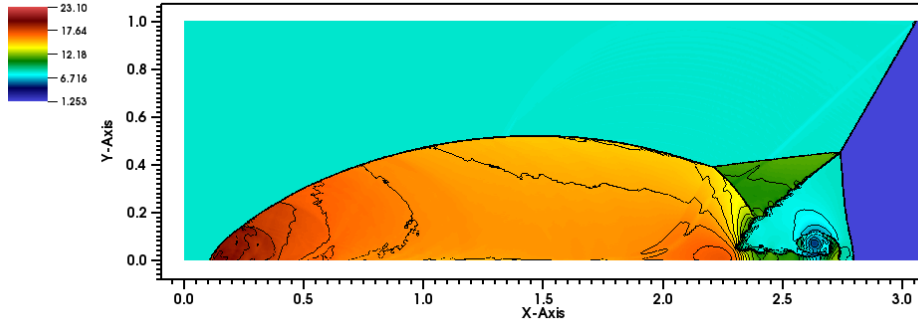
Figure 4.10: Double mach reflection test case with a grid-size of $\Delta x_1 = \Delta x_2 = \frac{1}{360}$, $\mathbb{Q}^2$ Legendre polynomial basis and with the density as variable for the KXRCF shock indicator.
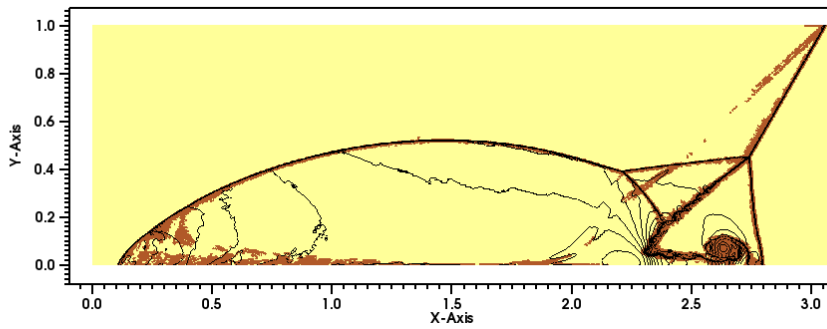

Figure 4.11: Flagged trouble cells using the KXRCF shock indicator with the energy as indicator variable.

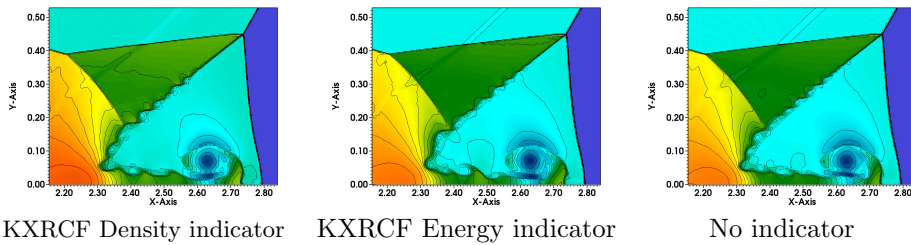KXRCF Density indicator   KXRCF Energy indicator   No indicator

Figure 4.12: Double Mach reflection zoomed comparison, polynomial order $k = 2$, $\Delta x_1 = \frac{1}{360}$.
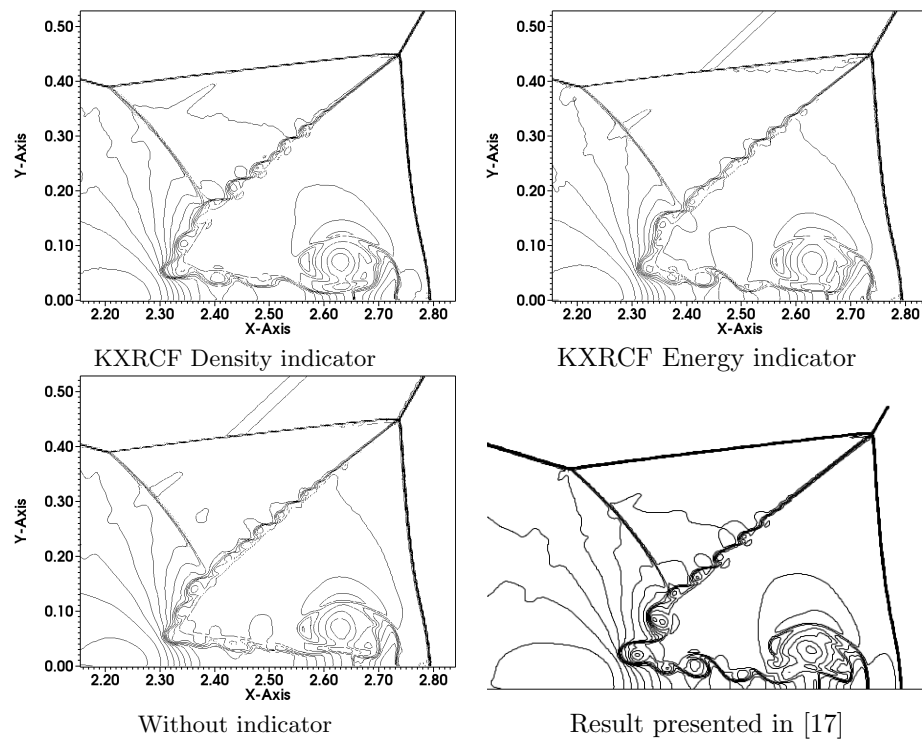
Figure 4.13: Double Mach reflection zoomed comparison, polynomial order $k = 2$, $\Delta x_1 = \frac{1}{360}$. For the case presented in [17] the grid size is $\Delta x_1 = \frac{1}{480}$
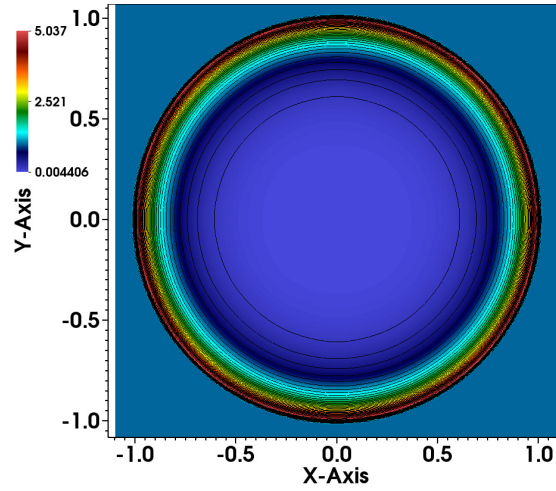
Figure 4.14: Sedov blast wave test case with a grid-size of $\Delta x_1 = \Delta x_2 = \frac{1}{320}$, $\mathbb{Q}^2$ Legendre polynomial basis and without the shock indicator.

energy depends on the size of the center cell $|\tau_c|$ in order to ensure the same shock speed is produced at different resolutions of the grid.

Because of its nearly-vacuum conditions, this test case is used to test whether the scheme preserves the positivity of state variables, which is usually not the case due to the spurious oscillations around the discontinuity when a high order method is used. In this particular case, the use of the positivity preserving limiter presented in Section 3.3 is needed, since even after applying a slope limiter strategy it may happen that some values of the solution at the quadrature points are negative.

Figures 4.14 and 4.15 show the profiles of the density after 1 second of simulation, without shock indicator and using the energy as measure variable for the KXRCF shock indicator respectively. The cells flagged as trouble are shown in Figure 4.16, when the energy is used as indicator variable. Contour plots are also added to highlight the location of the shock in the solution.

Due to the uniform distribution of the density in the initial condition, the KXRCF shock indicator fails at the shock wave detection when the density is used as input variable. As a result the code crashes after a single time step, and therefore in this particular case only the energy was used as indicator.

4.3 Ideal magnetohydrodynamical (MHD) model - Test cases

Since the final goal of this work is to consider a discontinuous Galerkin scheme to solve ideal MHD equations introduced in Section 1.2.2, some test cases were selected in order to check the behavior of both the scheme and the post-processing tools used to control spurious oscillations around discontinuous solutions.

Even though only two-dimensional test cases will be presented, the numerical solution of this system requires the use of the three components of both the momentum and the magnetic field equations, which was not the case for the
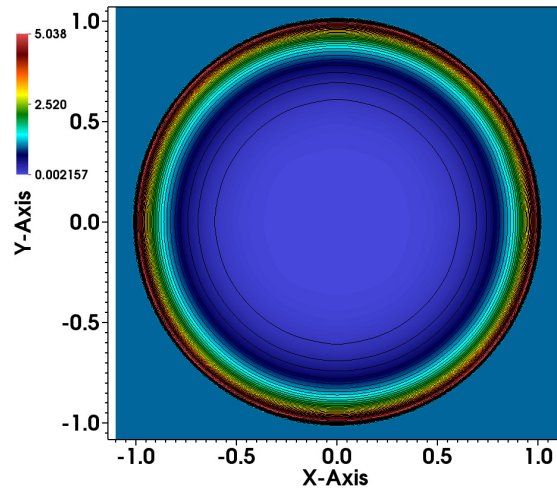
Figure 4.15: Sedov blast wave test case with a grid-size of $\Delta x_1 = \Delta x_2 = \frac{1}{320}$, $\mathbb{Q}^2$ Legendre polynomial basis and with the energy as variable for the KXRCF shock indicator.
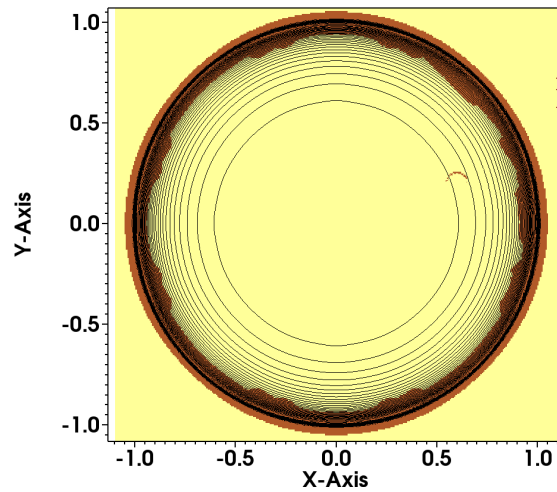

Figure 4.16: Flagged trouble cells for the Sedov blast wave test case with the energy as indicator variable for the KXRCF shock indicator.

Euler equations.

The main difficulty that has to be taken into account, and which is not considered in the previous PDE system, is the divergence free condition that has to be satisfied by the magnetic field ($\nabla \cdot \mathbf{B} = 0$). In general, numerical schemes will not preserve this property of the system by themselves, therefore a strategy needs to be selected in order to enforce this requirement.

4.3.1 Divergence free condition $\nabla \cdot \mathbf{B} = 0$

In order to fulfill this condition, this work follows the methodology introduced by Powell in [62], were the source terms $\varphi'(\mathbf{q}) \nabla \cdot \mathbf{B}$ are being added to the conservation form of the MHD equations as follows

$$\partial_t \mathbf{q}_h + \nabla \cdot \mathbf{F}(\mathbf{q}_h) + \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B} = 0, \tag{4.1}$$

where $\mathbf{F} = [\mathbf{f}_1, \mathbf{f}_2]$ in two dimensions and $\varphi'(\mathbf{q}) = [0, \mathbf{B}, \mathbf{u}, \mathbf{u} \cdot \mathbf{B}]^T$, $\mathbf{u} = [u_1, u_2, u_3]$ is the vector of velocities in all space dimensions. This procedure was applied even before by Godunov (see [4]) as he devised a method to symmetrize the MHD equations (see Appendix A.2 for details in the Godunov symmetrization procedure).

The procedure, as it was described by Powell, looks for a way to modify the MHD system such that the Jacobian matrices of the quasilinear form are no-singular, the new corresponding eigenvectors "make sense" physically and the seven other eigenvalues and eigenvectors are not modified.

After this procedure is applied, the zero eigenvalue of the MHD system is replaced by an advection wave $\lambda = u_\kappa$ in the κth-magnetic component of each $\mathbf{A}_\kappa$ Jacobian matrix. Therefore, the system becomes hyperbolic and the Jacobian matrices are no longer singular.

The resulting system is non-conservative, but the additional terms are multiples of $\nabla \cdot \mathbf{B}$. Taking a closer look to the modified system 4.1, the conservation of magnetic field is given by

$$\partial_t \mathbf{B} + \nabla \cdot (\mathbf{u}\mathbf{B} - \mathbf{B}\mathbf{u}) + \mathbf{u}\nabla \cdot \mathbf{B} = 0.$$

Now, defining a new variable $\mathcal{D} = \nabla \cdot \mathbf{B}$ and taking the divergence of the modified conservation of magnetic field, the following expression is obtained

$$\partial_t \mathcal{D} + \nabla \cdot (\mathbf{u}\mathcal{D}) = 0,$$

which, in combination with the conservation of mass, produces the following result

$$\partial_t \left(\frac{\mathcal{D}}{\rho} \right) + \mathbf{u} \cdot \nabla \left(\frac{\mathcal{D}}{\rho} \right) = 0.$$

This means that the variable $\frac{\mathcal{D}}{\rho}$ is constant along particle paths, and therefore if the initial value of $\mathcal{D} = \nabla \cdot \mathbf{B}$ is zero, then it remains zero for later times. Numerically it is expected that, at least for smooth solutions, the term $\nabla \cdot \mathbf{B}$ should remain close to zero.

A review of methods used to deal with the divergence free condition of the magnetic field are shown by Tóth in [78], including The method by Powell just presented, the Constrained Transport (CT) method and projection schemes to eliminate the non physical component of the magnetic field.

A different approach is proposed by Fengyan Li in [53], where instead of using a single set of polynomial basis ϕ_j to approximate all the state variables, the magnetic field components are approximated using a special kind of vectorial basis function $\hat{\phi}$, which satisfies the condition $\nabla \cdot \hat{\phi} = 0$. Therefore, any approximation of the magnetic field as a linear combination of these basis functions will automatically satisfy the condition $\nabla \cdot \mathbf{B}_h = 0$.

4.3.2 The DG scheme for MHD

By applying the procedure by Powell, some modifications have to be done to the DG scheme presented in Equation (2.14). Due to the source term added to the equation, the following integral has to be added to the weak formulation

$$\int_{\tau_\ell} v_h \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B}_h d\mathbf{x} = -\int_{\tau_\ell} (\mathbf{B}_h \cdot \nabla)(v_h \varphi'(\mathbf{q}_h)) d\mathbf{x} + \sum_{\alpha \subseteq \partial\tau_\ell} \int_\alpha v_h^- \varphi'(\mathbf{q}_h^-) \mathbf{B}_h^* \cdot \mathbf{n} d\sigma$$

where the value $\mathbf{B}_h^*$ is replaced by a numerical flux. Integrating by parts a second time it follows

$$\int_{\tau_\ell} v_h \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B}_h d\mathbf{x} = \int_{\tau_\ell} v_h \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B}_h d\mathbf{x} + \sum_{\alpha \subseteq \partial\tau_\ell} \int_\alpha v_h^- \varphi'(\mathbf{q}_h^-)(\mathbf{B}_h^* - \mathbf{B}_h^-) \cdot \mathbf{n} d\sigma$$

Replacing the numerical flux in boundary integral by a central flux $\mathbf{B}_h^* = \frac{\mathbf{B}_h^- + \mathbf{B}_h^+}{2}$, a discrete version of the divergence of the magnetic field will appear at the boundary fluxes as

$$\int_{\tau_\ell} v_h \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B}_h d\mathbf{x} = \int_{\tau_\ell} v_h \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B}_h d\mathbf{x} + \frac{1}{2} \sum_{\alpha \subseteq \partial\tau_\ell} \int_\alpha v_h^- \varphi'(\mathbf{q}_h^-)(\mathbf{B}_h^+ - \mathbf{B}_h^-) \cdot \mathbf{n} d\sigma.$$

Then, the semi-discrete scheme for the MHD system at each cell τ_ℓ of the domain will have the following structure

$$\int_{\tau_\ell} \left[(\partial_t \mathbf{q}_h) v_h - \nabla v_h \cdot (\mathbf{f}_1(\mathbf{q}_h) + \mathbf{f}_2(\mathbf{q}_h)) + v_h \varphi'(\mathbf{q}_h) \nabla \cdot \mathbf{B}_h \right] d\mathbf{x}$$

$$+ \int_{\partial\tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) + \frac{1}{2} \varphi'(\mathbf{q}_h^-)(\mathbf{B}_h^+ - \mathbf{B}_h^-) \cdot \mathbf{n} \right] v_h^- d\sigma = 0, \quad (4.2)$$

where $\mathbf{n}$ is the outward normal vector at each interface α of the element τ_ℓ and $\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n})$ is replaced by a selected numerical flux. Due to these additional integrals the numerical scheme becomes non-conservative.

4.3.3 Numerical fluxes for the MHD system

Two different numerical fluxes were implemented and used for the simulations of the test cases presented later in this chapter.

Local Lax-Friedrichs-type (LXF) numerical flux

The first one is a local Lax-Friedrichs-type (LXF) numerical flux of the form.

$$\hat{h}(\mathbf{q}_h^+, \mathbf{q}_h-) = \frac{1}{2}(\mathbf{F}(\mathbf{q}_h^+) + \mathbf{F}(\mathbf{q}_h^-)) - \frac{\lambda_{max,\partial\tau_\ell}}{2}(\mathbf{q}_h^+ - \mathbf{q}_h^-) \qquad (4.3)$$

where $\lambda_{max,\partial\tau_\ell}$ is the maximum eigenvalue of the Jacobian matrix of system at the cell interface $\partial\tau_\ell$. As it was shown in Section 1.2.2, the maximum eigenvalue for the MHD system is given by $\lambda = \|\mathbf{u}\| + c_f$, where $\|\mathbf{u}\|$ is the Euclidean norm in $\mathbb{R}^3$ of the local velocity and c_f is the fast speed given by

$$c_f^2 = \frac{1}{2}\left(a^2 + b^2\right) + \frac{1}{2}\sqrt{\left(a^2 + b^2\right)^2 - 4a^2(b\cdot\mathbf{n})},$$

where $\mathbf{n}$ is the vector normal to the interface where the flux is computed. The maximum value of c_f happens when the term $4a^2(b\cdot\mathbf{n}) = 0$, in which case

$$c_{f,max} = \sqrt{a^2 + b^2},$$

where

$$a^2 = \frac{\gamma p}{\rho} \quad \text{and} \quad b^2 = \frac{|\mathbf{B}_h|^2}{\rho}.$$

Then the value of $\lambda_{max,\partial\tau_\ell}$, used in the flux function describe in (4.3), is given by the maximum of the following expression

$$\lambda_{max} = \|\mathbf{u}_h\| + \sqrt{\frac{\gamma p + |\mathbf{B}_h|^2}{\rho}}$$

evaluated on the quadrature points of the interface $\partial\tau_\ell$.

Entropy Stable (ES) numerical flux [7]

The details for the design of the ES numerical flux are shown in Appendix A.4. It satisfies the conditions needed for the entropy stability proof presented in Appendix A.3. This flux derivation follows a similar procedure as the one introduced in [9], which was designed for a FV entropy stable scheme.

Even though the entropy stability was proven for the semi-discrete DG scheme, this work will test the behavior of the ES numerical flux for the explicit RKDG numerical scheme. The general form of the numerical flux is given by

$$\mathcal{H}(\mathbf{q}_h^-,\mathbf{q}_h^+,\mathbf{n}) = \mathcal{H}^*(\mathbf{q}_h^-,\mathbf{q}_h^+,\mathbf{n}) - \frac{1}{2}\mathbf{D}(\mathbf{q}_h^-,\mathbf{q}_h^+)(\mathbf{w}^+ - \mathbf{w}^-), \quad \mathbf{D} = \mathbf{D}^T \geq 0 \quad (4.4)$$

---

[7] The author of this work thanks Praveen Chandrashekar for private communications and the calculations for the development of this flux.

where $\mathcal{H}^*(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n})$ components are defined as

$$\mathcal{H}_1^* = \hat{\rho}\overline{u}_n,$$

$$\mathcal{H}_2^* = P^* n_1 + \overline{u}_1 \mathcal{H}_1^* - \overline{B}_n \overline{B}_1, \qquad P^* = \frac{\overline{\rho}}{2\overline{\beta}} + \frac{1}{2}|\mathbf{B}_h|^2,$$

$$\mathcal{H}_3^* = P^* n_2 + \overline{u}_1 \mathcal{H}_1^* - \overline{B}_n \overline{B}_2,$$

$$\mathcal{H}_4^* = P^* n_3 + \overline{u}_1 \mathcal{H}_1^* - \overline{B}_n \overline{B}_3,$$

$$\mathcal{H}_5^* = \frac{1}{\overline{\beta}}\left(\overline{\beta u_n B_1} - \overline{\beta u_1 B_n}\right),$$

$$\mathcal{H}_6^* = \frac{1}{\overline{\beta}}\left(\overline{\beta u_n B_2} - \overline{\beta u_2 B_n}\right),$$

$$\mathcal{H}_7^* = \frac{1}{\overline{\beta}}\left(\overline{\beta u_n B_3} - \overline{\beta u_3 B_n}\right),$$

$$\mathcal{H}_8^* = \frac{1}{2}\left[\frac{1}{(\gamma-1)\hat{\beta}} - \overline{|\mathbf{u}_h|^2}\right]\mathcal{H}_1^* + \overline{u_1}\mathcal{H}_2^* + \overline{u_2}\mathcal{H}_3^* + \overline{u_3}\mathcal{H}_4^*$$
$$+ \overline{B_1}\mathcal{H}_5^* + \overline{B_2}\mathcal{H}_6^* + \overline{B_3}\mathcal{H}_7^* - \frac{1}{2}\overline{u_n}\overline{|\mathbf{B}_h|^2} + \left(\overline{u_1 B_1} + \overline{u_2 B_2} + \overline{u_3 B_3}\right)\overline{B_n}.$$

Here $u_n = \mathbf{u}_h \cdot \mathbf{n}$ and $B_n = \mathbf{B}_h \cdot \mathbf{n}$ denote the normal component of the vectors $\mathbf{u}_h$ and $\mathbf{B}_h$ to the interface for which the numerical flux is being computed, the values n_i are the components of the normal vector $\mathbf{n}$, the operator $\overline{(\cdot)}$ denotes the arithmetic average between the values at both sides of the interface and the operator $\hat{(\cdot)}$ denotes the logarithmic average denoted respectively by

$$\overline{\eta} = \frac{1}{2}(\eta^- + \eta^+), \qquad \hat{\eta} = \frac{\eta^+ - \eta^-}{\ln(\eta^+) - \ln(\eta^-)}.$$

The dissipation matrix $\mathbf{D}$ in (4.4) is a semi-definite positive matrix similar to the dissipation used in a Roe-type scheme (see Appendix A.4). It is calculated using the scaled right eigenvectors matrix $\tilde{\mathbf{R}}$ of symmetrization procedure to transform from conserved to entropy variables shown in [4, 9], as follows

$$\mathbf{D} = \tilde{\mathbf{R}}\mathbf{\Lambda}\tilde{\mathbf{R}}^T. \tag{4.5}$$

This work will investigate the results using an explicit strong stability preserving (SSP) third order Runge-Kutta (RK) time integration (as the one used in [17] and [25]) when it is applied to an entropy stable scheme designed for the semi-discrete scheme.

4.3.4   Limiting procedure

As in the case of the Euler equations, it is necessary to use a limiting procedure to ensure non-oscillatory approximations for discontinuous solutions. The TVD-type limiters implemented in the code have been also used for the MHD test cases.

In Section 3.2, it was pointed out that for one-dimensional problems, there was an advantage in limiting over the characteristic variables instead of the conserved variables. It is basically a procedure in which the Jacobian matrices are

diagonalized, then the limiting procedure is applied over the characteristic variables of the system and finally the system is transformed back to the conserved variables.

For the test cases considered in the following sections, a similar procedure is followed. Three different eigensystems were found in order to calculate the diagonalization of the MHD system. The first one is the Godunov symmetrization (See Appendix A.2), however in this case the limiting would be applied not on the characteristic variables, but over the entropy variables. The second method is a symmetrization method presented by Jameson in [42], were two sets of eigensystem are proposed, the only difference being a scaling applied so the symmetrizing variables have either dimension of velocities or densities.

The last diagonalization procedure found in the literature, which is the one used in this work, was introduced by Bristo and Wu in [6], and it was also used in [43]. With the other eigensystems, it may happen that the eigenvectors develop singularities in points where the eigenvalues degenerate. On the other hand, the eigensystem introduced in [6] has a proper choice of normalization, which avoids singularities in the eigenvectors and guarantees a complete set of eigenvectors (see Appendix A.5).

Additional to the limiting procedure, the shock indicator introduced in [24] and [46] and explained in Section 3.5 was also used to detect the places where the limiting procedure has to be applied.

4.3.5 Polarized Alfvèn wave

The first MHD test case presented in this work is the Polarized Alfvèn wave, which is described by Tóth in [78], however the setup of the implementation was taken from [26] with a different orientation.

On a rectangular domain $\left[0, \frac{\sqrt{5}}{2}\right] \times \left[0, \sqrt{5}\right]$, a grid of square cells is used with periodic boundary conditions. The Alfvèn wave is fixed to propagate in the direction $\tan^{-1}(\frac{1}{2}) \approx 26.6^o$ with respect to the x_1 axis. The density and the pressure are constants fixed to $\rho = 1$ and $p = 0.1$.

The magnetic and velocity components are easier to define over a rotated system of coordinates, built with the direction of the Alfvèn wave called $x_\parallel$ and its orthogonal direction $x_\perp$. Then the values of the magnetic field are given by $B_\parallel = 1$, $B_\perp = 0.1 \sin(2\pi x_\parallel)$ and $B_3 = 0.1 \cos(x_\parallel)$ and the velocities are given by $u_\parallel = 0.1$, $u_\perp = 0.1 \sin(2\pi x_\parallel)$ and $u_3 = 0.1 \cos(2\pi x_\perp)$. Transforming these values into Cartesian coordinates, the following expressions arise for the velocity components

$$u_1 = u_\parallel \cos(\theta) - u_\perp \sin(\theta),$$
$$u_2 = u_\parallel \sin(\theta) + u_\perp \cos(\theta), \tag{4.6}$$
$$u_3 = 0.1 \cos(\theta_0)$$

and for the magnetic field components

$$B_1 = B_\parallel \cos(\theta) - B_\perp \sin(\theta),$$
$$B_2 = B_\parallel \sin(\theta) + B_\perp \cos(\theta), \tag{4.7}$$
$$B_3 = 0.1 \cos(\theta_0)$$

where $\theta = \tan^{-1}(\frac{1}{2})$ and $\theta_0 = 2\pi(x_1 \cos(\theta) + x_2 \sin(\theta))$. Due to the periodicity of the test case, the solution will return to the initial state every $\Delta t = 1.0$.
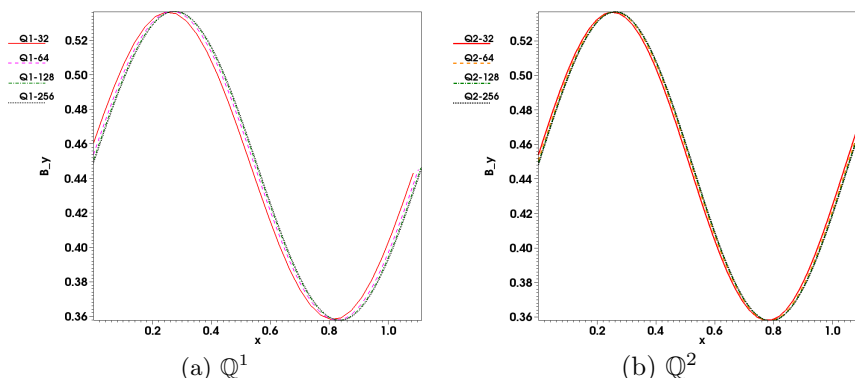
(a) $\mathbb{Q}^1$ (b) $\mathbb{Q}^2$

Figure 4.17: Cross section at $x_2 = 0$ of the second component of the magnetic field B_2 for both $\mathbb{Q}^1$ (a) and $\mathbb{Q}^2$ (b) polynomial basis at $t = 5$, using the LXF flux.

(a) $\mathbb{Q}^1$ (b) $\mathbb{Q}^2$
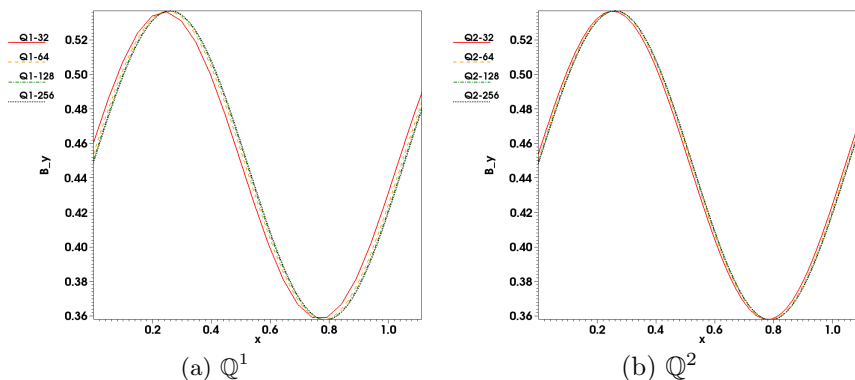
Figure 4.18: Cross section at $x_2 = 0$ of the second component of the magnetic field B_2 for both $\mathbb{Q}^1$ (a) and $\mathbb{Q}^2$ (b) polynomial basis at $t = 5$, using the ES flux.

This test case was simulated with two different numerical fluxes, and four different grid sizes. Since the solution is smooth and periodic, convergence rates where computed in order to study the accuracy of the DG method constructed for the MHD system. It is also worth noting that since the solution does not form discontinuities, the limiters where not activated in this simulations.

Figure 4.17 shows the cross sections of the second component of the magnetic field B_2 at the x_1-axis ($x_2 = 0$) for four different grid-sizes using the LXF numerical flux with both $\mathbb{Q}^1$ (Figure 4.17(a)) and $\mathbb{Q}^2$ (Figure 4.17(b)) polynomial basis.

In the same fashion, Figure 4.18 shows the cross sections of the second component of the magnetic field B_2 at the x_1-axis ($x_2 = 0$) for four different grid-sizes using the ES numerical flux with both $\mathbb{Q}^1$ (Figure 4.18(a)) and $\mathbb{Q}^2$ (Figure 4.18(b)) polynomial basis.

The L^2-error was then calculated for the four different grid-sizes using the LXF flux and the ES flux. The results are shown in Table 4.1 at $t = 5.0$. It can

	n	LXF		ES	
		L^2-error	order	L^2-error	order
$\mathbb{Q}^1$	32	7.83E-04	–	9.30E-04	–
	64	1.60E-04	2.29	1.90E-04	2.29
	128	5.71E-05	2.10	4.47E-05	2.09
	256	7.59E-06	2.03	1.10E-05	2.02
$\mathbb{Q}^2$	32	1.58E-04	–	2.04E-05	–
	64	1.36E-05	3.54	1.49E-06	3.78
	128	9.56-07	3.83	1.56E-07	3.26
	256	6.66E-08	3.84	1.79E-08	3.12

Table 4.1: L^2- errors of the polarized Alfvèn test case at $t = 5.0$ using four different grid-sizes, for both $\mathbb{Q}^1$ and $\mathbb{Q}^2$ polynomial basis, and both the LXF and the ES numerical fluxes.
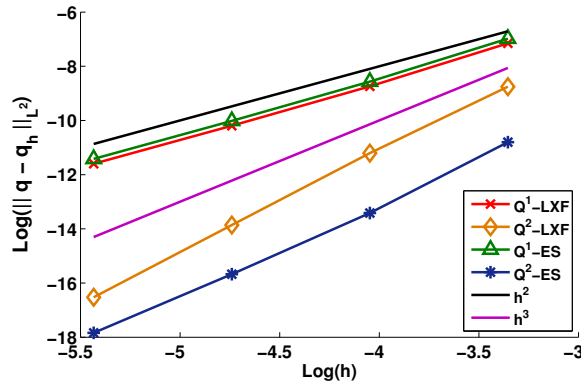

Figure 4.19: Convergence rates of the L^2-errors for the Alfvèn wave test case for both numerical fluxes, the LXF and the ES. The h^2 and h^3 lines are ploted for comparison.

be seen that for the $\mathbb{Q}^1$ polynomial basis the results are very similar for both numerical fluxes, however for the $\mathbb{Q}^2$ polynomial basis the errors produced by the ES numerical flux are considerably smaller than those from the LXF flux for both time levels.

The rate of convergence p can also be calculated using the following formula

$$p = \frac{\log(E_h)}{\log(2)}, \quad \text{with} \quad E_h = \frac{e_h}{e_{\frac{h}{2}}},$$

where e_h represents the L^2-error for a grid-size h. The convergence rates are also shown in Table 4.1 at time $t = 5.0$. It can be noticed that the convergence rates for both $\mathbb{Q}^1$ and $\mathbb{Q}^1$ polynomial basis show the expected value, which are $p = 2$ and $p = 3$ respectively. The results, as suspected from Table 4.1, are very similar for both numerical fluxes.

The L^2-error results are ploted in Figure 4.19 together with h^2 and h^3 lines for comparison.

Finally, table 4.2 shows the L^∞-errors for the $\nabla \cdot \mathbf{B} = 0$ condition after

	n	LXF L^∞-error	LXF order	ES L^∞-error	ES order
$\mathbb{Q}^1$	32	3.76E-02	–	5.12E-02	–
	64	1.88E-02	1.00	2.49E-02	1.04
	128	9.41E-03	1.00	1.23E-02	1.02
	256	4.71E-03	1.00	6.16E-03	1.00
$\mathbb{Q}^2$	32	1.72E-03	–	1.40E-03	–
	64	3.89E-04	2.14	3.95E-04	1.83
	128	1.83E-04	1.09	9.68E-05	2.03
	256	7.88E-05	1.22	2.41E-05	2.01

Table 4.2: L^∞-errors for the $\nabla \cdot \mathbf{B} = 0$ condition of the Polarized Alfvèn wave test case at $t = 5.0$, using both $\mathbb{Q}^1$ and $\mathbb{Q}^2$ polynomial basis, both the LXF and the ES flux, and four different grid-sizes.

crossing five times the domain ($t = 5.0$). It can be seen that for $k = 1$ there are not significant differences between the use of the LXF flux and the ES flux. However, for the $\mathbb{Q}^2$ polynomial basis the L^∞-errors are reduced considerably as it is also showed in Table 4.2, where convergence rates are shown for the $\nabla \cdot \mathbf{B} = 0$ condition. A better behavior is observed when the ES numerical flux is used.

4.3.6   Orszag-Tang vortex

This test case was first introduced by Orszag and Tang in [59], and it has become a reference test case to validate numerical solutions for the MHD system of equations, which is also the motivation of including it in this work.

The set up of the test case is the following, in a square domain of $[0, 1] \times [0, 1]$ with periodic boundary conditions a gas ($\gamma = \frac{5}{3}$) has constant pressure $p = \frac{5}{12\pi}$ and density $\rho = \frac{25}{36\pi}$. The initial speeds are $u_1 = \sin(2\pi x_2)$ and $u_2 = \sin(2\pi x_1)$, and the initial values of the magnetic field are given by $B_1 = -B_0 \sin(2\pi x_2)$ and $B_2 = B_0 \sin(4\pi x_1)$, where $B_0 = \frac{1}{\sqrt{4\pi}}$.

Two different numerical fluxes were used for the simulations of this test case, the LXF flux and the ES numerical flux, previously introduced in this chapter. The $\mathbb{Q}^k$ $k = \{1, 2\}$ Legendre polynomial basis were used, therefore the Powell terms are added into the scheme to satisfy the divergence free condition $\nabla \cdot \mathbf{B}$. As in the case of the Euler equations, the limiting procedure used was the $\mathbb{Q}^k$ polynomial basis limiter.

Finally, the KXRCF shock indicator will be activated only using the energy as indicator variable, since for most of the results of the Euler equations there were no remarkable difference, except for the Sedov blast wave test case, in which the density indicator was not able to detect the discontinuity after only one time step.

The results using the LXF numerical flux are presented first. Figures 4.20 and 4.21 show the values of the density respectively, when no shock indicator and the energy is used as measure variable for the KXRCF indicator. The cell size is fixed to $\Delta x_1 = \Delta x_2 = \frac{1}{256}$.

The cells flagged as trouble, when the energy is used in the KXRCF shock indicator, are displayed in Figure 4.22. The figure includes the contour plots as
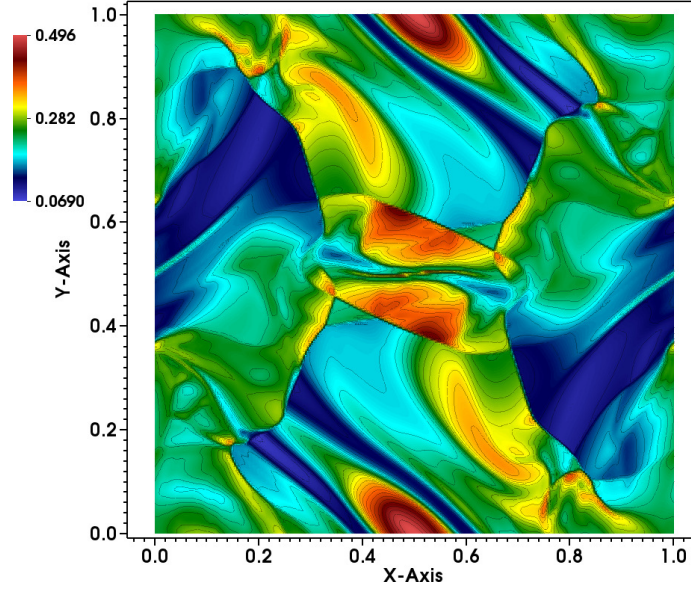
Figure 4.20: Density values of the Orszag-Tang test case at $t = 0.5$ seconds, using the LXF numerical flux, the $\mathbb{Q}^1$ polynomial basis for second order approximation, a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{256}$ and without shock indicator.
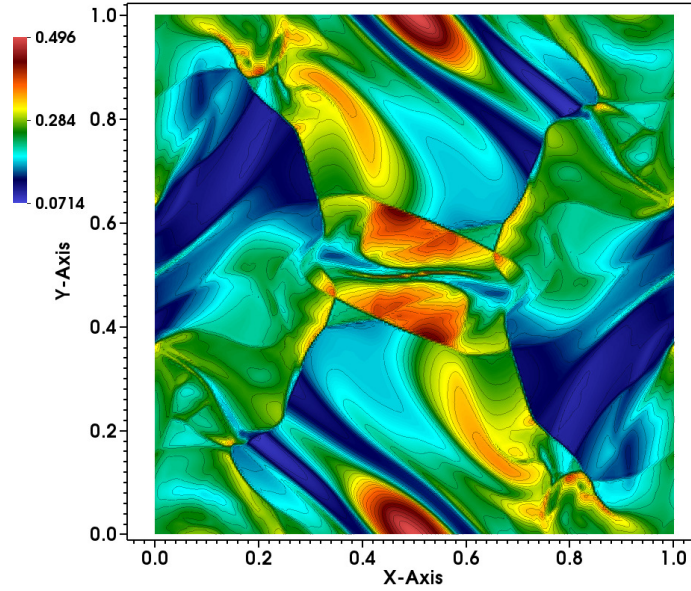

Figure 4.21: Density values of the Orszag-Tang test case at $t = 0.5$ seconds, using the LXF numerical flux, the $\mathbb{Q}^1$ polynomial basis for second order approximation, a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{256}$ and with the energy as variable for the shock indicator.
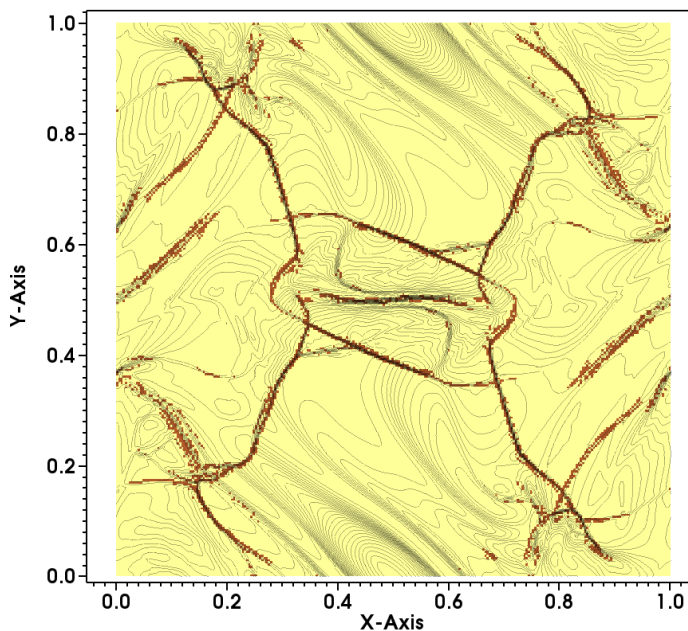
Figure 4.22: Energy indicator for the Orszag-Tang test case using the energy as indicator variable, with a grid size $\Delta x_1 = \Delta x_2 = 1/256$ and the $\mathbb{Q}^1$ polynomial basis for second order approximation.

a guide to locate the shocks in the results. It can be seen that in this case the shock indicator detects the discontinuities accurately.

Figure 4.23 shows a map of colors of the values for the divergence of the magnetic field $\nabla \cdot \mathbf{B}$. These values are near to zero on a great part of the domain except in regions near discontinuities or where different shocks interact.

It is expected that when discontinuous solutions are present, the numerical errors in the divergence condition $\nabla \cdot \mathbf{B} = 0$ are higher than the cases in which no discontinuities are involved. This can be due to the limiting procedure itself, since it was not designed to correct or preserve in any way the $\nabla \cdot \mathbf{B} = 0$ condition.

For a finer grid of $\Delta x_1 = \Delta x_2 = 1/512$, Figures 4.24 and 4.25 show the values of the density when no shock indicator is used and when the Energy is used as indicator variable respectively. The cells flagged as trouble are shown in Figure 4.26.

A comparison of the divergence of the magnetic field for these cases are shown in Figure 4.27. By increasing the resolution it can be seen that the regions with higher magnitude of the divergence values are sharper than in the case with bigger cell-size.

As a way to validate the results obtained so far with the dflo code, a comparison is shown in Figures 4.28 and 4.29. The test case was ran using the $\mathbb{Q}^1$ polynomial basis, three different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/128, 1/256, 1/512\}$) and the LXF numerical flux for the dflo code. The results are contrasted with those generated using the Athena code for a second order approximation with a grid size of $\Delta x_1 = \Delta x_2 = 1/516$ for two slices of the domain, one at $x_2 = 0.3125$ and the second one at $x_2 = 0.4277$.

It can be seen that as the grid size is reduced for the dflo curves, the results

(a) No indicator

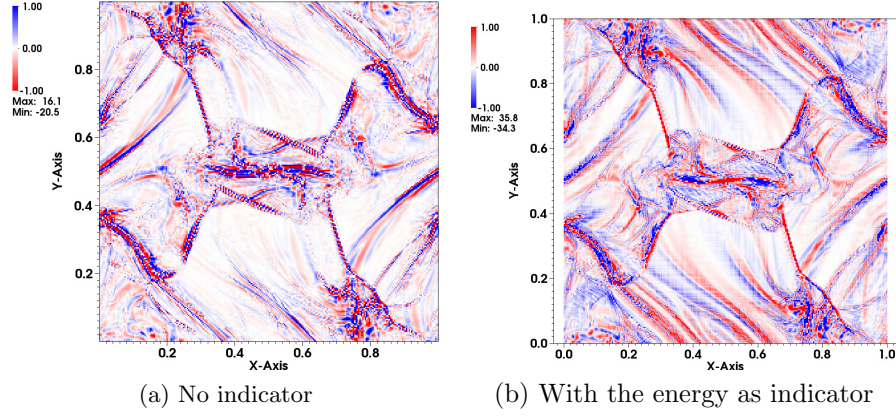(b) With the energy as indicator

Figure 4.23: Values of the $\nabla \cdot \mathbf{B}$ for the Orszag-Tang test case at $t = 0.5$ seconds, using the LXF numerical flux, the $\mathbb{Q}^1$ polynomial basis for second order approximation and a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{256}$.

Figure 4.24: Density values of the Orszag-Tang test case at $t = 0.5$ seconds, using the LXF numerical flux, the $\mathbb{Q}^1$ polynomial basis for second order approximation, a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{512}$ and without shock indicator.
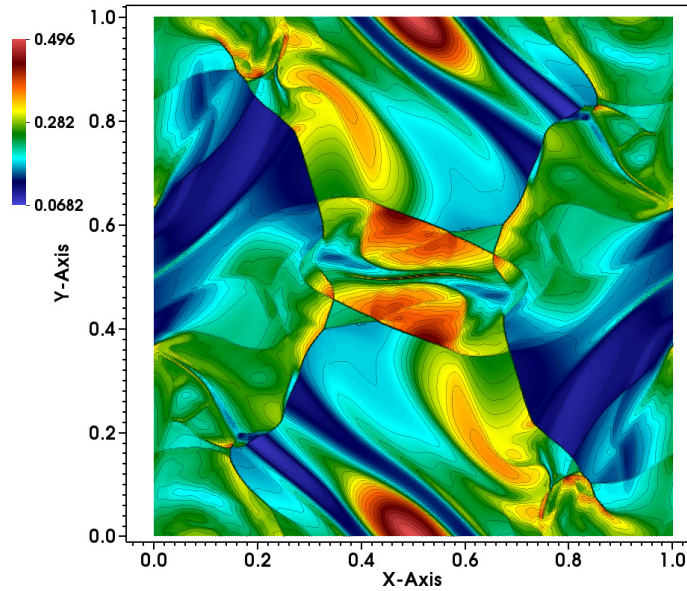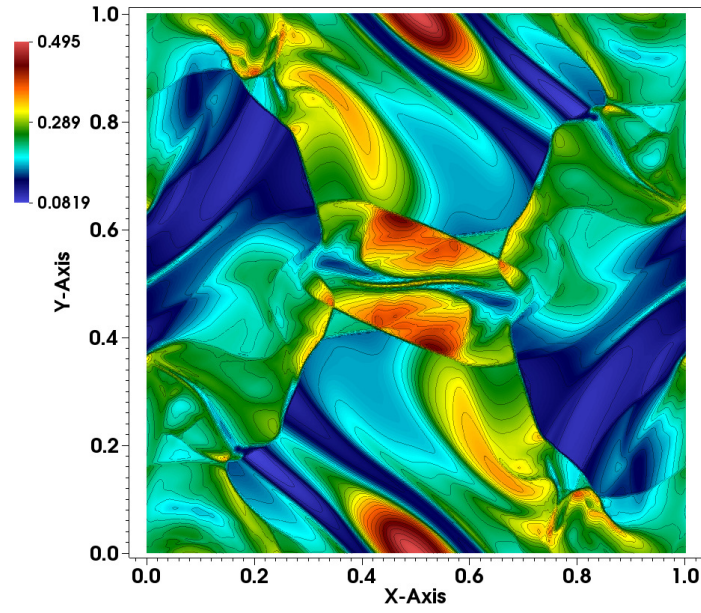
Figure 4.25: Density values of the Orszag-Tang test case at $t = 0.5$ seconds, using the LXF numerical flux, the $\mathbb{Q}^1$ polynomial basis for second order approximation, a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{512}$ and with the energy as indicator variable.
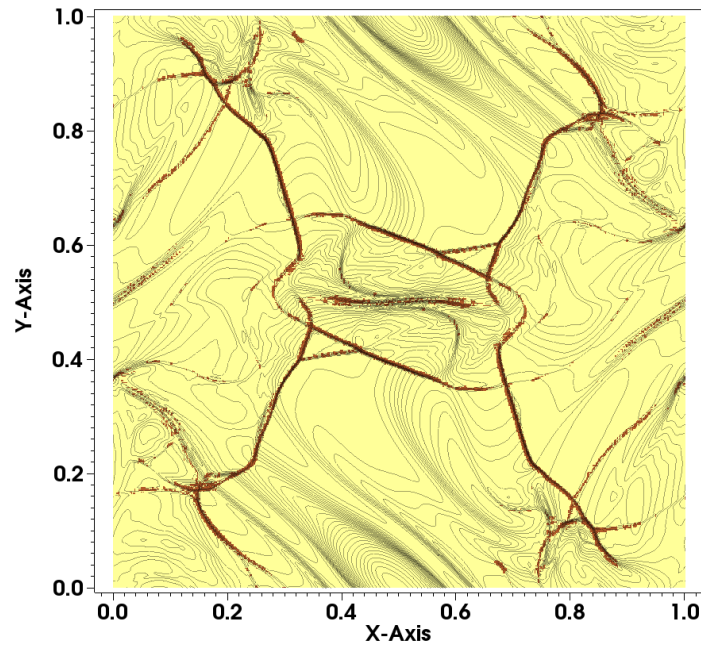

Figure 4.26: Energy indicator for the Orszag-Tang test case present in Figure 4.25 using the energy as indicator functions, with $\Delta x_1 = \Delta x_2 = 1/512$ and the $\mathbb{Q}^1$ polynomial basis for second order approximation.

(a) No indicator

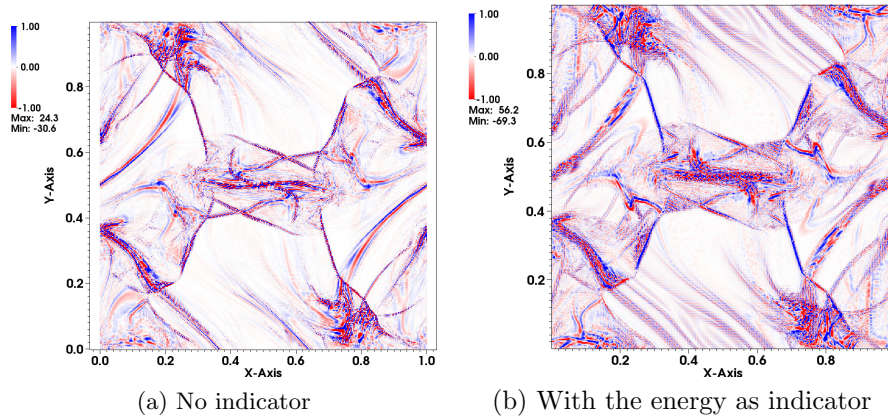(b) With the energy as indicator

Figure 4.27: Values of the $\nabla \cdot \mathbf{B}$ for the Orszag-Tang test case at $t = 0.5$ seconds, using the LXF numerical flux, the $\mathbb{Q}^1$ polynomial basis for second order approximation and a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{512}$.
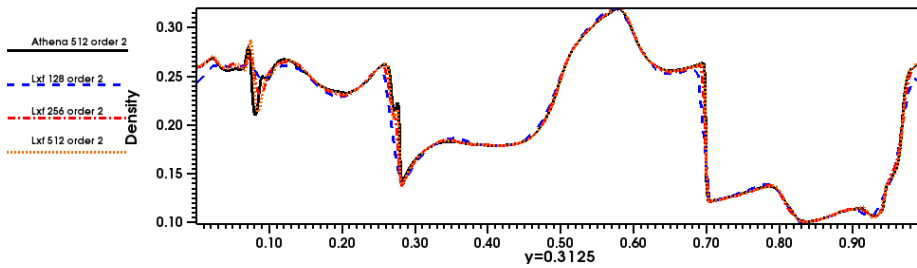

Figure 4.28: Cut comparison ($x_2 = 0.3125$) of the second order test case solution computed using the `dflo` code for three different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/128, 1/256, 1/512\}$), and the results from the Athena code for $\Delta x_1 = \Delta x_2 = 1/512$.

are more similar to those from the Athena code. The `dflo` curves corresponding to the grid size of $\Delta x_1 = \Delta x_2 = 1/512$ agrees with the results from the Athena code.

Running again the Orszag-Tang test case, but this time using the ES numerical flux, the results are shown in Figure 4.30, here the energy was once again used as indicator variable. The cells flagged as troubled are shown in Figure 4.31 together with the contour plots of the density only as a guide.

Once again it can be seen that the KXRCF shock indicator detects correctly the place in which the discontinuities are for in the solution.

Finally, the last example presented here for the Orszag-Tang test case was perform using the $\mathbb{Q}^2$ polynomial basis for a solution with third order of approximation of the solution, a grid with a cell size of $\Delta x_1 = \Delta x_2 = 1/512$, the same limiter procedure used in the previous results and using the energy as indicator variable for the KXRCF shock detector. The results are shown in Figure 4.32.

As it can be seen in Figure 4.33, much more cells are flagged as trouble cells in comparison with the previous cases where the polynomial basis used was the
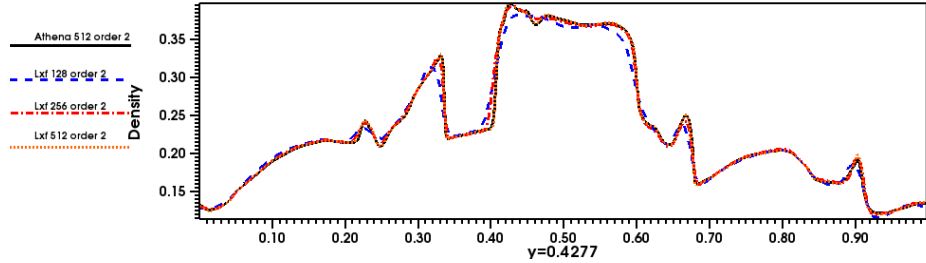
Figure 4.29: Cut comparison ($x_2 = 0.4277$) of the second order test case solution computed using the `dflo` code for three different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/128, 1/256, 1/512\}$), and the results from the Athena code for $\Delta x_1 = \Delta x_2 = 1/512$.

Figure 4.30: Density values of the Orszag-Tang test case at $t = 0.5$ seconds, using the ES numerical flux, $\mathbb{Q}^1$ polynomial basis for second order approximation, using a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{512}$, and with the energy as indicator variable

Figure 4.31: Energy indicator for the Orszag-Tang test case present in Figure 4.30 using the energy as indicator functions, with $\Delta x_1 = \Delta x_2 = 1/512$ and $\mathbb{Q}^1$ for second order approximation

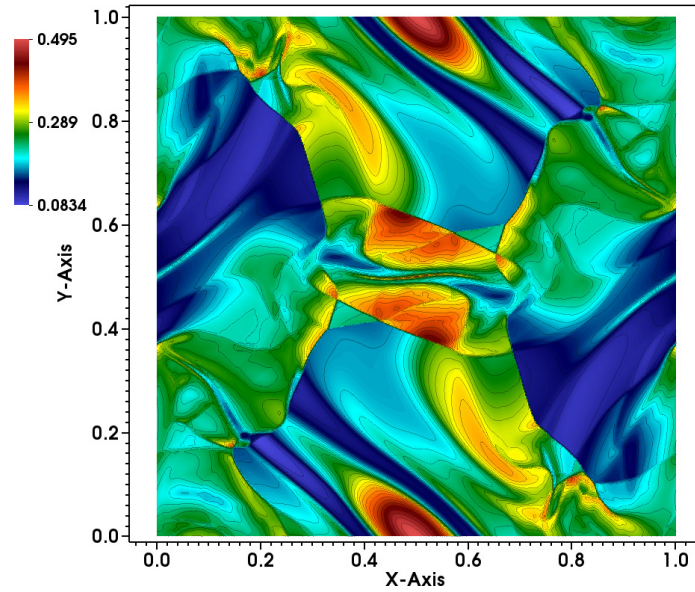Figure 4.32: Density values of the Orszag-Tang test case at $t = 0.5$ seconds, using the ES numerical flux, $\mathbb{Q}^2$ polynomial basis for third order approximation, using a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{512}$, and with the energy as indicator variable
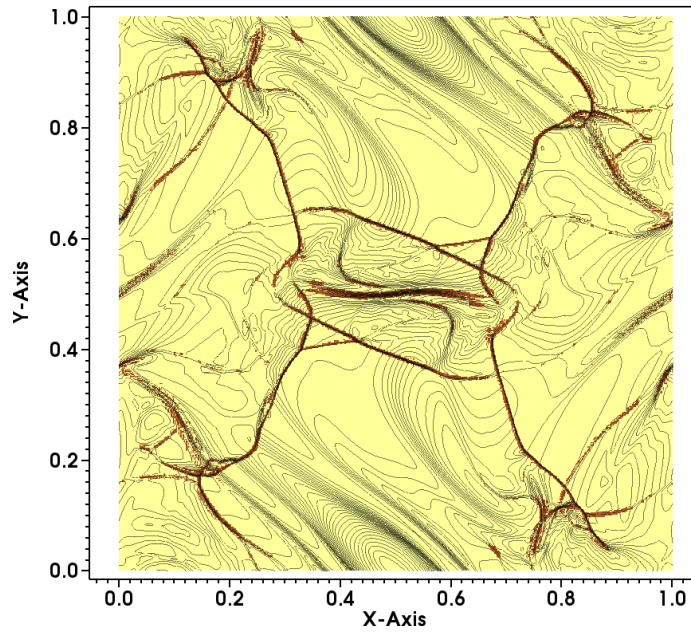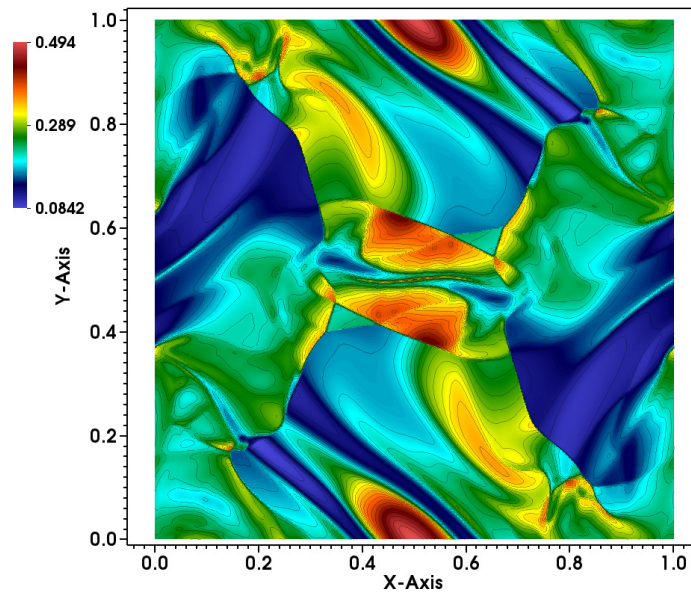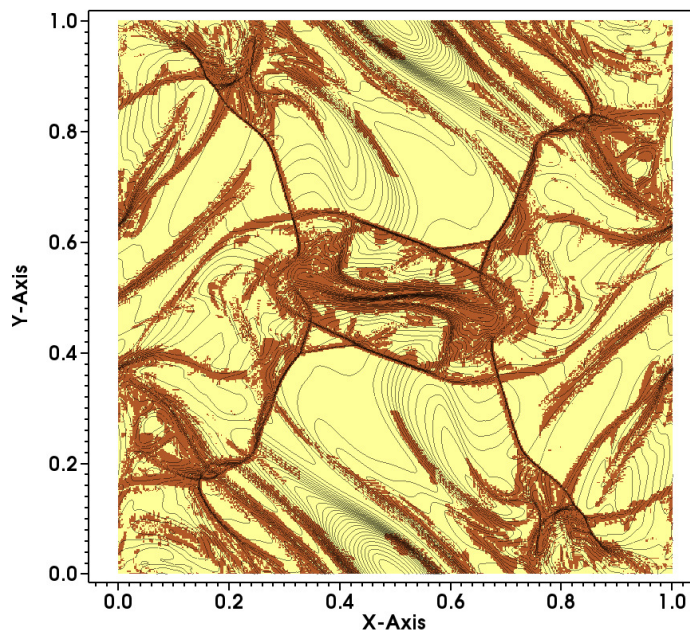
Figure 4.33: Energy indicator for the Orszag-Tang test case present in Figure 4.32 using the energy as indicator functions, with $\Delta x_1 = \Delta x_2 = 1/512$ and $\mathbb{Q}^2$ for third order approximation

$\mathbb{Q}^1$.

Figure 4.34 shows a comparison of the color maps for the $\nabla \cdot \mathbf{B}$ condition using the ES numerical flux and both $\mathbb{Q}^1$ and $\mathbb{Q}^2$ polynomial basis. Comparing with the case in which the LXF flux is used with the shock indicator (Figure 4.27(b)), the areas with a higher magnitude of the magnetic field's divergence are sharper and the maximum magnitudes are lower when the ES numerical flux and of the same order is used (Figure 4.34(a)).

When the $\mathbb{Q}^2$ polynomials basis are used together with the ES numerical flux (Figure 4.34(b)), the regions with a higher magnitude of the divergence are sharper than for the $\mathbb{Q}^1$ case. However, it is clear that more work has to be done in order to enforce the $\nabla \cdot \mathbf{B} = 0$ condition.

The results obtained from running the Orszag-Tang test case using the `dflo` code with the ES numerical flux are compared with the results from the Athena code. Figure 4.35 and 4.36 show two slices at $x_2 = 0.3125$ and $x_2 = 0.4277$ respectively, running the `dflo` code for two different grid sizes $\Delta x_1 = \Delta x_2 = \{1/256, 1/512\}$ and using the $\mathbb{Q}^1$ Legendre polynomial basis for second order approximation.

On the other hand, Figure 4.37 and 4.38 show two slices at $x_2 = 0.3125$ and $x_2 = 0.4277$ respectively, running the `dflo` code for two different grid sizes $\Delta x_1 = \Delta x_2 = \{1/256, 1/512\}$ and using $\mathbb{Q}^2$ Legendre polynomial basis for third order approximation.

It can be seen from these plots that the results from the `dflo` code coincide greatly with the Athena results.

(a) $\mathbb{Q}^1$ Legendre basis

(b) $\mathbb{Q}^2$ Legendre basis
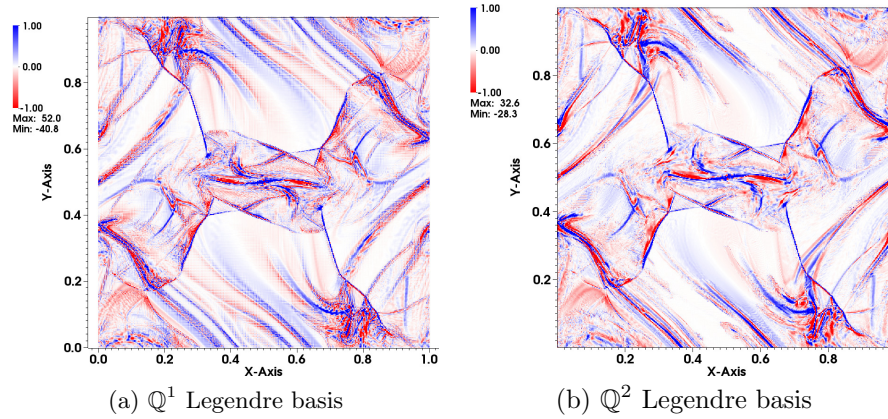
Figure 4.34: Values of the $\nabla \cdot \mathbf{B}$ for the Orszag-Tang test case at $t = 0.5$ seconds, using the ES numerical flux, for both the $\mathbb{Q}^1$ and the $\mathbb{Q}^2$ polynomial basis plus a grid size of $\Delta x_1 = \Delta x_2 = \frac{1}{512}$.
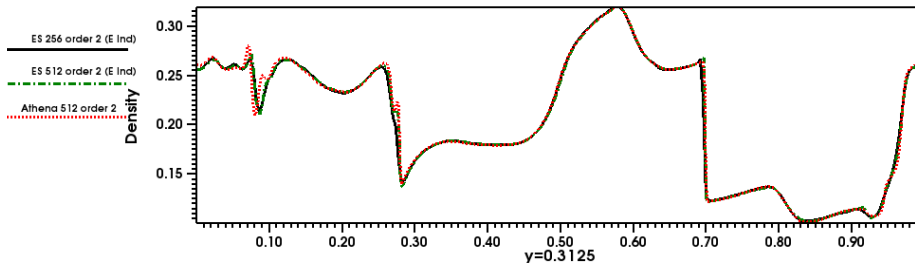
Figure 4.35: Cut comparison ($x_2 = 0.3125$) of the second order test case solution computed using the dflo code for two different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/256, 1/512\}$) using the ES numerical flux, and the results from the Athena code for $\Delta x_1 = \Delta x_2 = 1/512$.
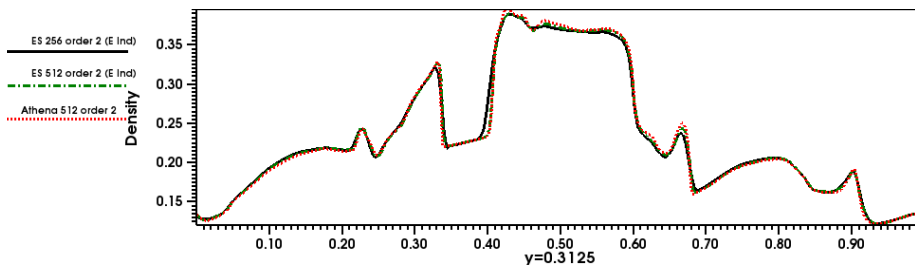
Figure 4.36: Cut comparison ($x_2 = 0.4277$) of the second order test case solution computed using the dflo code for two different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/256, 1/512\}$) using the ES numerical flux, and the results from the Athena code for $\Delta x_1 = \Delta x_2 = 1/512$.
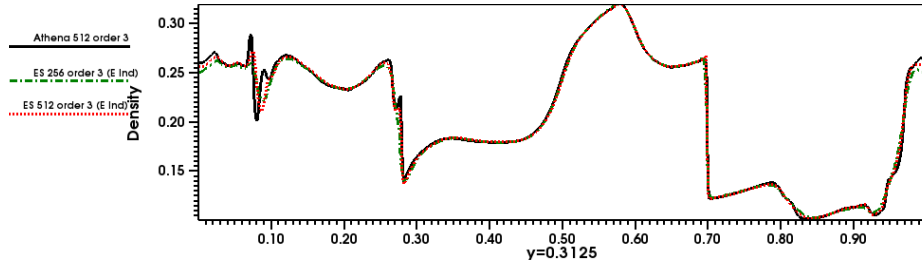
Figure 4.37: Cut comparison (x_2 = 0.3125) of the third order test case solution computed using the dflo code for two different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/256, 1/512\}$) using the ES numerical flux, and the results from the Athena code for $\Delta x_1 = \Delta x_2 = 1/512$.

Figure 4.38: Cut comparison (x_2 = 0.4277) of the third order test case solution computed using the dflo code for two different grid sizes ($\Delta x_1 = \Delta x_2 = \{1/256, 1/512\}$) using the ES numerical flux, and the results from the Athena code for $\Delta x_1 = \Delta x_2 = 1/512$.

Summary

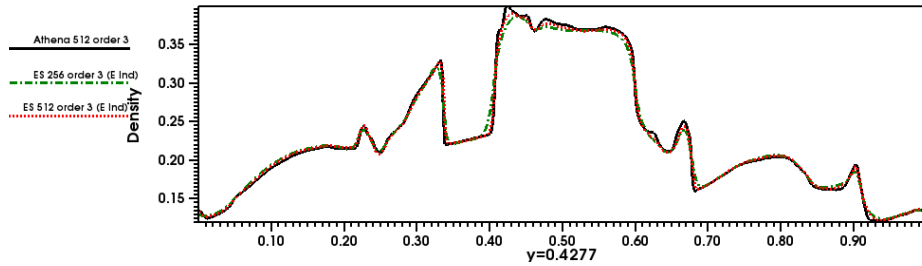Some test cases for the Euler equations and the MHD system were presented in this chapter, together with some additional implementation details for the MHD system. The following ideas can be drawn from the results presented here:

- A set of well known test cases, were implemented using the `dflo` code, for both the Euler system and the MHD system.

- The test cases for the Euler system show that the code is able to produce results which are comparable to what has been already shown in the literature.

- For the hydrodynamical test cases it can be seen that the KXRCF shock indicator is a useful tool that can be activated to reduce the amount of limiting applied in the computations, without degrading the quality of the results. However, it is also important to notice that the KXRCF shock indicator has a big dependence on the variable used as indicator. As it was shown for the Sedov blast wave test case, the shock detector did not work as the density was used as indicator variable.

- The positivity preserving limiter performs successfully when it is used for near-vacuum conditions, as it was shown with the Sedov blast wave test case. However, this limiter has to be always used in combination with a slope limiter. This is due to the fact that it does not eliminate spurious oscillations, but just ensures that they remain within the designed bounds (in this case $\rho > 0$ and $p > 0$).

- Two test cases for the MHD system were implemented, the Polarized Alfvèn wave (for four different grid-sizes) and the Orszag-Tang vortex (for two different grid-sizes). These test cases were simulated using two different numerical fluxes, a Lax-Friedrichs-type (LXF) flux and the Entropy Stable (ES) flux.

- The convergence rates computed for the Alfvèn wave test case coincide to the expected results. The magnitude of the L^2-errors resulting from the simulations where the ES numerical flux were used are smaller from those resulting as the LXF numerical flux is used.

- The results of the `dflo` code are shown to be competitive when they are contrasted with the results of the Athena code for the Orszag-Tang test case.

In the following section some concluding remarks will be shown, together with possible ideas to enhance the current work with future projects.

Discussion and Conclusions

An implementation based on `deal.II C++` libraries was developed to numerically solve the compressible Euler equations of gas dynamics and the ideal Magneto- hydrodynamical model. The implementation uses a discontinuous Galerkin scheme on a Cartesian mesh.

Different tools to control spurious oscillations around discontinuous solutions were reviewed. The tools mainly used for the numerical results were a TVD-type limiter, the positivity preserving limiter and the KXRCF shock indicator. The combination of these three tools has produce competitive results in both systems of equations under study.

The KXRCF shock indicator seems to detect well the locations of the shocks in the 2D examples, however special care has to be taken in the selection of the indicator variable. It was shown in the simulations of the Sedov blast wave test case, that the shock indicator cannot detect the initial pressure discontinuity when the density is used as indicator variable.

The implementation of the shock indicator can be justified as it avoids the usage of limiters all over the domain, which in turn may lead to an improvement in the efficiency of the algorithm.

The positivity preserving limiter may be a very useful tool when the RKDG scheme is used in astrophysical simulations, where near vacuum conditions are often encounter. This limiting procedure may guarantee positivity of variables like density an pressure (or temperature), while preserving the high order approximation. Thus, it is worthwhile to investigate the behavior of the scheme and the limiter in these applications.

A DG scheme for the MHD system has been presented in this work, and it has been proven that the semi-discrete scheme is entropy stable. The numerical experiments with the scheme were performed using an explicit third order SSP RK time integration, in combination with a Lax-Friedrichs type flux or an entropy stable flux, which arises from the proof of the entropy stability for the semi-discrete scheme.

For a smooth test case the expected convergent rates are obtained for $\mathbb{Q}^1$ and $\mathbb{Q}^2$ polynomial basis, and the results of a test case with discontinuous solutions shows to be competitive when they are compared to a well known numerical implementation like the Athena code.

In order to ensure the divergence free condition of the magnetic field for the MHD system, the Powell terms were added to the DG scheme. The numerical simulations show that this technique preserves the values of the divergence close to their initial value for smooth solutions. In the case of discontinuous solutions the condition is not preserved anymore around the shocks.

Concerning the divergence condition, the results could be improved in different ways. One option could be projecting the magnetic components of the solution into a locally divergence free set of polynomial basis functions every certain amount of time steps. However, it has to be check whether this affects the entropy stability result.

The second option is to completely replace the polynomial basis of the magnetic field by a set of locally divergence free polynomial basis. In this case maybe a new numerical flux needs to be designed, since the proof of the entropy stability for the semi-discrete DG scheme in the appendix leads to a different restriction for the numerical flux when a locally divergence free polynomial basis is used. However, this needs to be studied through extensive numerical tests in a future work.

Appendix A

Additional remarks

A.1 Some notation

The following remarks will be needed for a better understanding of the ideas presented in this appendix. For a two times differentiable function $G : \mathbb{R}^m \to \mathbb{R}$

$$G'(\mathbf{r})^T := [\partial_{r_1} G(\mathbf{r}).\partial_{r_2} G(\mathbf{r}), ..., \partial_{r_m} G(\mathbf{r})]$$

and $G''(\mathbf{r})$ is defined as

$$G''(\mathbf{r}) := \left[\partial_{r_i} \partial_{r_j} G(\mathbf{r})\right].$$

where $\partial_{r_j} G(\mathbf{r})$ denotes de partial derivative of the function G with respect to the variable r_j. The matrix $G''(\mathbf{r})$ is a Hessian matrix, and since $G(\mathbf{r})$ is sufficiently smooth it is symmetric. If in addition G is a convex function on a convex set $\mathcal{X} \subset \mathbb{R}^m$, then the **Legendre tranformation** is a function $\mathcal{G}$ defined by

$$\mathcal{G}(\mathbf{s}) = \sup_{\mathbf{r} \in \mathcal{X}} \{\mathbf{s} \cdot \mathbf{r} - G(\mathbf{r})\}, \quad \mathbf{s} \in \mathcal{X}^* \tag{A.1}$$

where sup is the supremum, and the set $\mathcal{X}^*$ is defined as

$$\mathcal{X}^* := \{\mathbf{s} \in \mathbb{R}^m : \sup_{\mathbf{r} \in \mathcal{X}} (\mathbf{s} \cdot \mathbf{r} - G(\mathbf{r})) < \infty\}.$$

The function $\mathcal{G}$ is called the **convex conjugate** function of G.

A.2 Godunov symmetrization of the MHD system

In an attempt to symmetrize the MHD equations, the thermal entropy

$$s = \ln\left(p\rho^{-\gamma}\right) \tag{A.2}$$

is applied to the MHD system. The following equation can be reached from the MHD system

$$\partial_t s + \mathbf{u} \cdot \nabla s + (\gamma - 1)\frac{\mathbf{u} \cdot \mathbf{B}}{p} \nabla \cdot \mathbf{B} = 0.$$

This equation can be written as

$$\partial_t(\rho s) + \nabla(\rho s \mathbf{u}) + (\gamma - 1)\frac{\rho(\mathbf{u} \cdot \mathbf{B})}{p}\nabla \cdot \mathbf{B} = 0,$$

using the conservation of mass equation of the MHD system. From this equation the pair (U, F_κ), given by

$$U = -\frac{\rho s}{\gamma - 1}, \quad F_\kappa = -\frac{\rho s u_\kappa}{\gamma - 1}, \quad \kappa \in \{1, 2, 3\}, \tag{A.3}$$

can be recover and used as an entropy-flux pair when $\nabla \cdot \mathbf{B} = 0$ is assumed. The corresponding entropy variables $\mathbf{w}$ are given by

$$\mathbf{w} = U'(\mathbf{q}) = \left[\frac{\gamma - s}{\gamma - 1} - \beta|\mathbf{u}|^2, 2\beta\mathbf{u}, 2\beta\mathbf{B}, -2\beta\right]^T, \quad \text{with} \quad \beta = \frac{\rho}{2p} \tag{A.4}$$

However, the transformation $\mathbf{q} \to \mathbf{w}$ fails to symmetrize the MHD system, since instead of $F'_\kappa = U'(\mathbf{q})\mathbf{f}'_\kappa(\mathbf{q})$ one obtains[1]

$$F'_\kappa = U'(\mathbf{q})\mathbf{f}'_\kappa(\mathbf{q}) - 2\beta(\mathbf{u} \cdot \mathbf{B})B'_\kappa(\mathbf{q}). \tag{A.5}$$

In order to overcome this issue, the approach of Godunov in [4] is followed in this work. Therefore, a homogeneous function of degree one $\varphi(\mathbf{w})$ which satisfies

$$\mathbf{w} \cdot \varphi'(\mathbf{w})^T = \varphi(\mathbf{w}) \tag{A.6}$$

is used to modify the MHD system as

$$\partial_t\mathbf{q} + \partial_{x_\kappa}\mathbf{f}_\kappa(\mathbf{q}) + \varphi'(\mathbf{w})\nabla \cdot \mathbf{B} = 0, \tag{A.7}$$

which is consistent, since $\nabla \cdot \mathbf{B} = 0$. Using the Legendre transformation showed in (A.1), let the supremum be attached to $\mathbf{w}^* = \mathbf{w}(\mathbf{q})$, such that

$$\mathcal{U}(\mathbf{w}(\mathbf{q})) = \mathbf{w}(\mathbf{q}) \cdot \mathbf{q} - U(\mathbf{q}) \tag{A.8}$$

where $U(\mathbf{q})$ is given by (A.3). Likewise define

$$\mathcal{F}_\kappa(\mathbf{w}(\mathbf{q})) = \mathbf{w}(\mathbf{q}) \cdot (\mathbf{f}_\kappa(\mathbf{q}) + \varphi'(\mathbf{w}(\mathbf{q}))B_\kappa) - F(\mathbf{q}) \tag{A.9}$$

where $F(\mathbf{q})$ is also given by (A.3). For the sake of simplicity from now on $\mathbf{w}$ will be used instead of $\mathbf{w}(\mathbf{q})$. The functions $\mathcal{U}(\mathbf{w})$ and $\mathcal{F}_\kappa(\mathbf{w})$ satisfy

$$\mathbf{q} = \mathcal{U}'(\mathbf{w})^T, \quad \mathbf{f}_\kappa = \mathcal{F}'_\kappa(\mathbf{w})^T - \varphi'(\mathbf{w})^T B_\kappa. \tag{A.10}$$

Replacing equation (A.10) in (A.7) follows

$$\partial_\mathbf{w}\mathbf{q}\partial_t\mathbf{w} + \partial_\mathbf{w}\mathbf{f}_\kappa\partial_{x_\kappa}\mathbf{w} + \varphi'(\mathbf{w})^T\nabla \cdot \mathbf{B} = 0$$

$$\Leftrightarrow \mathcal{U}''(\mathbf{w})\partial_t\mathbf{w} + \left(\mathcal{F}''_\kappa(\mathbf{w}) - \varphi''(\mathbf{w})B_\kappa - \varphi'(\mathbf{w})^T\partial_\mathbf{w}B_\kappa\right)\partial_{x_\kappa}\mathbf{w} + \varphi'(\mathbf{w})^T\nabla \cdot \mathbf{B} = 0$$

$$\Leftrightarrow \mathcal{U}''(\mathbf{w})\partial_t\mathbf{w} + \left(\mathcal{F}''_\kappa(\mathbf{w}) - \varphi''(\mathbf{w})B_\kappa\right)\partial_{x_\kappa}\mathbf{w} = 0$$

---

[1]Let us use for simplicity the Einstein summation convention on the repeated index κ.

with $\mathcal{U}''(\mathbf{w})$ semi-positive definite and $\mathcal{F}_\kappa''(\mathbf{w}) - \varphi''(\mathbf{w})B_\kappa$ symmetric, the equation is then in symmetric form. Then, taking the derivative with respect to the conserved variables in (A.8) and (A.9) follows

$$\mathbf{w} = U'(\mathbf{q})^T \quad \text{and} \quad F_\kappa'(\mathbf{q}) = U'(\mathbf{q})\mathbf{f}_\kappa'(\mathbf{q}) + \varphi(\mathbf{w})B_\kappa'(\mathbf{q}) \qquad (A.11)$$

and taking dot product of the entropy variable $\mathbf{w}$ with equation (A.7)

$$\begin{aligned}
0 &= \mathbf{w} \cdot \left(\partial_t \mathbf{q} + \partial_{x_\kappa} \mathbf{f}_\kappa + \varphi'(\mathbf{w})^T \nabla \cdot \mathbf{B} \right) \\
&= \partial_t U + \mathbf{w} \cdot \partial_{\mathbf{q}} \mathbf{f}_\kappa \partial_{x_\kappa} \mathbf{q} + \varphi(\mathbf{w})B_\kappa'(\mathbf{q})\partial_{x_\kappa} \mathbf{q} \\
&= \partial_t U + \mathbf{w} \cdot \left(\partial_{\mathbf{q}} \mathbf{f}_\kappa + \varphi(\mathbf{w})B_\kappa'(\mathbf{q}) \right) \partial_{x_\kappa} \mathbf{q} \\
&= \partial_t U + \partial_{\mathbf{q}} F_\kappa \partial_{x_\kappa} \mathbf{q} \\
&= \partial_t U + \partial_{x_\kappa} F_\kappa.
\end{aligned}$$

Comparing (A.5) and (A.11), the function $\varphi(\mathbf{w})$ can be computed as

$$\varphi(\mathbf{w}) = 2\beta(\boldsymbol{u} \cdot \mathbf{B}). \qquad (A.12)$$

This function can be written as

$$\varphi(\mathbf{w}) = -\frac{\mathbf{w}_2\mathbf{w}_5 + \mathbf{w}_3\mathbf{w}_6 + \mathbf{w}_4\mathbf{w}_7}{\mathbf{w}_8}$$

which is homogeneous of degree one, and

$$\varphi'(\mathbf{w})^T = [0, \mathbf{B}, \boldsymbol{u}, \boldsymbol{u} \cdot \mathbf{B}].$$

By using (A.3) and (A.4) in (A.8), the conjugate variables are given by

$$\mathcal{U} = \mathbf{w} \cdot \mathbf{q} - U = \rho + \beta|\mathbf{B}|^2 \qquad (A.13)$$

and similarly by using (A.3) and (A.4) in (A.9) follows

$$\mathcal{F}_\kappa = \mathbf{w} \cdot \mathbf{f}_\kappa' + \varphi B_\kappa - F_\kappa = \rho u_\kappa + \beta u_\kappa |\mathbf{B}|^2 = \mathcal{U}u_\kappa.$$

A.3 DG semi-discrete scheme's entropy stability for MHD

The numerical scheme presented in (4.2) can be rewritten as follows

$$\int_{\tau_\ell} \left[(\partial_t \mathbf{q}_h) \cdot \mathbf{v}_h - \mathbf{f}_\kappa(\mathbf{q}_h) \cdot \partial_{x_\kappa} \mathbf{v}_h + (\nabla \cdot \mathbf{B}_h)\varphi'(\mathbf{w}_h) \cdot \mathbf{v}_h \right] d\mathbf{x}$$
$$+ \int_{\partial\tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) + \frac{1}{2}\varphi'(\mathbf{w}_h^-)\Delta(B_{\kappa,h}n_\kappa) \right] \cdot \mathbf{v}_h^- d\sigma = 0, \quad (A.14)$$

where $\mathbf{v}_h := [v_h, ..., v_h]^T$, for all $v_h \in \mathcal{V}_h^k$. Replacing $\mathbf{v}_h$ in the semi-discrete scheme (A.14) by $\mathbf{w}_h$, taking into account that $\mathbf{w}_h \cdot \varphi'(\mathbf{w}_h) = \varphi(\mathbf{w}_h)$ and using (A.10)

$$\frac{d}{dt} \int_{\tau_\ell} U(\mathbf{w}_h)dx - \int_{\tau_\ell} \left[\mathbf{f}_\kappa(\mathbf{q}_h) \cdot \partial_{x_\kappa} \mathbf{w}_h - \varphi(\mathbf{w}_h) \nabla \cdot \mathbf{B}_h \right] d\mathbf{x}$$
$$+ \int_{\partial\tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \mathbf{w}_h^- + \frac{1}{2}\varphi(\mathbf{w}_h^-)\Delta(B_{\kappa,h}n_\kappa) \right] d\sigma = 0. \quad (A.15)$$

The integration by parts formula provides

$$\frac{d}{dt} \int_{\tau_\ell} U(\mathbf{w}_h) dx - \int_{\tau_\ell} \left[\mathbf{f}_\kappa(\mathbf{q}_h) + \varphi'(\mathbf{w}_h) B_{\kappa,h} \right] \cdot \partial_{x_\kappa} \mathbf{w}_h d\mathbf{x}$$
$$\int_{\partial \tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \mathbf{w}_h^- + \varphi(\mathbf{w}_h^-) \overline{B}_{\kappa,h} n_\kappa \right] d\sigma = 0. \quad \text{(A.16)}$$

Using (A.10) and the Gauss divergence theorem

$$\frac{d}{dt} \int_{\tau_\ell} U(\mathbf{w}_h) dx +$$
$$\int_{\partial \tau_\ell} \left[-\mathcal{F}_\kappa(\mathbf{w}_h^-) n_\kappa + \varphi(\mathbf{w}_h^-) \overline{B}_\kappa n_\kappa + \mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \mathbf{w}_h^- \right] d\sigma = 0. \quad \text{(A.17)}$$

Then, the following relations

$$\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \mathbf{w}_h^- = \mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \overline{\mathbf{w}}_h - \frac{1}{2} \mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \Delta \mathbf{w}_h,$$

$$-\mathcal{F}_\kappa(\mathbf{w}_h^-) n_\kappa = \frac{1}{2} \Delta(\mathcal{F}_\kappa n_\kappa) - \overline{\mathcal{F}}_\kappa n_\kappa,$$

$$\varphi(\mathbf{w}_h^-) \overline{B}_\kappa n_\kappa = -\frac{1}{2} \overline{B}_\kappa n_\kappa \Delta \varphi + \overline{\varphi} \overline{B}_\kappa n_\kappa$$

are used, and it follows

$$\frac{d}{dt} \int_{\tau_\ell} U(\mathbf{w}_h) dx + \int_{\partial \tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \overline{\mathbf{w}}_h + \overline{\varphi} \overline{B}_\kappa n_\kappa - \overline{\mathcal{F}}_\kappa n_\kappa \right] d\sigma$$

$$= \frac{1}{2} \int_{\partial \tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \Delta \mathbf{w}_h - \Delta(\mathcal{F}_\kappa n_\kappa) + \overline{B}_\kappa n_\kappa \Delta \varphi \right] d\sigma$$

$$= \frac{1}{2} \int_{\partial \tau_\ell} \left[\mathcal{H}^*(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \Delta \mathbf{w}_h - \Delta(\mathcal{F}_\kappa n_\kappa) + \overline{B}_\kappa n_\kappa \Delta \varphi \right] d\sigma$$

$$- \frac{1}{4} \int_{\partial \tau_\ell} \Delta \mathbf{w}_h^T \mathbf{D}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \Delta \mathbf{w} d\sigma,$$

since the numerical flux $\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n})$ can be written as a central flux $\mathcal{H}^*(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n})$ plus a dissipation flux $\frac{1}{2} \Delta \mathbf{w}_h^T \mathbf{D}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n})$ with a semi-positive defined matrix $\mathbf{D}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n})$.

Finally, the entropy stability result

$$\frac{d}{dt} \int_{\tau_\ell} U(\mathbf{w}_h) dx + \sum_{\kappa \subseteq \tau_\ell} \int_{\partial \tau_\ell} \left[\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \overline{\mathbf{w}}_h + \overline{\varphi} \overline{B}_\kappa n_\kappa - \overline{\mathcal{F}}_\kappa n_\kappa \right] d\sigma$$

$$= -\frac{1}{4} \int_{\partial \tau_\ell} \Delta \mathbf{w}_h^T \mathbf{D}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \Delta \mathbf{w} d\sigma$$

$$\leq 0$$

is obtained, when the condition

$$\mathcal{H}^*(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \Delta \mathbf{w}_h = \Delta(\mathcal{F}_\kappa n_\kappa) - \left(\overline{B}_\kappa n_\kappa \right) \Delta \varphi \quad \text{(A.18)}$$

is satisfied, since the term $\mathcal{H}(\mathbf{q}_h^-, \mathbf{q}_h^+, \mathbf{n}) \cdot \overline{\mathbf{w}}_h + \overline{\varphi} \overline{B}_\kappa n_\kappa - \overline{\mathcal{F}}_\kappa n_\kappa$ is a consistent numerical flux for the entropy flux (see equation A.9)

$$F_\kappa n_\kappa = \mathbf{w} \cdot \left(\mathbf{f}_\kappa n_\kappa + \varphi' B_\kappa n_\kappa \right) - \mathcal{F}_\kappa n_\kappa.$$

A.4 Entropy stable numerical flux for MHD

In Section A.3, it was shown that under the condition (A.18) the DG semi-discrete scheme becomes entropy stable. In the following, a numerical flux which respects the condition (A.18) is designed.

Let the arithmetic average and the logarithmic average be given by

$$\overline{\eta} = \frac{1}{2}(\eta^- + \eta^+), \qquad \hat{\eta} = \frac{\eta^+ - \eta^-}{\ln(\eta^+) - \ln(\eta^-)}.$$

For the case in which $\eta^+ \approx \eta^-$, a numerically stable procedure of the logarithmic average can be found in [41]. Additionally, the following property is satisfied

$$\Delta(\eta\xi) = \overline{\xi}\Delta\eta + \overline{\eta}\Delta\xi$$

for the difference of two quantities $\Delta\eta = \eta^+ - \eta^-$. The difference in the entropy variables can be written in terms of the differences in ρ, u_1, u_2, u_3, B_1, B_2, B_3, β, as

$$\Delta\mathbf{w} = \begin{bmatrix} \frac{1}{\rho}\Delta\rho - 2\overline{\beta}(\overline{u}_1\Delta u_1 + \overline{u}_2\Delta u_2 + \overline{u}_3\Delta u_3) + \left[\frac{1}{(\gamma-1)\beta} - |\overline{\mathbf{u}}|^2\Delta\beta\right] \\ 2\overline{\beta}\Delta u_1 + 2\overline{u}_1\Delta\beta \\ 2\overline{\beta}\Delta u_2 + 2\overline{u}_2\Delta\beta \\ 2\overline{\beta}\Delta u_3 + 2\overline{u}_3\Delta\beta \\ 2\overline{\beta}\Delta B_1 + 2\overline{B}_1\Delta\beta \\ 2\overline{\beta}\Delta B_2 + 2\overline{B}_2\Delta\beta \\ 2\overline{\beta}\Delta B_3 + 2\overline{B}_3\Delta\beta \\ -2\Delta\beta \end{bmatrix}, \quad \mathcal{H}^* = \begin{bmatrix} \mathcal{H}_1^* \\ \mathcal{H}_2^* \\ \mathcal{H}_3^* \\ \mathcal{H}_4^* \\ \mathcal{H}_5^* \\ \mathcal{H}_6^* \\ \mathcal{H}_7^* \\ \mathcal{H}_8^* \end{bmatrix}.$$

Then

$$\begin{aligned} \Delta\mathbf{w} \cdot \mathcal{H}^* = & \frac{\mathcal{H}_1^*}{\hat{\rho}}\Delta\rho + (-2\overline{u}_1\overline{\beta}\mathcal{H}_1^* + -2\overline{\beta}\mathcal{H}_2^*)\Delta u_1 + (-2\overline{u}_2\overline{\beta}\mathcal{H}_1^* + -2\overline{\beta}\mathcal{H}_3^*)\Delta u_2 \\ & + (-2\overline{u}_3\overline{\beta}\mathcal{H}_1^* + -2\overline{\beta}\mathcal{H}_4^*)\Delta u_3 \\ & + 2\overline{\beta}\mathcal{H}_5^*\Delta B_1 + 2\overline{\beta}\mathcal{H}_6^*\Delta B_2 + 2\overline{\beta}\mathcal{H}_7^*\Delta B_3 \\ & + \left[\left(\frac{1}{(\gamma-1)\beta} - |\overline{\mathbf{u}}|^2\Delta\beta\right)\mathcal{H}_1^* + 2\overline{u}_1\mathcal{H}_2^* + 2\overline{u}_2\mathcal{H}_3^* + 2\overline{u}_3\mathcal{H}_4^* \\ & + 2\overline{B}_1\mathcal{H}_5^* + 2\overline{B}_2\mathcal{H}_6^* + 2\overline{B}_3\mathcal{H}_7^* - 2\mathcal{H}_8^*\right]\Delta\beta. \end{aligned}$$

Similarly, the differences $\Delta(\mathcal{F}_\kappa n_\kappa)$ and $\Delta\varphi$ can be written as

$$\begin{aligned} \Delta(\mathcal{F}_\kappa n_\kappa) \quad = \quad & \Delta(\rho u_\kappa n_\kappa) + \Delta(\beta u_\kappa n_\kappa |\mathbf{B}|^2) \\ = \quad & \overline{\rho}\Delta(u_\kappa n_\kappa) + \overline{u_\kappa n_\kappa}\Delta\rho + \overline{\beta u_\kappa}n_\kappa\Delta|\mathbf{B}|^2 + |\mathbf{B}|^2\Delta(\beta u_\kappa n_\kappa) \\ = \quad & \overline{\rho}\Delta(u_\kappa n_\kappa) + \overline{u_\kappa n_\kappa}\Delta\rho + 2\overline{\beta u_\kappa}n_\kappa(\overline{B_1}\Delta B_1 + \overline{B_2}\Delta B_2 + \overline{B_3}\Delta B_3) \\ & + |\mathbf{B}|^2(\overline{\beta}\Delta(u_\kappa n_\kappa) + \overline{u_\kappa}n_\kappa\Delta\beta) \\ = \quad & \overline{u_n}\Delta\rho + (\overline{\rho} + \overline{\beta|\mathbf{B}|^2})n_\kappa\Delta u_\kappa + \overline{u_n}\overline{|\mathbf{B}|^2}\Delta\beta \\ & + 2\overline{\beta u_n}(\overline{B_1}\Delta B_1 + \overline{B_2}\Delta B_2 + \overline{B_3}\Delta B_3), \end{aligned}$$

where $u_n = u_\kappa n_\kappa$, and

$$
\begin{aligned}
\Delta\varphi =\ & 2\Delta(\beta(\mathbf{u}\cdot\mathbf{B})) \\
=\ & 2\left(\overline{\beta u_1}\Delta B_1 + \overline{\beta u_2}\Delta B_2 + \overline{\beta u_3}\Delta B_3 + \overline{B_1}\Delta(\beta u_1) + \overline{B_2}\Delta(\beta u_2) + \overline{B_3}\Delta(\beta u_3)\right) \\
=\ & 2\overline{\beta B_1}\Delta u_1 + 2\overline{\beta B_2}\Delta u_2 + 2\overline{\beta B_3}\Delta u_3 + 2(\overline{u_1 B_1} + \overline{u_2 B_2} + \overline{u_3 B_3})\Delta\beta \\
& + 2\overline{\beta u_1}\Delta B_1 + 2\overline{\beta u_2}\Delta B_2 + 2\overline{\beta u_3}\Delta B_3.
\end{aligned}
$$

Defining $B_n = B_\kappa n_\kappa$

$$
\begin{aligned}
\Delta(\mathcal{F}_\kappa n_\kappa) - \overline{B_\kappa}n_\kappa\Delta\varphi = &\ \overline{u_n}\Delta\rho \\
& + \left((\overline{\rho} + \overline{\beta|\mathbf{B}|^2})n_1 - 2\overline{\beta B_n}\,\overline{B_1}\right)\Delta u_1 \\
& + \left((\overline{\rho} + \overline{\beta|\mathbf{B}|^2})n_2 - 2\overline{\beta B_n}\,\overline{B_2}\right)\Delta u_2 \\
& + \left((\overline{\rho} + \overline{\beta|\mathbf{B}|^2})n_3 - 2\overline{\beta B_n}\,\overline{B_3}\right)\Delta u_3 \\
& + 2\left(\overline{\beta u_n}\,\overline{B_1} - \overline{\beta u_1}\,\overline{B_n}\right)\Delta B_1 \\
& + 2\left(\overline{\beta u_n}\,\overline{B_2} - \overline{\beta u_2}\,\overline{B_n}\right)\Delta B_2 \\
& + 2\left(\overline{\beta u_n}\,\overline{B_3} - \overline{\beta u_3}\,\overline{B_n}\right)\Delta B_3 \\
& + \left(\overline{u_n|\mathbf{B}|^2} - 2(\overline{u_1 B_1} + \overline{u_2 B_2} + \overline{u_3 B_3})\overline{B_n}\right)\Delta\beta.
\end{aligned}
$$

Equating the coefficients of $\Delta\rho$, Δu_1, Δu_2, Δu_3, ΔB_1, ΔB_2, ΔB_3 and $\Delta\beta$ on both sides of equation A.18, the flux components are obtained as

$$
\begin{aligned}
\mathcal{H}_1^* =\ & \hat{\rho}\overline{u_n} \\
\mathcal{H}_2^* =\ & P^* n_1 + \overline{u_1}\mathcal{H}_1^* - \overline{B_n}\,\overline{B_1}, \quad P^* = \frac{\overline{\rho}}{2\overline{\beta}} + \frac{1}{2}|\mathbf{B}|^2 \\
\mathcal{H}_3^* =\ & P^* n_2 + \overline{u_1}\mathcal{H}_1^* - \overline{B_n}\,\overline{B_2}, \\
\mathcal{H}_4^* =\ & P^* n_3 + \overline{u_1}\mathcal{H}_1^* - \overline{B_n}\,\overline{B_3}, \\
\mathcal{H}_5^* =\ & \frac{1}{\overline{\beta}}\left(\overline{\beta u_n B_1} - \overline{\beta u_1 B_n}\right) \\
\mathcal{H}_6^* =\ & \frac{1}{\overline{\beta}}\left(\overline{\beta u_n B_2} - \overline{\beta u_2 B_n}\right) \\
\mathcal{H}_7^* =\ & \frac{1}{\overline{\beta}}\left(\overline{\beta u_n B_3} - \overline{\beta u_3 B_n}\right) \\
\mathcal{H}_8^* =\ & \frac{1}{2}\left[\frac{1}{(\gamma-1)\hat{\beta}} - \overline{|\mathbf{u}|^2}\right]\mathcal{H}_1^* + \overline{u_1}\mathcal{H}_2^* + \overline{u_2}\mathcal{H}_3^* + \overline{u_3}\mathcal{H}_4^* \\
& + \overline{B_1}\mathcal{H}_5^* + \overline{B_2}\mathcal{H}_6^* + \overline{B_3}\mathcal{H}_7^* - \frac{1}{2}\overline{u_n|\mathbf{B}|^2} + \left(\overline{u_1 B_1} + \overline{u_2 B_2} + \overline{u_3 B_3}\right)\overline{B_n}
\end{aligned}
$$

which is a consistent numerical flux.

Dissipation flux

A dissipation flux can be constructed using the right-eigenvector matrix as in a VFRoe scheme [54]. Writing the modified MHD system in quasilinear form in conservative variables one has

$$
\partial_t \mathbf{q} + \mathbf{A}_\kappa \partial_{x_\kappa} \mathbf{q} = 0
$$

and in symmetric form using entropy variables

$$\tilde{\mathbf{A}}_0 \partial_t \mathbf{w} + \tilde{\mathbf{A}}_\kappa \partial_{x_\kappa} \mathbf{w} = 0, \quad \tilde{\mathbf{A}}_0 = \partial_{\mathbf{w}} \mathbf{q}, \quad \tilde{\mathbf{A}}_\kappa = \mathbf{A}_\kappa \tilde{\mathbf{A}}_0.$$

For any unit vector $\mathbf{n}$, define

$$\mathbf{A}(\mathbf{n}) = \tilde{\mathbf{A}}_\kappa n_\kappa, \quad \tilde{\mathbf{A}}_\kappa n_\kappa = \mathbf{A}(\mathbf{n}) \tilde{\mathbf{A}}_0.$$

Note that $\tilde{\mathbf{A}}_0$ is a right symmetrizer of the matrix $\mathbf{A}(\mathbf{n})$. From the eigenvector scaling theorem [4], it follows that there exist scaled eigenvectors of $\mathbf{A}(\mathbf{n})$ such that

$$\mathbf{A}(\mathbf{n}) = \tilde{\mathbf{R}}(\mathbf{n})\mathbf{\Lambda}(\mathbf{n})\tilde{\mathbf{R}}^{-1}(\mathbf{n}), \quad \tilde{\mathbf{A}}_0 = \tilde{\mathbf{R}}(\mathbf{n})\tilde{\mathbf{R}}^T(\mathbf{n}), \quad \tilde{\mathbf{A}}(\mathbf{n}) = \tilde{\mathbf{R}}(\mathbf{n})\mathbf{\Lambda}(\mathbf{n})\tilde{\mathbf{R}}^T(\mathbf{n}).$$

By the definition of $\tilde{\mathbf{A}}_0$ follows

$$d\mathbf{q} = \tilde{\mathbf{A}}_0 d\mathbf{w}, \quad \text{and hence} \quad \tilde{\mathbf{R}}^{-1}(\mathbf{n}) d\mathbf{q} = \tilde{\mathbf{R}}^T(\mathbf{n}) d\mathbf{w}.$$

A VFRoe-type numerical flux would be of the form

$$\mathcal{H} = \frac{1}{2}(\mathbf{f}_\kappa^- + \mathbf{f}_\kappa^+) n_\kappa - \frac{1}{2}\tilde{\mathbf{R}}(\mathbf{n})\mathbf{\Lambda}(\mathbf{n})\tilde{\mathbf{R}}^{-1}(\mathbf{n})\Delta\mathbf{q}.$$

Then, using the transformation property between $\mathbf{q}$ and $\mathbf{w}$, the dissipation term at the above flux can be rewritten as

$$\frac{1}{2}\tilde{\mathbf{R}}(\mathbf{n})|\mathbf{\Lambda}(\mathbf{n})|\tilde{\mathbf{R}}^{-1}(\mathbf{n})\Delta\mathbf{q} \approx \frac{1}{2}\tilde{\mathbf{R}}(\mathbf{n})|\mathbf{\Lambda}|(\mathbf{n})\tilde{\mathbf{R}}^T(\mathbf{n})\Delta\mathbf{w}$$

Hence, the dissipation matrix is given by

$$\mathbf{D} = \tilde{\mathbf{R}}(\mathbf{n})|\mathbf{\Lambda}|(\mathbf{n})\tilde{\mathbf{R}}^T,$$

where the eigenvectors and eigenvalues are computed at the average states corresponding to the interfaces. Now, defining the following quantities

$$\mathbf{b} = \frac{\mathbf{B}}{\sqrt{\rho}}, \quad a = \sqrt{\frac{\gamma p}{\rho}}, \quad \alpha_f^2 = \frac{a^2 - c_s^2}{c_f^2 - c_s^2}, \quad \alpha_f^2 = \frac{a^2 - c_s^2}{c_f^2 - c_s^2},$$

the fast (c_f) and slow (c_s) speeds as

$$c_f^2 = \frac{1}{2}\left(a^2 + b^2 + \sqrt{(a^2 + b^2)^2 - 4a^2(\mathbf{b}\cdot\mathbf{n})^2}\right),$$

$$c_s^2 = \frac{1}{2}\left(a^2 + b^2 - \sqrt{(a^2 + b^2)^2 - 4a^2(\mathbf{b}\cdot\mathbf{n})^2}\right)$$

and the unit vector $\mathbf{n}^\perp$ orthogonal to $\mathbf{n}$, which satisfies

$$\mathbf{n}^\perp \cdot \mathbf{n} = 0, \quad |\mathbf{n}^\perp| = 1 \quad \text{and} \quad \mathbf{n}^\perp \in \text{span}\{\mathbf{n}, \mathbf{b}\},$$

the expression for the scaled eigenvectors used to construct the matrix $\tilde{\mathbf{R}}(\mathbf{n})$ were taken from [4], and are given by:

Entropy wave and divergence waves $\lambda_{1,2} = \mathbf{u} \cdot \mathbf{n}$

$$\tilde{r}_1 = \sqrt{\frac{\gamma - 1}{\gamma}}\begin{bmatrix} \sqrt{\rho} \\ \mathbf{0} \\ \mathbf{0} \\ 0 \end{bmatrix}, \quad \tilde{r}_2 = \sqrt{\frac{1}{\gamma}}\begin{bmatrix} 0 \\ \mathbf{0} \\ a\mathbf{n} \\ 0 \end{bmatrix}.$$

Alfvén waves $\lambda_{\pm a} = \mathbf{u} \cdot \mathbf{n} \pm \mathbf{b} \cdot \mathbf{n}$

$$\tilde{r}_{\pm a} = \sqrt{\frac{1}{2}} \begin{bmatrix} 0 \\ \mp \frac{\sqrt{p}}{\rho} \left(\mathbf{n}^\perp \times \mathbf{n} \right) \\ \sqrt{\frac{p}{\rho}} \left(\mathbf{n}^\perp \times \mathbf{n} \right) \\ 0 \end{bmatrix}.$$

Fast waves $\lambda_{\pm f} = \mathbf{u} \cdot \mathbf{n} \pm c_f$

$$\tilde{r}_{\pm f} = \sqrt{\frac{1}{2\gamma}} \begin{bmatrix} \alpha_f \sqrt{\rho} \\ \pm \frac{\alpha_f a^2 \mathbf{n} + \alpha_s a \left((\mathbf{b} \cdot \mathbf{n}^\perp) \mathbf{n} - (\mathbf{b} \cdot \mathbf{n}) \mathbf{n}^\perp \right)}{\sqrt{\rho} c_f} \\ \alpha_s a \mathbf{n}^\perp \\ \alpha_f \sqrt{\rho} a^2 \end{bmatrix}.$$

Slow waves $\lambda_{\pm s} = \mathbf{u} \cdot \mathbf{n} \pm c_s$

$$\tilde{r}_{\pm s} = \sqrt{\frac{1}{2\gamma}} \begin{bmatrix} \alpha_s \sqrt{\rho} \\ \pm \mathrm{sign}(\mathbf{b} \cdot \mathbf{n}) \frac{\alpha_s a (\mathbf{b} \cdot \mathbf{n}) \mathbf{n} + \alpha_f c_f^2 \mathbf{n}^\perp}{\sqrt{\rho} c_f} \\ -\alpha_f a \mathbf{n}^\perp \\ \alpha_s \sqrt{\rho} a^2 \end{bmatrix}.$$

A.5 Eigensystem to transform the MHD equations into characteristic variables

Here the eigensystem presented in [6], which is used to transform the MHD system from conserved to characteristic variables and vise versa will be shown. To ease the notation, it is important to define the following quantities

$$\mathbf{b} = \frac{1}{\sqrt{\rho}} \mathbf{B}, \text{ and then } b_1 = \frac{B_1}{\sqrt{\rho}}, \quad b_2 = \frac{B_2}{\sqrt{\rho}}, \quad b_3 = \frac{B_3}{\sqrt{\rho}},$$

$$b^2 = \frac{|\mathbf{B}|^2}{\rho}, \quad a^2 = \frac{\gamma p}{\rho}$$

It is also important to define the directional vector depending on each Jacobian matrix, which will be denoted by $\mathbf{n}$. For example, for the Jacobian matrix in the first space dimension, it will be $\mathbf{n} = [1, 0, 0]^T$, and the parallel component of the magnetic field will be $\mathbf{B}_\parallel = [B_1, 0, 0]^T$.

In the same fashion, the orthogonal direction is denoted by $\mathbf{n}^\perp$, which will be calculated as a normalized vector orthogonal to $\mathbf{n}$. Specifically, it is defined for the Jacobiam matrix in the first space dimension as $\mathbf{n}^\perp = [0, \beta_2, \beta_3]^T$, which is given in terms of the orthogonal components of the magnetic field $\mathbf{B}_\perp = [0, B_2, B_3]^T$ by

$$\beta_{2,3} = \begin{cases} \frac{B_{2,3}}{\sqrt{B_2^2 + B_3^2}} & \text{if} \quad B_2^2 + B_3^2 \neq 0, \\ \frac{1}{\sqrt{2}} & \text{otherwise.} \end{cases}$$

The Alfvèn (c_a), fast (c_f) and slow (c_s) speeds for the MHD system are given by the expressions

$$c_a = |(\mathbf{b} \cdot \mathbf{n})|,$$

$$c_f^2 = \frac{1}{2}\left(a^2 + b^2 + \sqrt{(a^2+b^2)^2 - 4a^2(\mathbf{b}\cdot\mathbf{n})^2}\right),$$

$$c_s^2 = \frac{1}{2}\left(a^2 + b^2 - \sqrt{(a^2+b^2)^2 - 4a^2(\mathbf{b}\cdot\mathbf{n})^2}\right).$$

Additionally,

$$\alpha_{f,s}^2 = \begin{cases} \frac{|a^2 - c_{s,f}^2|}{c_f^2 - c_s^2} & \text{if } |\mathbf{B}|_\perp^2 \neq 0, \text{ or } \gamma p \neq |\mathbf{b}\cdot\mathbf{n}|^2 \\ \frac{1}{\sqrt{2}} & \text{otherwise.} \end{cases}$$

Then, the list of right (r_j) and left (l_j) eigenvectors for the eigenvalues (λ_j) of the Jacobian matrix is shown, where j denotes the row or column number of the $\mathcal{L}$ and $\mathcal{R}$ matrices respectively.

Fast waves $(\lambda_{1,8} = u_\kappa \mp c_f)$:[2]

$$l_{1,8} = \frac{1}{2a^2}\begin{bmatrix} -\frac{1-\gamma}{2}\alpha_f|\mathbf{u}|^2 \pm g_f \\ (1-\gamma)\alpha_f(\mathbf{u}\mp c_f\mathbf{n}) \pm \alpha_s c_s sgn(\mathbf{B}\cdot\mathbf{n})\mathbf{n}^\perp \\ (1-\gamma)\alpha_f\mathbf{B} + a\alpha_s\sqrt{\rho}\mathbf{n}^\perp \\ -(1-\gamma)\alpha_f \end{bmatrix}^T,$$

$$r_{1,8} = \begin{bmatrix} \alpha_f \\ \alpha_f(\mathbf{u}\mp c_f\mathbf{n}) \pm sgn(\mathbf{B}\cdot\mathbf{n})\alpha_s c_s\mathbf{n}^\perp \\ \frac{\alpha_s a}{\sqrt{\rho}}\mathbf{n}^\perp \\ \alpha_f\left(\frac{1}{2}|\mathbf{u}|^2 + c_f^2 - \gamma_2 a^2\right) \mp g_f \end{bmatrix},$$

where

$$g_f = \alpha_f c_f\mathbf{u}\cdot\mathbf{n} - \alpha_s c_s sgn(\mathbf{B}\cdot\mathbf{n})\mathbf{u}\cdot\mathbf{n}^\perp, \quad \gamma_2 = \frac{\gamma-2}{\gamma-1}.$$

Alfvèn waves $(\lambda_{2,7} = u_\kappa \mp c_a)$:

$$l_{2,7} = \frac{1}{2}\left[g_a, \ -sgn(\mathbf{B}\cdot\mathbf{n})(\mathbf{n}^\perp\times\mathbf{n}), \ \mp\sqrt{\rho}(\mathbf{n}^\perp\times\mathbf{n}), \ 0\right],$$

$$r_{2,7} = \begin{bmatrix} 0 \\ -sgn(\mathbf{B}\cdot\mathbf{n})(\mathbf{n}^\perp\times\mathbf{n}) \\ \mp\frac{1}{\sqrt{\rho}}(\mathbf{n}^\perp\times\mathbf{n}) \\ -sgn(\mathbf{B}\cdot\mathbf{n})(\mathbf{u}\times\mathbf{n}^\perp) \end{bmatrix}.$$

Slow waves $(\lambda_{3,6} = u_\kappa \mp c_s)$:

$$l_{3,6} = \frac{1}{2a^2}\begin{bmatrix} -\frac{1-\gamma}{2}\alpha_s|\mathbf{u}|^2 \pm g_s \\ (1-\gamma)\alpha_s(\mathbf{u}\mp c_s\mathbf{n}) \mp \alpha_f c_f sgn(\mathbf{B}\cdot\mathbf{n})\mathbf{n}^\perp \\ (1-\gamma)\alpha_s\mathbf{B} - a\alpha_f\sqrt{\rho}\mathbf{n}^\perp \\ -(1-\gamma)\alpha_s \end{bmatrix}^T,$$

$$r_{3,6} = \begin{bmatrix} \alpha_s \\ \alpha_s(\mathbf{u}\mp c_s\mathbf{n}) \mp sgn(\mathbf{B}\cdot\mathbf{n})\alpha_f c_f\mathbf{n}^\perp \\ -\frac{\alpha_f a}{\sqrt{\rho}}\mathbf{n}^\perp \\ \alpha_s\left(\frac{1}{2}|\mathbf{u}|^2 + c_s^2 - \gamma_2 a^2\right) - g_f \end{bmatrix},$$

---

[2]Here it is important note that when a pair of indices appears, the first one corresponds to the upper sign and the second one with the lower sign. Also, the sub-index $\kappa = \{1,2\}$ in the eigenvalues indicates the Jacobian matrix $\mathbf{A}_\kappa$ and the non-zero component of $\mathbf{n}$.

where

$$g_s = \alpha_s c_s \mathbf{u} \cdot \mathbf{n} + \alpha_f c_f \, sgn(\mathbf{B} \cdot \mathbf{n}) \mathbf{u} \cdot \mathbf{n}^\perp.$$

Advection and entropy waves $(\lambda_{4,5} = u_\kappa)$:

$$l_4 = \left[1 + \tfrac{1-\gamma}{2a^2} |\mathbf{u}|^2, \ -\tfrac{1-\gamma}{a^2} \mathbf{u}^T, \ -\tfrac{1-\gamma}{a^2} \mathbf{B}^T, \ \tfrac{1-\gamma}{a^2} \right],$$

$$r_4 = \left[1, \mathbf{u}, \mathbf{0}^T, \tfrac{|\mathbf{u}|^2}{2} \right]^T,$$

$$l_5 = \left[0, \quad \mathbf{0}^T, \quad \mathbf{n}^T, \quad 0 \right],$$

$$r_5 = \left[0, \mathbf{0}^T, \mathbf{n}, \mathbf{B} \cdot \mathbf{n} \right]^T,$$

where $\mathbf{0} = [0, 0, 0]^T$.

Bibliography

[1] S. Adjerid and T. C. Massey. Superconvergence of discontinuous galerkin solutions for a nonlinear scalar hyperbolic problem. *Computer Methods in Applied Mechanics and Engineering*, 195(25-28):3331 – 3346, 2006. Discontinuous Galerkin Methods.

[2] W. Bangerth, D. Davydov, T. Heister, L. Heltai, G. Kanschat, M. Kronbichler, M. Maier, B. Turcksin, and D. Wells. The `deal.II` library, version 8.4. *Journal of Numerical Mathematics*, 24, 2016.

[3] G. E. Barter and D. L. Darmofal. Shock capturing with pde-based artificial viscosity for dgfem: Part i. formulation. *Journal of Computational Physics*, 229(5):1810–1827, 2010.

[4] T. J. Barth. Numerical methods for gasdynamic systems on unstructured meshes. In *An introduction to recent developments in theory and numerics for conservation laws*, pages 195–285. Springer, 1999.

[5] R. Biswas, K. D. Devine, and J. E. Flaherty. Parallel, adaptive finite element methods for conservation laws. *Applied Numerical Mathematics*, 14(1):255–283, 1994.

[6] M. Brio and C. C. Wu. An upwind differencing scheme for the equations of ideal magnetohydrodynamics. *Journal of computational physics*, 75(2):400–422, 1988.

[7] A. Burbeau, P. Sagaut, and C.-H. Bruneau. A problem-independent limiter for high-order runge–kutta discontinuous galerkin methods. *Journal of Computational Physics*, 169(1):111–150, 2001.

[8] R. Camassa and D. D. Holm. An integrable shallow water equation with peaked solitons. *Physical Review Letters*, 71(11):1661, 1993.

[9] P. Chandrashekar and C. Klingenberg. Entropy stable finite volume scheme for ideal compressible mhd on 2-d cartesian meshes. *SIAM Journal on Numerical Analysis*, 2016.

[10] Y. Cheng and C.-W. Shu. Superconvergence of discontinuous galerkin and local discontinuous galerkin schemes for linear hyperbolic and convection-diffusion equations in one space dimension. *SIAM Journal on Numerical Analysis*, 47(6):4044–4072, 2010.

[11] B. Cockburn. Discontinuous galerkin methods. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift fr Angewandte Mathematik und Mechanik*, 83(11):731–754, 2003.

[12] B. Cockburn, S. Hou, and C.-W. Shu. The runge-kutta local projection discontinuous galerkin finite element method for conservation laws. iv. the multidimensional case. *Mathematics of Computation*, 54(190):545–581, 1990.

[13] B. Cockburn, S.-Y. Lin, and C.-W. Shu. Tvb runge-kutta local projection discontinuous galerkin finite element method for conservation laws iii: One-dimensional systems. *Journal of Computational Physics*, 84(1):90 – 113, 1989.

[14] B. Cockburn, M. Luskin, C.-W. Shu, and E. Süli. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations. *Mathematics of Computation*, 72(242):577–606, 2003.

[15] 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(186):411–435, 1989.

[16] B. Cockburn and C.-W. Shu. The local discontinuous galerkin method for time-dependent convection-diffusion systems. *SIAM Journal on Numerical Analysis*, 35(6):2440–2463, 1998.

[17] B. Cockburn and C.-W. Shu. The runge–kutta discontinuous galerkin method for conservation laws v: multidimensional systems. *Journal of Computational Physics*, 141(2):199–224, 1998.

[18] B. Cockburn and C.-W. Shu. Runge-kutta discontinuous galerkin methods for convection-dominated problems. *Journal of scientific computing*, 16(3):173–261, 2001.

[19] R. M. Colombo. A 2× 2 hyperbolic traffic flow model. *Mathematical and computer modelling*, 35(5-6):683–688, 2002.

[20] C. De Lellis and L. Székelyhidi. On admissibility criteria for weak solutions of the euler equations. *Archive for rational mechanics and analysis*, 195(1):225–260, 2010.

[21] C. De Lellis and L. Székelyhidi Jr. The euler equations as a differential inclusion. *Annals of mathematics*, pages 1417–1436, 2009.

[22] M. O. Delchini, J. C. Ragusa, and J. Morel. Entropy-based artificial viscosity stabilization for non-equilibrium grey radiation-hydrodynamics. *Journal of Computational Physics*, 296:293–313, 2015.

[23] A. F. Emery. An evaluation of several differencing methods for inviscid fluid flow problems. *Journal of Computational Physics*, 2(3):306–331, 1968.

[24] J. E. Flaherty, L. Krivodonova, J.-F. Remacle, and M. S. Shephard. Aspects of discontinuous galerkin methods for hyperbolic conservation laws. *Finite Elements in Analysis and Design*, 38(10):889–908, 2002. 2001 Robert J. Melosh Medal Competition.

[25] J. P. Gallego-Valencia, C. Klingenberg, and P. Chandrashekar. On limiting for higher order discontinuous galerkin method for 2d euler equations. *Bulletin of the Brazilian Mathematical Society, New Series*, 47(1):335–345, 2016.

[26] T. A. Gardiner and J. M. Stone. An unsplit godunov method for ideal mhd via constrained transport. *Journal of Computational Physics*, 205(2):509–539, 2005.

[27] T. A. Gardiner and J. M. Stone. An unsplit godunov method for ideal mhd via constrained transport in three dimensions. *Journal of Computational Physics*, 227(8):4123–4141, 2008.

[28] S. Genel, M. Vogelsberger, V. Springel, D. Sijacki, D. Nelson, G. Snyder, V. Rodriguez-Gomez, P. Torrey, and L. Hernquist. Introducing the illustris project: the evolution of galaxy populations across cosmic time. *Monthly Notices of the Royal Astronomical Society*, 445(1):175–200, 2014.

[29] J. Glimm. Solutions in the large for nonlinear hyperbolic systems of equations. *Communications on Pure and Applied Mathematics*, 18(4):697–715, 1965.

[30] E. Godlewski and P. Raviart. Hyperbolic systems of conservation laws, mathématiques & applications [mathematics and applications], 3/4. *Ellipses, Paris*, 1991.

[31] S. Gottlieb, D. I. Ketcheson, and C.-W. Shu. *Strong stability preserving Runge-Kutta and multistep time discretizations*. World Scientific, 2011.

[32] S. Gottlieb and C.-W. Shu. Total variation diminishing runge-kutta schemes. *Mathematics of computation of the American Mathematical Society*, 67(221):73–85, 1998.

[33] S. Gottlieb, C.-W. Shu, and E. Tadmor. Strong stability-preserving high-order time discretization methods. *SIAM review*, 43(1):89–112, 2001.

[34] J.-L. Guermond and B. Popov. Viscous regularization of the euler equations and entropy principles. *SIAM Journal on Applied Mathematics*, 74(2):284–305, 2014.

[35] R. Haberman. *Mathematical models: mechanical vibrations, population dynamics, and traffic flow*. SIAM, 1998.

[36] M. Herzog and F. K. Röpke. Three-dimensional hydrodynamic simulations of the combustion of a neutron star into a quark star. *Physical Review D*, 84(8):083002, 2011.

[37] J. S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Springer Publishing Company, Incorporated, 1st edition, 2007.

[38] E. Hewitt and R. E. Hewitt. The gibbs-wilbraham phenomenon: An episode in fourier analysis. *Archive for History of Exact Sciences*, 21(2):129–160, 1979.

[39] U. Hochstrasser. *Orthogonal polynomials - In Abramowitz, Milton and Stegun, Irene A (Eds) Handbook of mathematical functions: with formulas, graphs, and mathematical tables*, volume 55. Courier Corporation, 1964.

[40] H. Holden and N. H. Risebro. A mathematical model of traffic flow on a network of unidirectional roads. *SIAM Journal on Mathematical Analysis*, 26(4):999–1017, 1995.

[41] F. Ismail and P. L. Roe. Affordable, entropy-consistent euler flux functions ii: Entropy production at shocks. *Journal of Computational Physics*, 228(15):5410–5436, 2009.

[42] A. Jameson. Eigenvalues, eigenvectors and symmetrization of the magnetohydrodynamic (mhd) equations. In *Presented at AFOSR Grantees and Contractors Meeting*, 2006.

[43] G.-S. Jiang and C.-c. Wu. A high-order weno finite difference scheme for the equations of ideal magnetohydrodynamics. *Journal of Computational Physics*, 150(2):561–594, 1999.

[44] C. Johnson. *Numerical solution of partial differential equations by the finite element method*. Courier Corporation, 2012.

[45] L. Krivodonova. Limiters for high-order discontinuous galerkin methods. *Journal of Computational Physics*, 226(1):879–896, 2007.

[46] L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon, and J. Flaherty. Shock detection and limiting with discontinuous galerkin methods for hyperbolic conservation laws. *Applied Numerical Mathematics*, 48(3–4):323–338, 2004. Workshop on Innovative Time Integrators for {PDEs}.

[47] D. Kröner. *Numerical schemes for conservation laws*. Advances in numerical mathematics. Wiley, 1997.

[48] S. N. Kružkov. First order quasilinear equations in several independent variables. *Mathematics of the USSR-Sbornik*, 10(2):217, 1970.

[49] D. Kuzmin. Hierarchical slope limiting in explicit and implicit discontinuous galerkin methods. *Journal of Computational Physics*, 257:1140–1162, 2014.

[50] D. Kuzmin and S. Turek. Flux correction tools for finite elements. *Journal of Computational Physics*, 175(2):525 – 558, 2002.

[51] P. D. Lax. *1. Hyperbolic Systems of Conservation Laws and the Mathematical Theory of Shock Waves*. 1973.

[52] R. J. LeVeque and H. C. Yee. A study of numerical methods for hyperbolic conservation laws with stiff source terms. *Journal of computational physics*, 86(1):187–210, 1990.

[53] F. Li and C.-W. Shu. Locally divergence-free discontinuous galerkin methods for mhd equations. *Journal of Scientific Computing*, 22(1):413–442, 2005.

[54] J.-M. Masella, I. Faille, and T. Gallouet. On an approximate go-
dunov scheme. *International Journal of Computational Fluid Dynamics*,
12(2):133–149, 1999.

[55] S. Nayakshin, J. Cuadra, and V. Springel. Simulations of star formation
in a gaseous disc around sgr a*–a failed active galactic nucleus. *Monthly
Notices of the Royal Astronomical Society*, 379(1):21–33, 2007.

[56] D. A. Nield and A. Bejan. *Convection in porous media*. Springer Science
& Business Media, 2006.

[57] S. T. Ohlmann, F. K. Röpke, R. Pakmor, and V. Springel. Hydrodynamic
moving-mesh simulations of the common envelope phase in binary stellar
systems. *The Astrophysical Journal Letters*, 816(1):L9, 2015.

[58] E. S. Oran and J. P. Boris. *Numerical simulation of reactive flow*. Cam-
bridge University Press, 2005.

[59] S. A. Orszag and C.-M. Tang. Small-scale structure of two-
dimensional magnetohydrodynamic turbulence. *Journal of Fluid Mechan-
ics*, 90(01):129–143, 1979.

[60] P.-O. Persson and J. Peraire. *Sub-Cell Shock Capturing for Discontinuous
Galerkin Methods*. Aerospace Sciences Meetings. American Institute of
Aeronautics and Astronautics, Jan 2006. 0.

[61] I. P. P.J. Davis. *Numerical interpolation, differentiation and integration - In
Abramowitz, Milton and Stegun, Irene A (Eds) Handbook of Mathematical
Functions with Formulas, Graphs, and Mathematical Tables*, volume 55.
Courier Corporation, 1964.

[62] K. G. Powell. An approximate riemann solver for magnetohydrodynamics
(that works in more than one dimension). Technical report, 1994.

[63] D. J. Price and M. R. Bate. The impact of magnetic fields on single and
binary star formation. *Monthly Notices of the Royal Astronomical Society*,
377(1):77–90, 2007.

[64] J. Qian, J. Li, and S. Wang. The generalized riemann problems for hy-
perbolic balance laws: A unified formulation towards high order. *arXiv
preprint arXiv:1303.2941*, 2013.

[65] T. Qin, C.-W. Shu, and Y. Yang. Bound-preserving discontinuous galerkin
methods for relativistic hydrodynamics. *Journal of Computational Physics*,
315:323 – 347, 2016.

[66] W. J. Rider and L. G. Margolin. Simple modifications of monotonicity-
preserving limiter. *Journal of Computational Physics*, 174(1):473–488,
2001.

[67] F. Röpke and J. Niemeyer. Delayed detonations in full-star models of type
ia supernova explosions. *Astronomy & Astrophysics*, 464(2):683–686, 2007.

[68] G. Schnücke. *Arbitrary Lagrangian-Eulerian Discontinous Galerkin methods for nonlinear time-dependent first order partial differential equations.* PhD thesis, Universität Würzburg, 2016.

[69] M. Schulz, S. Trimper, and B. Schulz. Variations of the asset prices. *Physical Review E*, 64(2):026104, 2001.

[70] C.-W. Shu. Total-variation-diminishing time discretizations. *SIAM Journal on Scientific and Statistical Computing*, 9(6):1073–1084, 1988.

[71] V. Springel. The cosmological simulation code gadget-2. *Monthly notices of the royal astronomical society*, 364(4):1105–1134, 2005.

[72] V. Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. *Monthly Notices of the Royal Astronomical Society*, 401(2):791–851, 2010.

[73] V. Springel and L. Hernquist. The history of star formation in a λ cold dark matter universe. *Monthly Notices of the Royal Astronomical Society*, 339(2):312–334, 2003.

[74] V. Springel, S. D. White, A. Jenkins, C. S. Frenk, N. Yoshida, L. Gao, J. Navarro, R. Thacker, D. Croton, J. Helly, et al. Simulations of the formation, evolution and clustering of galaxies and quasars. *nature*, 435(7042):629–636, 2005.

[75] V. Springel, N. Yoshida, and S. D. White. Gadget: a code for collisionless and gasdynamical cosmological simulations. *New Astronomy*, 6(2):79–117, 2001.

[76] A. Suresh and H. Huynh. Accurate monotonicity-preserving schemes with runge–kutta time stepping. *Journal of Computational Physics*, 136(1):83–99, 1997.

[77] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction.* Springer Science & Business Media, 2013.

[78] G. Tóth. The $\nabla \cdot b = 0$ constraint in shock-capturing magnetohydrodynamics codes. *Journal of Computational Physics*, 161(2):605–652, 2000.

[79] J. VonNeumann and R. D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *Journal of applied physics*, 21(3):232–237, 1950.

[80] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of computational physics*, 54(1):115–173, 1984.

[81] J. Xin and J. E. Flaherty. Viscous stabilization of discontinuous galerkin solutions of hyperbolic conservation laws. *Applied Numerical Mathematics*, 56(3):444–458, 2006.

[82] L. Yuan and C.-W. Shu. Discontinuous galerkin method based on non-polynomial approximation spaces. *Journal of Computational Physics*, 218(1):295–323, 2006.

[83] X. Zhang and C.-W. Shu. On maximum-principle-satisfying high order schemes for scalar conservation laws. *Journal of Computational Physics*, 229(9):3091 – 3120, 2010.

[84] X. Zhang and C.-W. Shu. On positivity-preserving high order discontinuous galerkin schemes for compressible euler equations on rectangular meshes. *Journal of Computational Physics*, 229(23):8918 – 8934, 2010.

[85] X. Zhang and C.-W. Shu. Positivity-preserving high order discontinuous galerkin schemes for compressible euler equations with source terms. *Journal of Computational Physics*, 230(4):1238 – 1248, 2011.

[86] V. Zingan, J.-L. Guermond, J. Morel, and B. Popov. Implementation of the entropy viscosity method with the discontinuous galerkin method. *Computer Methods in Applied Mechanics and Engineering*, 253:479–490, 2013.

Erklärung

Hiermit erkläre ich, dass ich die eingereichte Doktorarbeit eigenständig, d.h. insbesondere selbständig und ohne Hilfe einer kommerziellen Promotionsberatung angefertigt und keine anderen als die von mir angegebenen Quellen und Hilfsmittel benutzt habe. Würzburg, den May 19, 2017

_____

(Juan Pablo Gallego Valencia)

Erklärung Promotionsvorhaben

Hiermit versichere ich an Eides statt,

a) dass ich die Gelegenheit zum Promotionsvorhaben nicht kommerziell vermittelt bekommen habe und insbesondere nicht eine Person oder Organisation eingeschaltet habe, die gegen Entgelt Betreuer bzw. Betreuerin für die Anfertigung sucht.

b) dass die Regeln der Universität Würzburg über gute wissenschaftliche Praxis eingehalten wurden.

Würzburg, den May 19, 2017

_____

(Juan Pablo Gallego Valencia)

96

Curriculum Vitae

Persönliche Daten:

Name	Juan Pablo Gallego Valencia
Geburtstag	15.03.1984
Geburtsort	La Union (Antioquia), Kolumbien
Nationalität	kolumbianisch

Bildungsweg:

2013-2017	Promotionsstudium in Mathematik am Institut für Mathematik der Universität Würzburg
2010-2013	M. Sc. Technomathematik an der Fakultät für Mathematik, Informatik und Naturwissenschaften, Universität Hamburg
2007-2010	M. Sc. Managemenet Engineering an der Facultad de Minas, Universidad Nacional de Colombia
2001-2006	B. Sc. Control Engineering an der Facultad de Minas, Universidad Nacional de Colombia

Würzburg, den May 19, 2017

_____

(JUAN PABLO GALLEGO VALENCIA)

Publikationsliste

Verzeichnis veröffentlichter Wissenschaftlicher Arbeiten

1. **Gallego-Valencia, J. P.**, Chandrashekar, P. & Klingenberg, C. (2017). A Runge-Kutta discontinuous Galerkin scheme for the ideal Magnetohydrodynamical model. To appear in *Springer Proceedings in Mathematics and Statistics of the International Conference on Hyperbolic Problems: Theory, Numeric and Applications in Aachen 2016.*

2. **Gallego-Valencia, J. P.**, Klingenberg, C. & Chandrashekar, P. (2016). On limiting for higher order discontinuous Galerkin method for 2D Euler equations. *Bulletin of the Brazilian Mathematical Society, New Series*, 47(1), 335-345.

3. **Gallego-Valencia, J. P.**, Löbbert, J., Müthing, S., Bastian, P., Klingenberg, C. & Xia, Y. (2014). Implementing a discontinuous Galerkin method for the compressible, inviscid Euler equations in the DUNE framework. *PAMM*, 14(1), 953-954.

4. Hurtado, S. M., Cadavid, J. M. & **Valencia, J. P. G.** (2012). Pronóstico de la demanda de energía eléctrica horaria en Colombia mediante redes neuronales artificiales. *Revista Facultad de Ingenieria*, (59), 98-107.(Spanisch)

5. López, M. D. R., Arango, P. & **Gallego, J. P.** (2009). Confianza para efectuar compras por internet. *Dyna*, 76(160), 263-272. (Spanisch)

6. **Gallego, J. P.**, Sánchez-Torres, J. D., Vera-Ciro, C. & Rodríguez-Rey, B. A. (2009). Sistemas caóticos y su aplicación a la encriptación de señales. *Revista de la Sociedad Colombiana de Fsica*, 41(2), 436-439. (Spanisch)

Würzburg, den May 19, 2017

—————————————————————

(Juan Pablo Gallego Valencia)