ADER Discontinuous Galerkin schemes

Lecture Notes for the course at the Institute for Theoretical Physics, 3-5 of May 2016, Frankfurt, Germany

Olindo Zanotti

Laboratory of Applied Mathematics, University of Trento, Italy,

The organization of this course has received funding from the European Union's Horizon 2020 Research and Innovation Programme under the project *ExaHyPE*, grant agreement number no. 671698 (call FETHPC-1-2014).

This short course is meant to offer a basic understanding of ADER (Arbitrary high order using Derivatives) Discontinuous Galerkin (DG) schemes for hyperbolic systems of partial differential equations. Theoretical explanations and practical sessions will alternate to provide a fruitful understanding of these flexible schemes, which can be successfully applied to the relativistic framework. The course will cover the following aspects:

- Tue 03.05
  - 14:00-16:00: Introduction to high order methods and the ADER approach
- Wed 04.05
  - 10:00-12:00: Practical session
  - 15:00-17:00: Discontinuous Galerkin schemes
- Thu 05.05
  - 10:00-12:00: Practical session
  - 15:00-17:00: Limiting DG schemes through "a-posteriori" sub-cell limiters

Contents

I thank Luciano Rezzolla and Michael Dumbser for several illuminating discussions undertaken over the years.
Disclaimer: part of the material presented in this lecture has been taken from Chapters 9 and 10 of the book "Relativistic Hydrodynamics", by Luciano Rezzolla and Olindo Zanotti, Oxford University Press (2013) [50], where the reader can find additional details.

Basic classification

- Finite differences numerical schemes evolve in time the point-values of the solution at cell centers.
- Finite volume numerical schemes evolve in time the cell averages of the solution.
- Discontinous Galerkin schemes evolve in time the so-called degrees of freedom of the solution, i.e. the expansion coefficients with respect to given basis functions.

In this course we illustrate the third methodology, but a basic understanding of finite volume methods is also needed and it will be summarized in the following.

Chapter 1

Finite Volume schemes

1.1 Hyperbolic PDEs

We are assuming a system of PDEs in conservative form as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}\left(\mathbf{U}\right) = 0, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \quad t \in \mathbb{R}_0^+, \tag{1.1}$$

where $\mathbf{U}$ is the vector of so-called conserved quantities, while $\mathbf{F}(\mathbf{U}) = (\mathbf{f}, \mathbf{g}, \mathbf{h})$ is a non-linear flux tensor that depends on the state $\mathbf{U}$. A system like (2.2) can always be written as

$$\partial_t \mathbf{U} + \boldsymbol{A} \cdot \boldsymbol{\nabla} \mathbf{U} = 0\,, \tag{1.2}$$

where $\boldsymbol{A}(\mathbf{U}) = \partial \boldsymbol{F}/\partial \mathbf{U}$ is the Jacobian of the flux vector. The system above is said to be *hyperbolic* if the matrix of coefficients $\boldsymbol{A}$ is diagonalisable with a set of real eigenvalues, or eigenspeeds, $\lambda_1, \ldots, \lambda_N$ and a corresponding set of N linearly independent *right eigenvectors* $\boldsymbol{R}^{(1)}, \ldots, \boldsymbol{R}^{(m)}$, such that $\boldsymbol{A}\boldsymbol{R}^{(i)} = \lambda_i \boldsymbol{R}^{(i)}$, $\boldsymbol{\Lambda} = \boldsymbol{R}^{-1}\boldsymbol{A}\boldsymbol{R} = \mathrm{diag}(\lambda_1, \ldots, \lambda_N)$ is the diagonal matrix of eigenvalues and $\boldsymbol{R}$ the matrix of right eigenvectors. Additional definitions of system (1.2) are possible on the basis of the properties of the eigenvalues and eigenvectors. More specifically, the system is said to be *strictly hyperbolic* if the matrix $\boldsymbol{A}$ has a set of eigenvalues that are real and also distinct, while it is said to be symmetric hyperbolic if the matrix $\boldsymbol{A}$ is symmetric, i.e. $\boldsymbol{A} = \boldsymbol{A}^{\mathrm{T}}$. Finally, the system is said to be *weakly hyperbolic* if the matrix $\boldsymbol{A}$ has a set of real eigenvalues but is not diagonalisable, so that it does not have a complete set of eigenvectors.

Remark: A system like (2.2) can always be written like (1.2), while the opposite is not true. Hence, numerical methods for the solution of (2.2) and (1.2) are intrinsecally different.

1.2 Hyperbolic PDEs in physical sciences

There are good physical motivations to conjecture that all dynamic laws of physics can be written as a hyperbolic system, although such a formulation is not always available. Let us list the most prominent examples

- Classical [40] and relativistic [50] inviscid hydrodynamics
- Classical [28] and relativistic [2] (ideal) magnetohydrodynamics
- Relativistic [37] (resistive) magnetohydrodynamics
- Relativistic irreversible thermodynamics [32]
- Radiation hydrodynamics [4]
- Hydrodynamical modeling of semiconductors [3]
- Einstein equations [1]

1.3 Finite volume discretization

We are again assuming a system of PDE in conservative form as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}\left(\mathbf{U}\right) = 0, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \quad t \in \mathbb{R}_0^+, \tag{1.3}$$

where $\mathbf{U}$ is the vector of so-called conserved quantities, while $\mathbf{F}(\mathbf{U}) = (\mathbf{f}, \mathbf{g}, \mathbf{h})$ is a non-linear flux tensor that depends on the state $\mathbf{U}$. On each time-slice let use discretise the spatial domain into J computing cells $I_j = [x_{j-1/2}, x_{j+1/2}]$ of size $\Delta x = x_{j+1/2} - x_{j-1/2}$, with $j = 1, ..., J$. In addition, we define a spacetime control volume as $\Omega_j^{n+1/2} = I_j \times [t^n, t^{n+1}]$ and integrate Eq. (2.2) first in space over I_j

$$\frac{d}{dt}\int_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{U}(x,t)dx = \mathbf{F}(\mathbf{U}(x_{j-1/2},t)) - \mathbf{F}(\mathbf{U}(x_{j+1/2},t))\,, \tag{1.4}$$

where the partial time derivative has now become a total derivative, and then in time between t^n and t^{n+1} to obtain

$$\begin{aligned}\int_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{U}(x,t^{n+1})dx = & \int_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{U}(x,t^n)dx \\ & + \int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{U}(x_{j-1/2},t))dt - \int_{t^n}^{t^{n+1}} \mathbf{F}(\mathbf{U}(x_{j+1/2},t))dt\,.\end{aligned} \tag{1.5}$$

Equation (1.5) represents the integral form of the conservative equations (2.2). We next define two new quantities, the *cell (volume) averages*

$$\mathbf{U}_j^n = \frac{1}{\Delta x}\int_{x_{j-1/2}}^{x_{j+1/2}} \mathbf{U}(x,t^n)dx\,, \tag{1.6}$$

and the *numerical fluxes*

$$\mathbf{F}_{j\pm 1/2} = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{F}[\mathbf{U}(x_{j\pm 1/2},t)]dt\,, \tag{1.7}$$

such that (1.5) is rewritten as

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n + \frac{\Delta t}{\Delta x}(\mathbf{F}_{j-1/2} - \mathbf{F}_{j+1/2})\,. \tag{1.8}$$

Because of the volume averages introduced in the definition (1.6), the numerical methods that will be built in this way are known as *finite-volume methods*.

It is important to stress that although we have introduced a discretisation of space and time, expression (1.8) does not (yet) represent a numerical scheme and is indeed exact as no mathematical approximation has been done yet. The exact *mathematical method* (1.8) becomes an approximate *numerical method* only when an approximation (and hence a truncation error) is introduced for the computation of the cell averages $\mathbf{U}_j$ and of the numerical fluxes $\mathbf{F}_{j\pm 1/2}$.

1.4 Conservative numerical schemes

A numerical scheme is called a *conservative numerical scheme* if it is based on the conservation form (2.2) of the PDEs. More specifically if

1. The numerical flux $\mathbf{F}_{j+1/2}$ (and analogously $\mathbf{F}_{j-1/2}$) depends on the values taken by $\mathbf{U}$ on the neighbouring cells, namely if
$$\mathbf{F}_{j+1/2} = \boldsymbol{\mathcal{F}}(\mathbf{U}_{j-q}^n, \mathbf{U}_{j-q+1}^n, \ldots, \mathbf{U}_{j+r}^n)\,, \tag{1.9}$$
where q and r are integers and $\boldsymbol{\mathcal{F}}$ is a numerical flux function of $q + r + 1$ arguments.

2. The flux function $\boldsymbol{\mathcal{F}}$ reduces to the true physical flux in the case of constant flow, i.e. it satisfies the consistency condition
$$\boldsymbol{\mathcal{F}}(\mathbf{U},\ldots,\mathbf{U}) = \mathbf{F}(\mathbf{U})\,. \tag{1.10}$$

In practice, because for hyperbolic problems information propagates with finite speed and because the CFL condition on the time-step ensures that such information does not move across a single cell over a time-step, it is often convenient to assume that $\mathbf{F}_{j+1/2}$ depends only on the values of $\mathbf{U}$ in the two adjacent cells, i.e. $r = 0 = q$ and
$$\mathbf{F}_{j+1/2} = \boldsymbol{\mathcal{F}}(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n)\,. \tag{1.11}$$

As an illustrative example, we note that the Lax–Friedrichs scheme, for the case of the scalar advection equation with speed λ, can be written as a (consistent) conservative finite-volume scheme with a numerical flux function given by
$$\mathbf{F}_{j+1/2} = \mathcal{F}(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n) = \frac{1}{2}\left[\mathbf{F}(\mathbf{U}_j^n) + \mathbf{F}(\mathbf{U}_{j+1}^n)\right] - \frac{\Delta x}{2\Delta t}(\mathbf{U}_{j+1}^n - \mathbf{U}_j^n)\,, \tag{1.12}$$
where $\mathbf{U}_j^n = u_j^n$ and $\mathbf{F}(\mathbf{U}_j^n) = \lambda u_j^n$. Similarly, the Lax–Wendroff scheme for the scalar advection equation can be re-interpreted as a finite-volume scheme with a numerical flux function
$$\mathbf{F}_{j+1/2} = \mathcal{F}(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n) = \mathbf{F}(\boldsymbol{U}_{\text{LW}})\,, \tag{1.13}$$
where
$$\boldsymbol{U}_{\text{LW}} = \frac{1}{2}(\mathbf{U}_j^n + \mathbf{U}_{j+1}^n) - \frac{\Delta t}{2\Delta x}[\mathbf{F}(\mathbf{U}_{j+1}^n) - \mathbf{F}(\mathbf{U}_j^n)]\,. \tag{1.14}$$
Since conservative numerical schemes rely on the conservative formulation of the equations, serious problems may arise when such a conservative formulation is not available and in particular for those solutions admitting discontinuous waves.

1.5 The importance of conservative formulations

A solution to the Euler equations admitting shocks and with derivatives that are not defined at the shock, would not satisfy the hyperbolic partial differential equation in conservation form (2.2). For this reason, an alternative *integral* formulation can be adopted in order to rewrite a differential equation in a form that would admit also non-smooth "solutions". The basic idea is that of multiplying the partial differential equation (2.2) by a smooth test function ϕ with compact support in space and of integrating it by parts to move the derivatives from the function $\mathbf{U}$ over to the test function ϕ. If we consider the simpler case of one spatial dimension only, we can write
$$\partial_t \mathbf{U} + \partial_x \mathbf{F} = 0\,, \tag{1.15}$$
so that if we multiply it by a continuously differentiable function $\phi(x,t)$ of compact support, the integration over space, with $x \in (-\infty,\infty)$, and time, with $t \in [0,\infty)$, will lead to
$$\int_0^\infty \int_{-\infty}^{+\infty} [\phi\,\partial_t \mathbf{U} + \phi\,\partial_x \mathbf{F}]dx\,dt = 0\,. \tag{1.16}$$
Integration by parts both in time and in space then leads to
$$\int_{-\infty}^{+\infty} \phi\,\mathbf{U} dx\Big|_{t=0}^{t=\infty} + \int_0^\infty \phi\,\mathbf{F} dt\Big|_{x=-\infty}^{x=+\infty} - \int_0^\infty \int_{-\infty}^{+\infty} \left[\mathbf{U}\,\partial_t\phi\, + \mathbf{F}\,\partial_x\phi\right]\,dx\,dt = 0\,, \tag{1.17}$$
and thus to
$$\int_0^\infty \int_{-\infty}^{+\infty} \left[\mathbf{U}\,\partial_t\phi\, + \mathbf{F}\,\partial_x\phi\right]\,dx\,dt = -\int_{-\infty}^{+\infty} \phi(x,0)\,\mathbf{U}(x,0)\,dx\,, \tag{1.18}$$
where we have used the property that ϕ has compact support and therefore
$$\phi(x,t=\infty) = 0 = \phi(x=-\infty,t) = \phi(x=+\infty,t)\,. \tag{1.19}$$

A function $\mathbf{U}$ is then called a weak solution of the conservative equation (1.15) if it satisfies the so-called weak formulation (1.18) for all functions ϕ. As a result, also solutions involving discontinuities can be taken as weak solutions of the conservation equation.

The importance of a conservative formulation for the numerical solution of hyperbolic PDEs is underlined by the following two theorems:

> **Theorem** *Conservative numerical schemes, if convergent, converge to the weak solution of the problem* [39].

> **Theorem** *Non-conservative schemes do not converge to the correct solution if a shock wave is present in the flow* [31].

In other words, the two theorems above state that if a conservative formulation *is used*, then we are guaranteed that the numerical solution will converge to the correct one, while if a conservative formulation *is not used*, we are guaranteed to converge to the incorrect solution in the likely event in which the flow develops a discontinuity.

A well-known example of the importance of a conservative formulation is provided by the Burgers equation whose non-conservative representation

$$\partial_t u + u\,\partial_x u = 0 \tag{1.20}$$

fails dramatically in providing the correct propagation velocity of a shock wave that might form during the evolution [40]. On the other hand, the solution of the conservative formulation of the Burgers equation (1.20), i.e.

$$\partial_t u + \partial_x \left(\frac{1}{2}u^2\right) = 0\,, \tag{1.21}$$

leads to the correct propagation velocity with essentially all numerical schemes evolving smooth initial data.

1.6 Godunov methods

Godunov's original idea

Consider the finite-volume representation (1.8) of the conservation equation (2.2). The ingenious observation of Godunov was that at each interface between adjacent numerical cells the quantity $\mathbf{U}_j^n$, which is the volume average of $\mathbf{U}(x,t)$ over the cell $[x_{j-1/2}, x_{j+1/2}]$ at time $t = t^n$, manifests a *jump*, thus generating the "left" and "right" states of local Riemann problems. As a result, we can define as a Godunov method any method in which the fluxes $\mathbf{F}_{j-1/2}$ and $\mathbf{F}_{j+1/2}$ in Eq. (1.8) are calculated by solving a local Riemann problem at $x_{j\pm1/2}$. Because this logic can be extended to any other cell, a Godunov method consists in the solution of a *sequence* of local Riemann problems [27, 55].

In its *original* form, Godunov's approach used as left and right states the piecewise-constant distribution of data given by (1.6), thus building local Riemann problems, for systems of hyperbolic PDEs, with initial conditions given by

$$\mathbf{U}(x,0) = \begin{cases} \mathbf{U}_j^n & \text{if} \quad x < x_{j+1/2}\,, \\ \\ \mathbf{U}_{j+1}^n & \text{if} \quad x > x_{j+1/2}\,. \end{cases} \tag{1.22}$$

The formal solution of the local Riemann problem will then provide the terms $\mathbf{U}(x_{j\pm1/2}, t)$, which can be used in Eq. (1.64) to calculate the fluxes $\mathbf{F}_{j\pm1/2}$. In particular, because the solution of a Riemann problem is self-similar (indeed, it consists either of a set of piecewise-constant values or of a set of rarefaction waves, which are themselves self-similar) and therefore constant along the line $x/t = 0$ emerging from the points $(x_{j\pm1/2}, t^n)$, the two states $\mathbf{U}(x_{j\pm1/2}, t)$ are constant along the line $x/t = 0$. As a result, the time integrals on the right-hand side of (1.64) can be computed analytically as soon as the values $\mathbf{U}(x_{j\pm1/2}, t)$ are known.

A schematic representation of the original Godunov method as a sequence of local Riemann problems is offered in Fig. 1.1, where the continuous solution (blue solid line) is discretised as a constant value corresponding to the

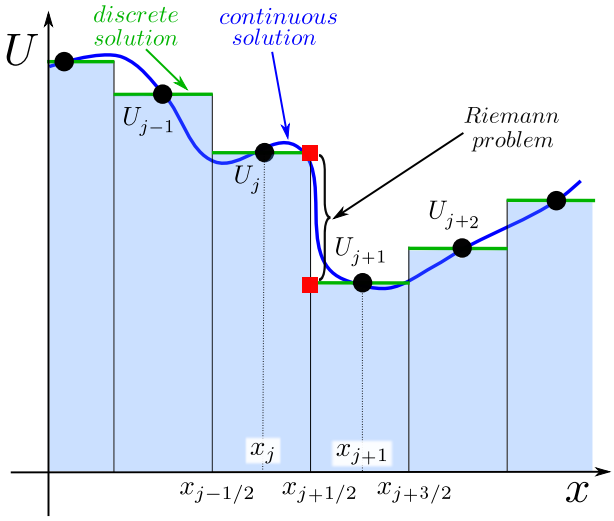

Figure 1.1: Schematic representation of local Riemann problems appearing at each cell interface in the original first-order Godunov method. Note that the continuous solution (blue solid line) is discretised as a constant value (green solid line) corresponding to the volume-average in the cell at x_j. As a result, a series of piecewise-constant states is produced, which can be seen as a sequence of (small) elementary Riemann problems. Figure taken from [50].

volume-average in the cell at x_j and leading to a series of piecewise-constant states (green solid lines). The latter can be viewed as a sequence of (small) elementary Riemann problems as the one at $x_{j+1/2}$ between the states U_j and U_{j+1}. The evolution of the Riemann problems depicted in Fig. 1.1 from t^n to t^{n+1} is shown schematically in Fig. 1.2, where we report the characteristic waves originating from each cell interface. These waves represent a local Riemann fan and comprise shock waves (blue solid lines), rarefaction waves (red solid lines) and contact discontinuities (green dotted lines) whose direction of propagation will be dictated by the initial conditions of the local Riemann problem.

Note that, because the time-step is constrained by the CFL condition, it is possible that the characteristics emerging at time t^n from a given interface intersect with those emerging from the adjacent interface (this is indeed shown in Fig. 1.2 for the characteristic emanating from $x_{j+1/2}$ and $x_{j+3/2}$). However, as long as the new characteristics produced at the intersection point at a time between t^n and t^{n+1} are not faster than the original ones generated at time t^n (i.e. there is no "wave acceleration"), the perturbation due to the wave interaction will have no time to reach the interface, leaving the computation of the numerical fluxes unaffected. Indeed, the condition that $c_{\rm CFL}$ needs to be chosen smaller than 1 in Godunov methods reflects the constraint on the choice of t^{n+1} that accounts for a possible wave acceleration.

A number of remarks can be made about the original Godunov scheme (1.8) with a piecewise-constant distribution of the data:

1. The scheme is *conservative* since it adopts the form (1.8) with a numerical flux function

$$\mathbf{F}_{j+1/2} = \frac{1}{\Delta t}\int_{t^n}^{t^{n+1}} \mathbf{F}[\mathbf{U}(x_{j+1/2},t)]dt = \boldsymbol{\mathcal{F}}(\mathbf{U}_j^n, \mathbf{U}_{j+1}^n)\,, \tag{1.23}$$

   in which $\boldsymbol{\mathcal{F}}$ is in the form prescribed by Eq. (1.11) because the solution of the Riemann problem only depends on the two (constant) states $\mathbf{U}_j$ and $\mathbf{U}_{j+1}$.

2. The scheme is *upwind* since it adopts the solution of the Riemann problem for the computation of the fluxes $\mathbf{F}_{j\pm1/2}$. The proof is indeed straightforward if we apply the scheme (1.8) to the linear advection equation

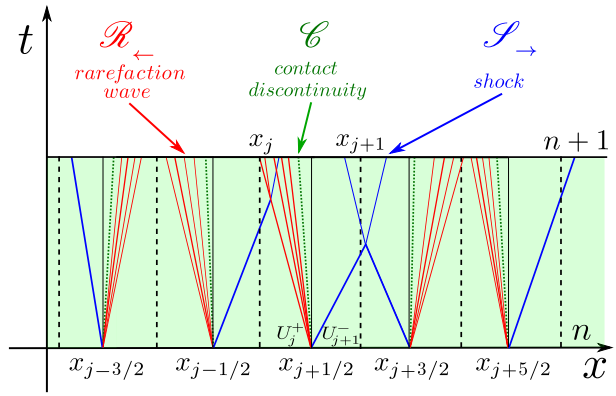

Figure 1.2: Schematic spacetime representation of the characteristic waves originating from each cell interface as a result of the evolution from t^n to t^{n+1} of the sequence of Riemann problems in the first-order Godunov scheme of Fig. 1.1. Figure taken from [50].

with the flux given by $\mathbf{F} = \lambda \mathbf{U}$. In this case, the solution of each local Riemann problem at the generic cell interface $x_{j+1/2}$ is given by $\mathbf{U}_{j+1}^n$, if $\lambda < 0$, and by $\mathbf{U}_j^n$, if $\lambda > 0$. Therefore, the resulting scheme is given by

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n - \alpha(\mathbf{U}_j^n - \mathbf{U}_{j-1}^n)\,, \qquad \text{if} \quad \lambda > 0\,, \tag{1.24}$$

and

$$\mathbf{U}_j^{n+1} = \mathbf{U}_j^n - \alpha(\mathbf{U}_{j+1}^n - \mathbf{U}_j^n)\,, \qquad \text{if} \quad \lambda < 0\,, \tag{1.25}$$

where $\alpha = \lambda \Delta t/\Delta x$. The schemes (1.24) and (1.25) are nothing but the upwind methods first introduced by [17]. For systems of nonlinear equations, the upwind property is guaranteed by the fact that the exact Riemann solver provides the full wave structure at each numerical interface, and therefore the resulting numerical flux respects the direction from which perturbations propagate.

3. The scheme is *monotone* and therefore non-oscillatory (easy to prove for a scalar advection equation).

4. The scheme is just *first-order accurate* in time and in space as a consequence of Godunov's theorem and as evident from Eqs. (1.24)–(1.25), which are indeed first-order schemes.

5. The scheme is *time-constrained* by the CFL condition even when applied to a system of N nonlinear equations. In other words, the time-step between n and $n+1$ must be such that $\Delta t = c_{\text{CFL}} \min_k (\Delta x/|\lambda_k^n|)$, where $|\lambda_k^n|$ is the propagation velocity at time $t = t^n$, while the index $1 \le k \le N$ spans the set of possible different eigenvalues of the system.

From these considerations it emerges that the original Godunov method had the virtue of combining the least-accurate strategy for the calculation of the left and right states of the local Riemann problem (i.e the use of piecewise-constant data), with the most accurate strategy for the solution of such Riemann problems (i.e. the use of an exact Riemann solver).

Modern Godunov methods

If we want to go beyond the first order accurate Godunov method, we must face the limitations imposed by the following theorem, also due to Godunov

Theorem *A linear (i.e. with constant coefficients) and monotonicity-preserving scheme is at most first-order accurate.*

In practice, high-order linear schemes and the absence of oscillations are two incompatible requirements. Starting in the early 1970s, a great deal of effort has been dedicated to circumvent the strong limitations imposed by this theorem. Indeed, it was soon found that the numerical scheme must be made nonlinear. Hence, the spatial accuracy of the original first-order Godunov method can be improved if the left and right states of the Riemann problem solved at the interface $x_{j+1/2}$ are not simply given by the constant values of $\mathbf{U}_j$ and $\mathbf{U}_{j+1}$, but are rather obtained after *reconstructing* a higher-order polynomial representation of $\mathbf{U}$ within each cell, starting from the averages $\mathbf{U}_j$ of $\mathbf{U}$ in a prescribed number of neighbouring cells.

Modern Godunov methods, unlike the original one, combine sophisticated strategies for the computation of the left and right states of the local Riemann problems with approximate solutions of such Riemann problems. The resulting methods have become so widely adopted that they are collectively known as *high-resolution shock-capturing (HRSC)* methods *tout-court*, which, broadly speaking, share the following features

- at least second order of accuracy on smooth parts of the solution;
- sharp resolution of discontinuities without large smearing;
- absence of spurious oscillations in the solution;
- convergence to the "true" solution as the grid is refined;
- no use of artificial-viscosity terms.

1.7 TVD reconstruction

The first implementation of non-linear conservative numerical schemes is obtained through so called Total Variation Diminishing methods. These methods improve the simple piecewise-constant representation adopted in the original Godunov method by providing a piecewise-linear reconstruction of $\mathbf{U}_j(x)$ inside each cell, i.e.by prescribing $\mathbf{U}_j^n(x)$ as

$$\mathbf{U}_j^n(x) = \mathbf{U}_j^n + \boldsymbol{\sigma}_j^n(x - x_j) \qquad \text{with } x_{j-1/2} \le x \le x_{x+j/2}\,, \tag{1.26}$$

where $x_j = (x_{j-1/2} + x_{j+1/2})/2$ is the coordinate value at the cell centre, while $\boldsymbol{\sigma}_j^n$ is the *"slope"* of the linear reconstruction inside the cell. Several choices are possible for the slope and the most commonly adopted TVD slope limiters are listed below.

- The *minmod slope limiter* [36, 59]

$$\boldsymbol{\sigma}_j^n = \text{minmod}\left(\frac{\mathbf{U}_j^n - \mathbf{U}_{j-1}^n}{\Delta x}, \frac{\mathbf{U}_{j+1}^n - \mathbf{U}_j^n}{\Delta x}\right)\,, \tag{1.27}$$

  where

$$\begin{aligned}\text{minmod}(\alpha,\beta) &= \begin{cases} \alpha & \text{if} \quad |\alpha| < |\beta| \text{ and} \quad \alpha\beta > 0\,, \\ \beta & \text{if} \quad |\beta| < |\alpha| \text{ and} \quad \alpha\beta > 0\,, \\ 0 & \text{if} \quad \alpha\beta \le 0\,, \end{cases} \\ &= \text{sign}(\alpha)\max\{0, \min\{|\alpha|, \beta\,\text{sign}(\alpha)\}\}\,. \end{aligned} \tag{1.28}$$

- The *monotonised central-difference limiter* (MC) [58]

$$\boldsymbol{\sigma}_j^n = \text{minmod}\left(\frac{\mathbf{U}_{j+1}^n - \mathbf{U}_{j-1}^n}{2\Delta x}, 2\frac{\mathbf{U}_j^n - \mathbf{U}_{j-1}^n}{\Delta x}, 2\frac{\mathbf{U}_{j+1}^n - \mathbf{U}_j^n}{\Delta x}\right)\,. \tag{1.29}$$

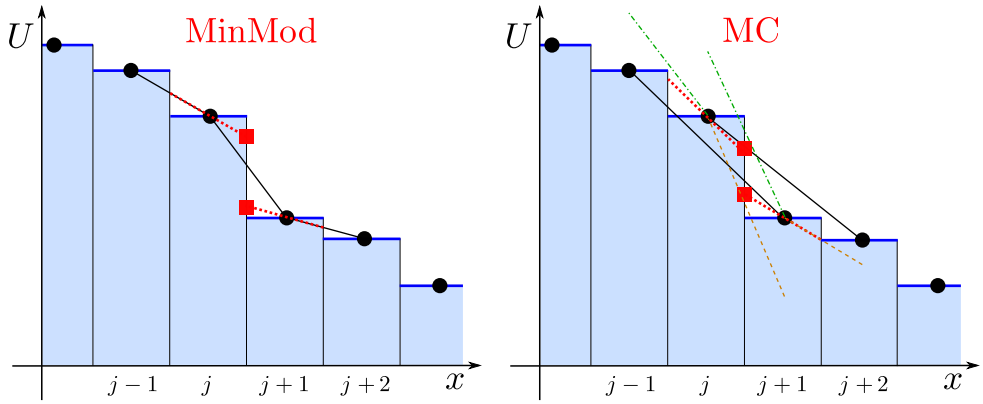

Figure 1.3: Schematic representation and comparison between two slope limiters: the minmod limiter (left panel) and the MC limiter (right panel), when acting on the same set of data. Figure taken from [50].

where

$$\text{minmod}(\alpha,\beta,\gamma) \quad = \quad \begin{cases} \min(\alpha,\beta,\gamma) & \text{if} \quad \alpha,\beta,\gamma > 0\,, \\ \max(\alpha,\beta,\gamma) & \text{if} \quad \alpha,\beta,\gamma < 0\,, \\ 0 & \text{otherwise}\,. \end{cases} \tag{1.30}$$

Figure 1.3 shows the comparison between the slopes provided by the minmod and the MC limiters. Note that minmod always takes the least steep among the slopes provided by the backward and the forward finite-difference. The MC limiter, on the other hand, compares three different slopes and it is slightly more involved. To fix ideas, let us consider the graphical representation of the MC algorithm when applied to the cell $j+1$, as shown in the right panel of Fig. 1.3. The first slope that is considered is that among the values at x_j and x_{j+2}, reported with a black solid line; the second one has twice the slope of the straight line among the values at x_j and x_{j+1}, and is reported with the green dash-dotted line originating from the value at x_{j+1}; finally, the third one has twice the slope of the straight line among the values at x_{j+1} and x_{j+2}, and is reported with the orange short-dashed line originating from the value at x_{j+1}. The resulting slope provided by the MC algorithm is the least steep among the above three, and is reported with the red dashed line inside the cell $j+1$. This procedure is then repeated for each cell. Once $\mathbf{U}_j(x)$ has been reconstructed, the values of $\mathbf{U}_j(x)$ at the two extremes of each cell, also called the *boundary-extrapolated values*,

$$\mathbf{U}_j^- = \mathbf{U}_j(x_{j-1/2})\,, \qquad\qquad \mathbf{U}_j^+ = \mathbf{U}_j(x_{j+1/2})\,, \tag{1.31}$$

provide the left and right constant states of the Riemann problem that has to be solved at every cell interface. For instance, at $x_{j+1/2}$ the local Riemann problem is defined as

$$\mathbf{U}(x,0) = \begin{cases} \mathbf{U}_j^+ & \text{if} \quad x < x_{j+1/2}\,, \\ \mathbf{U}_{j+1}^- & \text{if} \quad x > x_{j+1/2}\,. \end{cases} \tag{1.32}$$

This configuration is represented schematically in Fig. 1.4, where $\mathbf{U}_j^+$ and $\mathbf{U}_{j+1}^-$ are the left and right states of the Riemann problem to be solved at $x_{j+1/2}$.

In spite of their success in modelling sharp discontinuities without producing oscillations, TVD slope-limiter methods suffer from excessive numerical dissipation near local extrema, where the reconstructed profile of the TVD method is a straight line with zero slope, thus reducing to the first-order Godunov method.

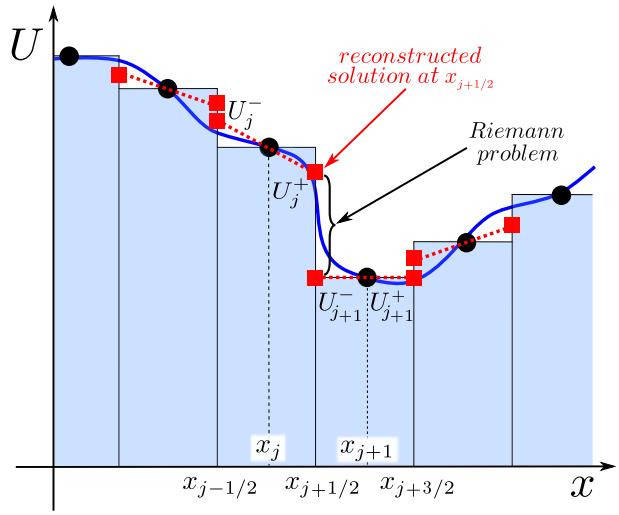

Figure 1.4: Schematic representation of the boundary-extrapolated values (red solid boxes) providing the left and right states of the Riemann problem at each cell interface. The reconstruction is made using the minmod limiter and the red dashed lines represent the reconstructed final slopes. Figure taken from [50].

1.8 WENO reconstruction

When implementing a numerical scheme of order higher than the second, the TVD condition based on slope-limiter reconstructions turns out to be too restrictive. It produces excessive numerical dissipation in regions of smooth flow and reduces the order of accuracy to first-order near local extrema. These restrictions have led to looser criteria that allow for a small increase of the total variation near extremal points while suppressing oscillations when required. *Essentially non-oscillatory* (ENO) and *weighted essentially non-oscillatory* (WENO) methods have been devised precisely to meet these criteria.

Both ENO and WENO methods use the idea of adaptive stencils when performing the reconstruction of the numerical solution to achieve high-order accuracy and the non-oscillatory property near discontinuities. However, while the ENO method uses just one out of many candidate stencils for the reconstruction and in particular the "smoothest" one, the WENO method uses a special combination of all the candidate stencils, with each being assigned a nonlinear weight depending on the local smoothness of the numerical solution on that stencil. The weights are then adjusted by the local smoothness of the solution, so that essentially zero weights are given to non-smooth stencils while optimal weights are prescribed in smooth regions.

1.8.1 Point-wise WENO

The original pointwise WENO reconstruction of Jiang and Shu [34] produces a high order accurate pointwise reconstruction of the solution at the element interfaces $x_{i\pm 1/2}$. It is however rather difficult to generalize to unstructured meshes in two and three space dimensions, since it requires the determination of the optimal linear weights. We will not discuss it further in these lectures.

1.8.2 Entire polynomial WENO

A valuable alternative to the pointwise WENO reconstruction of Jiang and Shu [34] has been proposed by [21, 24] for unstructured triangular and tetrahedral meshes and it produces an *entire polynomial* reconstruction. In practice, we first introduce spacetime reference coordinates $\xi, \eta, \zeta, \tau \in [0, 1]$, which are defined by

$$x = x_{i-\frac{1}{2}} + \xi \Delta x_i, \quad y = y_{j-\frac{1}{2}} + \eta \Delta y_j, \quad z = z_{k-\frac{1}{2}} + \zeta \Delta z_k, \quad t = t^n + \tau \Delta t\,. \tag{1.33}$$

After that, focusing on the x direction for convenience, we choose the degree M of the polynomial approximating the solution, and an orthogonal basis of polynomials [52], all of degree M, rescaled on the same unit interval $[0, 1]$. The basis is built in the following way

- Compute the $M+1$ Gauss-Legendre quadrature nodes $\{\xi_k\}_{k=1}^{M+1}$ as the zeroes of the Legendre polynomials of order $M+1$.
- Compute the $M+1$ Lagrange interpolation polynomials, $\{\psi_l(\xi)\}_{l=1}^{M+1}$, passing through the $M+1$ Gauss-Legendre quadrature nodes $\{\xi_k\}_{k=1}^{M+1}$

$$\psi_l(\xi) = \prod_{n=1, n\neq l}^{M+1} \frac{\xi - \xi_n}{\xi_l - \xi_n} \tag{1.34}$$

- Note that

$$\psi_l(\xi_k) = \delta_{lk} \qquad l, k = 1, 2, \ldots, M+1\,, \tag{1.35}$$

  where δ_{lk} is the "Kronecker delta", i.e. $\delta_{lk} = 1$ if $l = k$, $\delta_{lk} = 0$ otherwise. In the programming practice, the computation of the $M+1$ Lagrange interpolation polynomials is not performed through the definitions (1.34) but rather via solution of $M+1$ linear systems of the kind (example with $M = 2$)

$$\begin{pmatrix} 1 & \xi_1 & \xi_1^2 \\ 1 & \xi_2 & \xi_2^2 \\ 1 & \xi_3 & \xi_3^2 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \tag{1.36}$$

$$\begin{pmatrix} 1 & \xi_1 & \xi_1^2 \\ 1 & \xi_2 & \xi_2^2 \\ 1 & \xi_3 & \xi_3^2 \end{pmatrix} \begin{pmatrix} a' \\ b' \\ c' \end{pmatrix} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \tag{1.37}$$

$$\begin{pmatrix} 1 & \xi_1 & \xi_1^2 \\ 1 & \xi_2 & \xi_2^2 \\ 1 & \xi_3 & \xi_3^2 \end{pmatrix} \begin{pmatrix} a'' \\ b'' \\ c'' \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \tag{1.38}$$

- We also recall that, having selected the nodal points in this way, we will compute integrals through Gaussian quadrature rules

$$\int_0^1 g(\xi) d\xi \approx \sum_{k=1}^{M+1} \omega_k g(\xi_k) \tag{1.39}$$

  and this integral is known to be exact for all polynomials up to degree $2\underbrace{(M+1)}_{\text{nodal points}} - 1 = 2M + 1$.

Having done that, a (small) number of one-dimensional reconstruction stencils is adopted, each of them formed by the union of $M+1$ adjacent cells, i.e.

$$\mathcal{S}_{ijk}^{s,x} = \bigcup_{e=i-L}^{i+R} I_{ejk}, \quad \mathcal{S}_{ijk}^{s,y} = \bigcup_{e=j-L}^{j+R} I_{iek}, \quad \mathcal{S}_{ijk}^{s,z} = \bigcup_{e=k-L}^{k+R} I_{ije}, \tag{1.40}$$

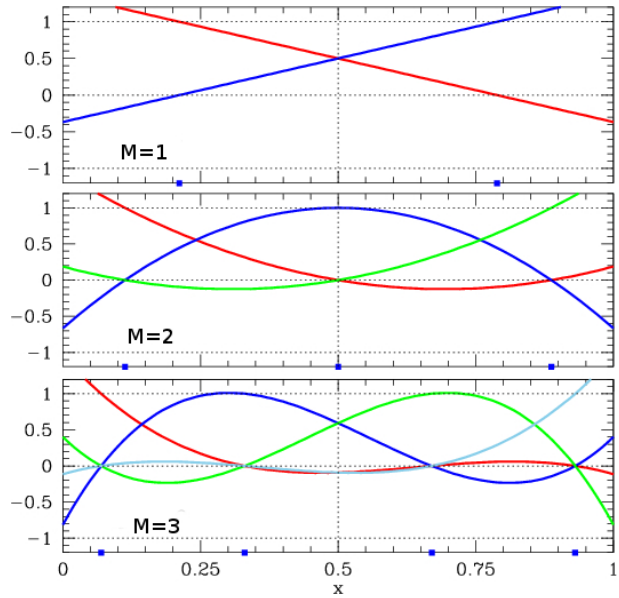

Figure 1.5: Functional form of the first polynomials of the nodal basis of degree M. The blue squares indicate the abscissas of the $M+1$ Gaussian points. Figure taken from [50].

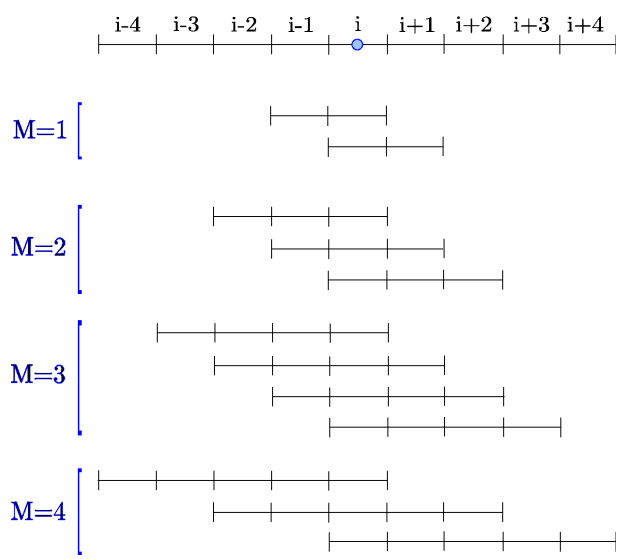

Figure 1.6: Representation of the one-dimensional stencils adopted up to $M = 4$. Odd order schemes (even polynomials of degree M) always use three stencils, while even order schemes (odd polynomials of degree M) always adopt four stencils, with the exception of the $M = 1$ case, for which there are only two stencils. Figure taken from [62].

Table 1.1: The table shows the coordinates of the Gauss–Legendre nodes and the corresponding nodal basis polynomials for a few values of M.

	ξ_k	ψ_l
$M=1$		
	$\xi_1 = 0.2113248654051$	$\psi_1 = 1.366025403784 - 1.732050807568\xi$
	$\xi_2 = 0.7886751345948$	$\psi_2 = -0.3660254037844 + 1.732050807568\xi$
$M=2$		
	$\xi_1 = 0.1127016653792$	$\psi_1 = 1.478830557701 - 4.624327782069\xi$
		$+3.333333333333\xi^2$
	$\xi_2 = 0.5$	$\psi_2 = -0.6666666666666 + 6.666666666666\xi$
		$-6.666666666666\xi^2$
	$\xi_3 = 0.8872983346207$	$\psi_3 = 0.1878361089654 - 2.042338884597\xi$
		$+3.333333333333\xi^2$
$M=3$		
	$\xi_1 = 6.9431844202973 \times 10^{-2}$	$\psi_1 = 1.526788125457 - 8.546023607872\xi$
		$+14.32585835417\xi^2 - 7.42054006803894\xi^3$
	$\xi_2 = 0.3300094782075$	$\psi_2 = -0.8136324494869 + 13.80716692568\xi$
		$-31.38822236344\xi^2 + 18.79544940755\xi^3$
	$\xi_3 = 0.6699905217924$	$\psi_3 = 0.4007615203116 - 7.417070421462\xi$
		$+24.99812585921\xi^2 - 18.79544940755\xi^3$
	$\xi_4 = 0.9305681557970$	$\psi_4 = -0.1139171962819 + 2.155927103645\xi$
		$-7.935761849944\xi^2 + 7.420540068038\xi^3$
$M=4$		
	$\xi_1 = 4.6910077030668 \times 10^{-2}$	$\psi_1 = 1.551408049094 - 13.47028450119\xi$
		$+38.64449905534\xi^2 - 44.98898505587\xi^3$
		$+18.33972111443\xi^4$
	$\xi_2 = 0.2307653449471$	$\psi_2 = -0.8931583920000 + 22.92433355572\xi$
		$-88.22281082816\xi^2 + 117.8634151266\xi^3$
		$-51.93972111443\xi^4$
	$\xi_3 = 0.5$	$\psi_3 = 0.5333333333333 - 14.93333333333\xi$
		$+82.13333333333\xi^2 - 134.4000000000\xi^3$
		$+67.20000000000\xi^4$
	$\xi_4 = 0.7692346550528$	$\psi_4 = -0.2679416522233 + 7.689927178385\xi$
		$-46.27089213480\xi^2 + 89.89546933107\xi^3$
		$-51.93972111443\xi^4$
	$\xi_5 = 0.9530899229693$	$\psi_5 = 7.635866179581 \times 10^{-2} - 2.210642899581\xi$
		$+13.71587057429\xi^2 - 28.36989940184\xi^3$
		$+18.33972111443\xi^4$

where $L = L(M, s)$ and $R = R(M, s)$ are the spatial extension of the stencil to the left and to the right. In practice, odd order schemes (even polynomials of degree M) always use three stencils ($N_s = 3$), while even order schemes (odd polynomials of degree M) always adopt four stencils ($N_s = 4$), with the exception of the $M = 1$ case, for which there are only two stencils. The coordinates of the Gaussian points, the nodal basis polynomials and the corresponding stencils for a few values of M up to $M = 4$ are reported in Tab. 1.1.

With all this machinery at hands, we use the polynomial basis functions to reconstruct the solution at time t^n as

$$\mathbf{w}_h^{s,x}(x, t^n) = \sum_{p=0}^{M} \psi_p(\xi)\hat{\mathbf{w}}_{ijk,p}^{n,s} := \psi_p(\xi)\hat{\mathbf{w}}_{ijk,p}^{n,s}\,, \tag{1.41}$$

As usual for finite volume methods, the reconstructed polynomial must preserve the cell-average of the solution over each element I_{ijk}, namely

$$\frac{1}{\Delta x_e}\int_{x_{e-\frac{1}{2}}}^{x_{e+\frac{1}{2}}} \psi_p(\xi(x))\hat{\mathbf{w}}_{ijk,p}^{n,s}\,dx = \mathbf{U}_{ejk}^n, \qquad \forall I_{ejk}\in\mathcal{S}_{ijk}^{s,x}\,, \tag{1.42}$$

which provide a system of linear equations for the unknown coefficients $\hat{\mathbf{w}}_{ijk,p}^{n,s}$. This operation is repeated for each stencil relative to the element I_{ijk}. After that, we can construct a data-dependent nonlinear combination of the polynomials computed from each stencil, i.e.

$$\mathbf{w}_h^x(x,t^n) = \psi_p(\xi)\hat{\mathbf{w}}_{ijk,p}^n, \quad \text{with} \quad \hat{\mathbf{w}}_{ijk,p}^n = \sum_{s=1}^{N_s}\omega_s\hat{\mathbf{w}}_{ijk,p}^{n,s}\,. \tag{1.43}$$

The nonlinear weights ω_s are computed following the same logic as for the optimal WENO of [35], i.e.

$$\omega_s = \frac{\tilde{\omega}_s}{\sum_k\tilde{\omega}_k}\,, \qquad \tilde{\omega}_s = \frac{\lambda_s}{(\sigma_s+\epsilon)^r}\,. \tag{1.44}$$

However, the actual values of the linear weights λ_s are not the same as those of the optimal WENO and they are chosen according to a more pragmatic approach. In fact, the weight of the central stencils is given a very large value, $\lambda_s = 10^5$, while the weight of the one-sided stencils is set to $\lambda_s = 1$. Moreover, we have used $\epsilon = 10^{-14}$ and $r = 8$. The oscillation indicator σ_s of Eq. (1.44) is

$$\sigma_s = \Sigma_{pm}\hat{\mathbf{w}}_{ijk,p}^{n,s}\hat{\mathbf{w}}_{ijk,m}^{n,s}\,, \tag{1.45}$$

and it requires the computation of the oscillation indicator matrix [20]

$$\Sigma_{pm} = \sum_{\alpha=1}^{M}\int_0^1 \frac{\partial^\alpha\psi_p(\xi)}{\partial\xi^\alpha}\cdot\frac{\partial^\alpha\psi_m(\xi)}{\partial\xi^\alpha}d\xi\,, \tag{1.46}$$

which, compared to alternative expressions proposed in the literature, has the advantage that it does not depend on the grid spacing, and is therefore "universal". We emphasize that the reconstruction polynomial $\mathbf{w}_h^x(x,t^n)$ resulting from Eq. (1.43) is still an average in the y and z directions. Hence, the procedure explained so far is repeated along the two missing directions y and z. The net effect of this approach is to provide a genuine multidimensional reconstruction, although in such a way that each direction is treated separately (dimension-by-dimension approach). Alternatively, one can operate with truly multidimensional stencils [23, 19, 7].

1.9 The local spacetime Discontinuous Galerkin predictor

The high order computation of the numerical fluxes, which involves a time integration from t^n to t^{n+1}, requires the numerical flux (Riemann solver) $\tilde{\mathbf{f}}_{\mathrm{RP}}$ to be computed with high accuracy at any time in the interval $t\in[t^n;t^{n+1}]$. To this extent, we need an operation, to be performed locally for each cell, which uses as input the high order polynomial $\mathbf{w}_h$ obtained from the WENO reconstruction, and gives as output its evolution in time, namely

$$\mathbf{w}_h(x,y,z,t^n) \longrightarrow \mathbf{q}_h(x,y,z,t)\,. \tag{1.47}$$

Unlike the original ADER approach, where this operation was obtained through the so called Cauchy-Kowalevski procedure [56, 57, 22], in the version of ADER presented in these lectures the transformation represented by (1.47) is obtained through an element–local space–time Discontinuous Galerkin predictor that is based on the *weak* integral form of Eq. (2.2) [20]. The basic idea can be summarized as follows. The sought polynomial $\mathbf{q}_h(x,y,z,t)$ is supposed to be expanded in space and time as

$$\mathbf{q}_h = \mathbf{q}_h(\boldsymbol{\xi},\tau) = \theta_{\mathfrak{p}}\left(\boldsymbol{\xi},\tau\right)\hat{\mathbf{q}}_{\mathfrak{p}}\,, \tag{1.48}$$

where the polynomial basis functions $\theta_{\mathfrak{p}}$ are given by a dyadic–product of the basis functions ψ_l already used for the WENO reconstruction, namely

$$\theta_{\mathfrak{p}}(\boldsymbol{\xi},\tau) = \psi_p(\xi)\psi_q(\eta)\psi_r(\zeta)\psi_s(\tau)\,. \tag{1.49}$$

The terms $\hat{\mathbf{q}}_{\mathfrak{p}}$ are the so-called degrees of freedom and they are the unknowns of the problem. After multiplying the governing PDE

$$\partial_t \mathbf{U} + \partial_x \mathbf{f} + \partial_y \mathbf{g} + \partial_z \mathbf{h} = \mathbf{S}\,, \tag{1.50}$$

written in the reference coordinates (ξ,η,ζ,τ), with the space–time test functions $\theta_{\mathfrak{q}}$ and integrating over the space–time reference control volume, we obtain[1]

$$\int_0^1\int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}\left(\frac{\partial \mathbf{U}}{\partial\tau} + \frac{\partial \mathbf{f}^*}{\partial\xi} + \frac{\partial \mathbf{g}^*}{\partial\eta} + \frac{\partial \mathbf{h}^*}{\partial\zeta} - \mathbf{S}^*\right) d\xi d\eta d\zeta d\tau = 0. \tag{1.52}$$

The key aspect of the whole strategy is to perform an integration by parts in time, while keeping the treatment local in space. After doing so, we get

$$\begin{aligned}
&\int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}(\boldsymbol{\xi},1)\mathbf{U}(\boldsymbol{\xi},1)d\xi d\eta d\zeta - \int_0^1\int_0^1\int_0^1\int_0^1 \left(\frac{\partial}{\partial\tau}\theta_{\mathfrak{q}}\right)\mathbf{U} d\xi d\eta d\zeta d\tau \\
&+\int_0^1\int_0^1\int_0^1\int_0^1 \left[\theta_{\mathfrak{q}}\left(\frac{\partial \mathbf{f}^*}{\partial\xi} + \frac{\partial \mathbf{g}^*}{\partial\eta} + \frac{\partial \mathbf{h}^*}{\partial\zeta}\right)\right] d\xi d\eta d\zeta d\tau = \int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}(\boldsymbol{\xi},0)\underbrace{\mathbf{U}(\boldsymbol{\xi},0)}_{\mathbf{w}_h=\psi_p\hat{\mathbf{w}}^n_{ijk,p},} d\xi d\eta d\zeta + \\
&\int_0^1\int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}\mathbf{S}^* d\xi d\eta d\zeta d\tau\,.
\end{aligned} \tag{1.53}$$

In the first two integrands of Eq. (1.53) we can perform the replacement $\mathbf{U} \to \mathbf{q}_h$, since $\mathbf{q}_h$ is the discrete space-time solution we are looking for. In the integrand on the right hand side of Eq. (1.53), on the other hand, we can perform the replacement $\mathbf{U}(\boldsymbol{\xi},t^n) \to \mathbf{w}_h(\boldsymbol{\xi},t^n)$, since at time t^n the solution is known, and it is represented by the reconstructed polynomial computed according to the WENO reconstruction. In addition to this, we assume that the fluxes and the sources can also be expanded over the basis as we did in Eq. (1.48), namely

$$\mathbf{f}^* = \theta_{\mathfrak{p}}\hat{\mathbf{f}}^*_{\mathfrak{p}}, \qquad \mathbf{g}^* = \theta_{\mathfrak{p}}\hat{\mathbf{g}}^*_{\mathfrak{p}}, \qquad \mathbf{h}^* = \theta_{\mathfrak{p}}\hat{\mathbf{h}}^*_{\mathfrak{p}}\,. \qquad \mathbf{S}^* = \theta_{\mathfrak{p}}\hat{\mathbf{S}}^*_{\mathfrak{p}}\,. \tag{1.54}$$

From the computational point of view, the advantage of the nodal basis becomes apparent at this stage. In fact, the above degrees of freedom for the fluxes are simply the point–wise evaluation of the physical fluxes, hence

$$\hat{\mathbf{f}}^*_{\mathfrak{p}} = \mathbf{f}^*\left(\hat{\mathbf{q}}_{\mathfrak{p}}\right), \qquad \hat{\mathbf{g}}^*_{\mathfrak{p}} = \mathbf{g}^*\left(\hat{\mathbf{q}}_{\mathfrak{p}}\right), \qquad \hat{\mathbf{h}}^*_{\mathfrak{p}} = \mathbf{h}^*\left(\hat{\mathbf{q}}_{\mathfrak{p}}\right). \qquad \hat{\mathbf{S}}^*_{\mathfrak{p}} = \mathbf{S}^*\left(\hat{\mathbf{q}}_{\mathfrak{p}}\right). \tag{1.55}$$

Inserting Eqns. (1.48) and (1.54) into (1.53) yields

$$\begin{aligned}
&\int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}(\boldsymbol{\xi},1)\theta_{\mathfrak{p}}(\boldsymbol{\xi},1)\hat{\mathbf{q}}_{\mathfrak{p}}\, d\xi d\eta d\zeta - \int_0^1\int_0^1\int_0^1\int_0^1 \left(\frac{\partial}{\partial\tau}\theta_{\mathfrak{q}}\right)\theta_{\mathfrak{p}}\hat{\mathbf{q}}_{\mathfrak{p}}\, d\xi d\eta d\zeta d\tau \\
&+\int_0^1\int_0^1\int_0^1\int_0^1 \left[\theta_{\mathfrak{q}}\left(\frac{\partial}{\partial\xi}\theta_{\mathfrak{p}}\hat{\mathbf{f}}^*_{\mathfrak{p}} + \frac{\partial}{\partial\eta}\theta_{\mathfrak{p}}\hat{\mathbf{g}}^*_{\mathfrak{p}} + \frac{\partial}{\partial\zeta}\theta_{\mathfrak{p}}\hat{\mathbf{h}}^*_{\mathfrak{p}}\right)\right] d\xi d\eta d\zeta d\tau \\
&= \int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}(\boldsymbol{\xi},0)\mathbf{w}_h(\boldsymbol{\xi},t^n)\, d\xi d\eta d\zeta + \int_0^1\int_0^1\int_0^1\int_0^1 \theta_{\mathfrak{q}}\theta_{\mathfrak{p}}\hat{\mathbf{S}}^*_{\mathfrak{p}} d\xi d\eta d\zeta d\tau\,.
\end{aligned} \tag{1.56}$$

[1] Here we have defined

$$\mathbf{f}^* = \frac{\Delta t}{\Delta x_i}\,\mathbf{f}, \quad \mathbf{g}^* = \frac{\Delta t}{\Delta y_j}\,\mathbf{g}, \quad \mathbf{h}^* = \frac{\Delta t}{\Delta z_k}\,\mathbf{h}\,. \quad \mathbf{S}^* = \Delta t\,\mathbf{S}\,. \tag{1.51}$$

It may be noticed that Eq. (1.56) contains several integrals which only involve the basis functions and their derivatives, and which can be pre-computed in the code. Hence, after defining the integrals

$$\mathbf{K}^1_{\mathfrak{q}\mathfrak{p}} = \int_0^1\int_0^1\int_0^1 \theta_\mathfrak{q}(\boldsymbol{\xi},1)\theta_\mathfrak{p}(\boldsymbol{\xi},1)d\boldsymbol{\xi} - \int_0^1\int_0^1\int_0^1\int_0^1 \left(\frac{\partial}{\partial\tau}\theta_\mathfrak{q}\right)\theta_\mathfrak{p} d\boldsymbol{\xi}d\tau\,, \tag{1.57}$$

$$\mathbf{K}^{\boldsymbol{\xi}}_{\mathfrak{q}\mathfrak{p}} = \left(\mathbf{K}^\xi_{\mathfrak{q}\mathfrak{p}},\mathbf{K}^\eta_{\mathfrak{q}\mathfrak{p}},\mathbf{K}^\zeta_{\mathfrak{q}\mathfrak{p}}\right) = \int_0^1\int_0^1\int_0^1\int_0^1 \theta_\mathfrak{q}\frac{\partial}{\partial\boldsymbol{\xi}}\theta_\mathfrak{p} d\boldsymbol{\xi}d\tau\,, \tag{1.58}$$

$$\mathbf{F}^0_{\mathfrak{q}\mathfrak{p}} = \int_0^1\int_0^1\int_0^1 \theta_\mathfrak{q}(\boldsymbol{\xi},0)\psi_\mathfrak{p}(\boldsymbol{\xi})d\boldsymbol{\xi}\,, \tag{1.59}$$

$$\mathbf{M}_{\mathfrak{q}\mathfrak{p}} = \int_0^1\int_0^1\int_0^1\int_0^1 \theta_\mathfrak{q}\theta_\mathfrak{p} d\boldsymbol{\xi}d\tau\,. \tag{1.60}$$

where $d\boldsymbol{\xi} = d\xi d\eta d\zeta$, we can rewrite the system (1.56) in a compact form as an algebraic equation system for the unknown coefficients $\hat{\mathbf{q}}_\mathfrak{p}$, i.e.

$$\mathbf{K}^1_{\mathfrak{q}\mathfrak{p}}\hat{\mathbf{q}}_\mathfrak{p} + \mathbf{K}^\xi_{\mathfrak{q}\mathfrak{p}}\cdot\hat{\mathbf{f}}^*_\mathfrak{p} + \mathbf{K}^\eta_{\mathfrak{q}\mathfrak{p}}\hat{\mathbf{g}}^*_\mathfrak{p} + \mathbf{K}^\zeta_{\mathfrak{q}\mathfrak{p}}\hat{\mathbf{h}}^*_\mathfrak{p} = \mathbf{F}^0_{\mathfrak{q}\mathfrak{m}}\hat{\mathbf{w}}^n_\mathfrak{m} + \mathbf{M}_{\mathfrak{q}\mathfrak{p}}\hat{\mathbf{S}}^*_\mathfrak{p}\,. \tag{1.61}$$

This system of equations must be solved approximately through standard iterative procedures up to a desired tolerance, i.e.

For non-stiff sources:

$$\hat{\mathbf{q}}^{i+1}_\mathfrak{p} = (\mathbf{K}^1_{\mathfrak{q}\mathfrak{p}})^{-1}\left[-\mathbf{K}^\xi_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{f}}^*_\mathfrak{p})^i - \mathbf{K}^\eta_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{g}}^*_\mathfrak{p})^i - \mathbf{K}^\zeta_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{h}}^*_\mathfrak{p})^i + \mathbf{F}^0_{\mathfrak{q}\mathfrak{m}}\hat{\mathbf{w}}^n_\mathfrak{m} + \mathbf{M}_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{S}}^*_\mathfrak{p})^i\,.\right] \tag{1.62}$$

For stiff sources:

$$\hat{\mathbf{q}}^{i+1}_\mathfrak{p} - (\mathbf{K}^1_{\mathfrak{q}\mathfrak{p}})^{-1}\mathbf{M}_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{S}}^*_\mathfrak{p})^{i+1} = (\mathbf{K}^1_{\mathfrak{q}\mathfrak{p}})^{-1}\left[-\mathbf{K}^\xi_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{f}}^*_\mathfrak{p})^i - \mathbf{K}^\eta_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{g}}^*_\mathfrak{p})^i - \mathbf{K}^\zeta_{\mathfrak{q}\mathfrak{p}}(\hat{\mathbf{h}}^*_\mathfrak{p})^i + \mathbf{F}^0_{\mathfrak{q}\mathfrak{m}}\hat{\mathbf{w}}^n_\mathfrak{m}\right]\,. \tag{1.63}$$

The matrices $(\mathbf{K}^1_{\mathfrak{q}\mathfrak{p}})^{-1}\mathbf{K}^\xi_{\mathfrak{q}\mathfrak{p}}$ have the remarkable property that all their eigenvalues are zero, which means that, at least in the homogeneous case with no source terms, the conditions for Banach fixed point theorem hold and therefore uniqueness and convergence to the unique solution are proved. Once the $\mathbf{q}_\mathfrak{p}$ terms have been computed in this way, the local evolution of the the polynomial inside each cell will be provided by the expansion (1.48). The state $\mathbf{q}_h(x,y,z,t)$ as given by (1.48) can be regarded as a predictor of the true solution at the time-level t^{n+1}, which however requires taking into account the coupling of the cells through the numerical fluxes. However, fluxes and sources can now be easily obtained through the expressions

$$\mathbf{f}_{i+\frac{1}{2},jk} = \frac{1}{\Delta t}\frac{1}{\Delta y_j}\frac{1}{\Delta z_k}\int_{t^n}^{t^{n+1}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}}\tilde{\mathbf{f}}_{\mathrm{RP}}(x_{i+\frac{1}{2}},y,z,t)\,dz\,dy\,dt\,, \tag{1.64}$$

$$\mathbf{g}_{i,j+\frac{1}{2},k} = \frac{1}{\Delta t}\frac{1}{\Delta x_i}\frac{1}{\Delta z_k}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{z_{k-\frac{1}{2}}}^{z_{k+\frac{1}{2}}}\tilde{\mathbf{g}}_{\mathrm{RP}}(x,y_{j+\frac{1}{2}},z,t)\,dz\,dx\,dt, \tag{1.65}$$

$$\mathbf{h}_{ij,k+\frac{1}{2}} = \frac{1}{\Delta t}\frac{1}{\Delta x_i}\frac{1}{\Delta y_j}\int_{t^n}^{t^{n+1}}\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}}\int_{y_{j-\frac{1}{2}}}^{y_{j+\frac{1}{2}}}\tilde{\mathbf{h}}_{\mathrm{RP}}(x,y,z_{k+\frac{1}{2}},t)\,dy\,dx\,dt \tag{1.66}$$

where $\tilde{\mathbf{f}}_{\rm RP}(x_{i+\frac{1}{2}},y,z,t)$ is a numerical flux requiring the solution of the Riemann problem

$$\tilde{\mathbf{f}}_{\rm RP} = \tilde{\mathbf{f}}\Big(\mathbf{q}_h^-(x_{i+\frac{1}{2}},y,z,t),\mathbf{q}_h^+(x_{i+\frac{1}{2}},y,z,t)\Big)\ , \tag{1.67}$$

where $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$ are the left and right boundary extrapolated states at the interface, respectively. The overall scheme will finally adopt the form

$$\mathbf{U}_{ijk}^{n+1} \quad = \quad \mathbf{U}_{ijk}^{n} - \frac{\Delta t}{\Delta x_i}\left(\mathbf{f}_{i+\frac{1}{2},jk} - \mathbf{f}_{i-\frac{1}{2},jk}\right) - \frac{\Delta t}{\Delta y_j}\left(\mathbf{g}_{i,j+\frac{1}{2},k} - \mathbf{g}_{i,j-\frac{1}{2},k}\right) - \frac{\Delta t}{\Delta z_k}\left(\mathbf{h}_{ij,k+\frac{1}{2}} - \mathbf{h}_{ij,k-\frac{1}{2}}\right) \tag{1.68}$$

and therefore appear as a high-order, one-step time-update scheme, with no need for Runge–Kutta substeps.

What is worth stressing here is that because of the local feature of the predictor step, the local spacetime DG scheme is also suitable for treating stiff source terms and indeed it has been successfully applied to the solution of the relativistic resistive magnetohydrodynamic equations to study magnetic reconnection in relativistic regimes [61].

Chapter 2

Discontinuous Galerkin schemes

2.1 Introduction

Broadly speaking, *discontinuous Galerkin (DG) methods* can be considered as numerical methods for the weak formulation of the equations. They were first applied to first-order equations by [49], but their widespread use followed from the application to hyperbolic problems by Cockburn and collaborators in a series of articles [14, 11, 9]. In these works, the typical DG discretisation in space was combined with a stable high-order Runge–Kutta discretisation in time to solve nonlinear time-dependent conservation laws.

Very briefly, in the discontinuous Galerkin finite element framework the coefficients of higher order polynomials are directly evolved in time for each cell, without the need of using a reconstruction operator. This feature of DG schemes is in common with the classical finite element method (FEM). However, unlike classical finite elements, the numerical solution given by a DG scheme is discontinuous at element interfaces and this discontinuity is resolved by the use of a numerical flux function, which is a common feature with HRSC finite volume schemes. DG methods have a number of advantages with respect to classical finite volume schemes: they reach arbitrary order of accuracy in space on general unstructured meshes and allows easily for hp-adaptation, i.e. they allow for refinement and recoarsening of the mesh and for a dynamical adaptation of the polynomial degree of the numerical solution. Moreover, since they do not requires the introduction of reconstruction stencils, they typically require less MPI communications on parallel codes. Moreover, Jiang and Shu have found a very elegant and general proof of nonlinear stability in L_2-norm [33] for DG methods.

2.2 Modal (or hierarchical) basis

A modal, or hierarchical polynomial basis of maximum degree M, denoted as $\Psi_k(\xi)$ here for convenience, is a set of $M+1$ linearly independent polynomials, with degree from zero to the maximum degree M. In practice, the polynomials $\Psi_k(\xi)$ can be chosen among the orthogonal Legendre polynomials [47], which, rescaled on the reference element $\mathscr{E} = [0,1]$, are referred to as the shifted Legendre polynomials and are given by

$$\begin{aligned}
\Psi_0(\xi) &= 1\,, \\
\Psi_1(\xi) &= 2\xi - 1\,, \\
\Psi_2(\xi) &= 6\xi^2 - 6\xi + 1\,, \\
\Psi_3(\xi) &= 20\xi^3 - 30\xi^2 + 12\xi - 1\,, \\
\Psi_4(\xi) &= 70\xi^4 - 140\xi^3 + 90\xi^2 - 20\xi + 1\,, \\
\Psi_5(\xi) &= 252\xi^5 - 630\xi^4 + 560\xi^3 - 210\xi^2 + 30\xi - 1\,, \\
&\vdots
\end{aligned} \tag{2.1}$$

A few of them are reported for illustration in Fig. 2.3.

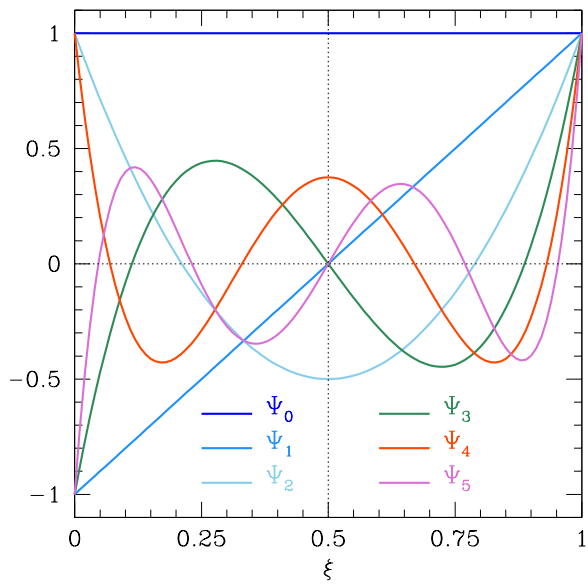

Figure 2.1: Functional behaviour of the first six shifted Legendre polynomials of a modal basis as given by expressions (2.1). Figure taken from [50].

2.3 The DG discretization

We are again assuming a system of PDE in conservative form as

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}\left(\mathbf{U}\right) = 0, \qquad \mathbf{x} \in \Omega \subset \mathbb{R}^d, \quad t \in \mathbb{R}_0^+, \tag{2.2}$$

where $\mathbf{U}$ is the vector of so-called conserved quantities, while $\mathbf{F}(\mathbf{U}) = (\mathbf{f}, \mathbf{g}, \mathbf{h})$ is a non-linear flux tensor that depends on the state $\mathbf{U}$. The computational domain Ω is discretized by a Cartesian grid composed by elements I_i, namely

$$\Omega = \bigcup_{i=1}^{N_E} I_i \,, \tag{2.3}$$

where the index i ranges from 1 to the total number of elements N_E. In the following, we denote the cell volume by $|I_i| = \int_{I_i} d\mathbf{x}$. At the beginning of each time-step, the numerical solution of Eq. (2.2) is represented within each cell I_i by piecewise polynomials of maximum degree $N \geq 0$ as

$$\mathbf{U}_h(\mathbf{x}, t^n) = \sum_{l=0}^{N} \Phi_l(\mathbf{x}) \hat{\mathbf{U}}_l^n = \Phi_l(\mathbf{x}) \hat{\mathbf{U}}_l^n \quad \mathbf{x} \in I_i \,, \tag{2.4}$$

where $\mathbf{U}_h$ is referred to as the *discrete representation* of the solution, while the coefficients $\hat{\mathbf{U}}_l^n$ are usually called the *degrees of freedom*.

We multiply Eq. (2.2) by a test function Φ_k, identical to the spatial basis functions of Eq. (2.4). Second, we integrate over the space element I_i. The flux divergence term is then integrated by parts in space, thus yielding

$$\int_{I_i} \Phi_k \frac{\partial \mathbf{U}_h}{\partial t} d\mathbf{x} + \int_{\partial I_i} \Phi_k \, \mathbf{F}\left(\mathbf{U}_h\right) \cdot \mathbf{n} \, dS - \int_{I_i} \nabla \Phi_k \cdot \mathbf{F}\left(\mathbf{U}_h\right) d\mathbf{x} = 0, \tag{2.5}$$

where $\mathbf{n}$ is the outward pointing unit normal vector on the surface ∂I_i of element I_i. Since the discrete solution is allowed to be discontinuous at element boundaries, the surface integration involved in the second term of (2.5) is done through the solution of a Riemann problem, which is therefore deeply rooted in the DG scheme and guarantees the overall upwind character of the method [15, 13, 12, 10, 16].

Explicit DG schemes are limited by a Courant-Friedrichs-Lewy (CFL) restriction. There is not universal agreement about the most appropriate form of the time-step restriction. According to [38] it is

$$\Delta t < \frac{1}{(2N+1)} \frac{h}{|\lambda_{\max}|} \,, \tag{2.6}$$

while, according to [29, 48], it should be

$$\Delta t < \frac{1}{(N+1)^2} h \tag{2.7}$$

where h and $|\lambda_{\max}|$ are a characteristic mesh size and the maximum signal velocity, respectively. Although the constraint imposed by Eq. (2.6) may appear very severe and such as to make DG methods of little practical use, especially at high orders, this limitation is mitigated by two properties of DG methods. The first one is that discretisation with spacings much larger than those used in lower-order methods can be used with success. The second one is that local time-stepping can be performed, whereby each element is updated in time at its own maximum stable time-step, with a potential speedup that is progressively higher if only a small fraction of elements requires a small time-step [see [30] for details].

2.4 Runge-Kutta DG schemes

Substituting (2.4) into (2.5) yields the expression

$$\sum_{l=0}^{M} \left(\int_0^1 \Phi_l \Phi_k \, d\xi \right) \frac{d\hat{\mathbf{U}}_l}{dt} + \Big[\Phi_k \mathbf{F}^* \Big]_0^1 - \int_0^1 \mathbf{F}^*(\mathbf{U}(\xi,t)) \, \frac{d\Phi_k}{d\xi} \, d\xi = 0 \,, \tag{2.8}$$

which represents a system of (coupled) ordinary differential equations in time for the degrees of freedom $\hat{\mathbf{U}}_l(t)$. The advantage of this procedure is that the basis functions Φ_l are known analytically, so that also their derivatives, $\frac{d\Phi_k}{d\xi}$, are also known analytically. As a result, the integral in the first term in (2.8) is analytic and needs to be calculated only once. As an example, let us consider the case of a fourth-order representation, namely with $M = 3$, of the function $\mathbf{U}(\xi,t)$. In this case it is more convenient to adopt a modal basis. We write

$$\mathbf{U}(\xi,t) = \hat{\mathbf{U}}_0(t)\Phi_0(\xi) + \hat{\mathbf{U}}_1(t)\Phi_1(\xi) + \hat{\mathbf{U}}_2(t)\Phi_2(\xi) + \hat{\mathbf{U}}_3(t)\Phi_3(\xi) \,. \tag{2.9}$$

The corresponding system of (coupled) ordinary differential equations obtained from (2.8) is

$$\frac{d\hat{\mathbf{U}}_0}{dt} + \mathbf{F}^*(1) - \mathbf{F}^*(0) = 0 \,, \tag{2.10}$$

$$\frac{1}{3} \quad \frac{d\hat{\mathbf{U}}_1}{dt} + \Phi_1(1)\mathbf{F}^*(1) - \Phi_1(0)\mathbf{F}^*(0) - \int_0^1 \mathbf{F}^*(\mathbf{U}(\xi,t)) \, \frac{d\Phi_1}{d\xi} \, d\xi = 0 \,, \tag{2.11}$$

$$\frac{1}{5} \quad \frac{d\hat{\mathbf{U}}_2}{dt} + \Phi_2(1)\mathbf{F}^*(1) - \Phi_2(0)\mathbf{F}^*(0) - \int_0^1 \mathbf{F}^*(\mathbf{U}(\xi,t)) \, \frac{d\Phi_2}{d\xi} \, d\xi = 0 \,, \tag{2.12}$$

$$\frac{1}{7} \quad \frac{d\hat{\mathbf{U}}_3}{dt} + \Phi_3(1)\mathbf{F}^*(1) - \Phi_3(0)\mathbf{F}^*(0) - \int_0^1 \mathbf{F}^*(\mathbf{U}(\xi,t)) \, \frac{d\Phi_3}{d\xi} \, d\xi = 0 \,, \tag{2.13}$$

and can be solved through a standard Runge–Kutta discretisation in time, leading to a *Runge–Kutta discontinuous Galerkin scheme*.

To first order, namely when considering only Eq. (2.10), the RKDG scheme coincides with a first-order finite-volume scheme. Moreover, the values of the fluxes at the cell borders, $\mathbf{F}^*(0)$ and $\mathbf{F}^*(1)$, can be obtained by solving a Riemann problem, thus incorporating the upwind property into the RKDG scheme. Finally, at least in principle,

the solution of such Riemann problems does not require any spatial reconstruction at the interface between adjacent cells. The value of **U** at the cell borders is in fact naturally provided by the expansion (2.4) computed at the proper locations. However, if the discontinuities are strong, the scheme generates significant oscillations. Moreover, the spatial integrals in Eq. (2.10)–(2.13) are evaluated using *Gaussian quadrature points* of the appropriate order (see, [44] for a basic introduction to Gaussian quadrature).

2.5 Non-linear L_2 stability

Since DG schemes are linear in the sense of Godunov, they will inevitably produce oscillations in the presence of discontinuities (recall Godunov theorem). There is nevertheless an important result due to Jiang and Shu [33] who proved a discrete cell entropy inequality for the square entropy of the Discontinuous Galerkin scheme when applied to scalar nonlinear hyperbolic conservation laws. Let us see how the analysis can be carried out.

We consider the scalar conservation law in the usual form

$$\partial_t \mathrm{U} + \partial_x \mathrm{f} = 0\,, \tag{2.14}$$

and we now introduce a few definitions.

Definition 1. *A numerical scheme is said to satisfy the L_2 stability condition if*

$$\partial_t \int_\Omega \left(\frac{\mathrm{U}_h^2}{2} \right) dx \;\leq 0\,. \tag{2.15}$$

Definition 2. *The pair of functions ($\mathcal{S}(U)$,$\mathcal{F}(U)$) is called an entropy pair for the conservation law* (2.14) *if the entropy $\mathcal{S}$ satisfies*

$$\frac{\partial^2 \mathcal{S}}{\partial \mathrm{U}^2} > 0 \tag{2.16}$$

and if the entropy flux $\mathcal{F}(U)$ satisfies

$$\frac{\partial \mathcal{F}(\mathrm{U})}{\partial \mathrm{U}} = \frac{\partial \mathcal{S}(\mathrm{U})}{\partial \mathrm{U}} \frac{\partial \mathrm{f}(\mathrm{U})}{\partial \mathrm{U}}\,. \tag{2.17}$$

We notice that,

- if U is a continuous differentiable solution of (2.14), then

$$\partial_t \mathrm{U} + \frac{\partial \mathrm{f}(\mathrm{U})}{\partial \mathrm{U}} \partial_x \mathrm{U} = 0\,, \tag{2.18}$$

  which, after multiplication by $\frac{\partial \mathcal{S}(\mathrm{U})}{\partial \mathrm{U}}$ becomes

$$\begin{aligned} &\frac{\partial \mathcal{S}(\mathrm{U})}{\partial \mathrm{U}} \partial_t \mathrm{U} + \frac{\partial \mathcal{S}(\mathrm{U})}{\partial \mathrm{U}} \frac{\partial \mathrm{f}(\mathrm{U})}{\partial \mathrm{U}} \partial_x \mathrm{U} = \\ &\frac{\partial \mathcal{S}(\mathrm{U})}{\partial \mathrm{U}} \partial_t \mathrm{U} + \frac{\partial \mathcal{F}(\mathrm{U})}{\partial \mathrm{U}} \partial_x \mathrm{U} = \\ &\partial_t \mathcal{S} + \partial_x \mathcal{F} = 0\,. \end{aligned} \tag{2.19}$$

- if U is a weak solution of the conservation law (2.14), then any entropy pair will satisfy

$$\partial_t \mathcal{S} + \partial_x \mathcal{F} \leq 0\,. \tag{2.20}$$

Definition 3. *We define the square entropy and its associated flux as*

$$\mathcal{S}_2 = \frac{\mathrm{U}^2}{2} \qquad\qquad \mathcal{F}_2 = \mathrm{U}\ \mathrm{f}(\mathrm{U}) - \int F(\mathrm{U})\, d\mathrm{U} \tag{2.21}$$

Theorem *The function pair $(\mathcal{S}_2, \mathcal{F}_2)$ is an entropy pair.*

Proof. The first property follows directly from the fact that

$$\frac{\partial^2 \mathcal{S}_2}{\partial \mathrm{U}^2} = 1 > 0\,. \tag{2.22}$$

Moreover,

$$\begin{aligned} \frac{\partial \mathcal{F}_2}{\partial \mathrm{U}} &= \mathrm{f}(\mathrm{U}) + \mathrm{U}\frac{\partial \mathrm{f}}{\partial \mathrm{U}} - \mathrm{f}(\mathrm{U}) = \\ &= \mathrm{U}\frac{\partial \mathrm{f}}{\partial \mathrm{U}} = \frac{\partial \mathcal{S}_2}{\partial \mathrm{U}}\frac{\partial \mathrm{f}}{\partial \mathrm{U}} \end{aligned} \tag{2.23}$$

□

Definition 4. *A Lipschitz continuous function $\mathrm{f}_{i+1/2} = \mathrm{f}_{i+1/2}(\mathrm{U}^-, \mathrm{U}^+)$ of the two states U^- and U^+ is called an e-flux for the conservation law* (2.14) *if*

$$\mathrm{f}_{i+1/2}(\mathrm{U}, \mathrm{U}) = f(\mathrm{U})\,, \tag{2.24}$$

and

$$\int_{\mathrm{U}^-}^{\mathrm{U}^+} (f(\mathrm{U}) - \mathrm{f}_{i+1/2}(\mathrm{U}^-, \mathrm{U}^+))d\mathrm{U} \geq 0\,. \tag{2.25}$$

With all this machinery at hand, we assume that $\mathrm{U} \in V_h$ is an approximate solution of (2.14) in a discrete function space $V_h \subset L_2$. We then multiply (2.14) by $\Phi \in V_h$ from the same function space and we integrate by parts, obtaining

$$\int_{I_i} \Phi \partial_t \mathrm{U}\, dx + (\mathrm{f}_{i+1/2}\Phi^-_{i+1/2} - \mathrm{f}_{i-1/2}\Phi^+_{i-1/2}) - \int_{I_i} \mathrm{f}(\mathrm{U})\partial_x \Phi\, dx = 0\,, \tag{2.26}$$

where $I_i = [x_{i-1/2}; x_{i+1/2}]$ is the cell size while

- $\mathrm{f}_{i-1/2} = \mathrm{f}_{i-1/2}(\mathrm{U}^-_{i-1/2}, \mathrm{U}^+_{i-1/2})$ is an e-flux between I_{i-1} and I_i.
- $\mathrm{f}_{i+1/2} = \mathrm{f}_{i+1/2}(\mathrm{U}^-_{i+1/2}, \mathrm{U}^+_{i+1/2})$ is an e-flux between I_i and I_{i+1}.
- $\Phi^-_{i+1/2} = \Phi(x^-_{i+1/2})$ and $\Phi^+_{i-1/2} = \Phi(x^+_{i-1/2})$

Theorem *The numerical solution $\mathrm{U} \in V_h$ of the Discontinuous Galerkin scheme* (2.26) *(in which $\mathrm{f}_{i\pm 1/2}$ is an e-flux) obeys the discrete cell-entropy condition*

$$\int_{I_i} \partial_t \left(\frac{\mathrm{U}^2}{2}\right) dx + \hat{F}_{i+1/2} - \hat{F}_{i-1/2} \leq 0\,. \tag{2.27}$$

Proof. Since both U and Φ belong to V_h, we can replace Φ with U in (2.26), to find

$$\int_{I_i} \partial_t \left(\frac{\mathrm{U}^2}{2}\right) dx + (\mathrm{f}_{i+1/2}\mathrm{U}^-_{i+1/2} - \mathrm{f}_{i-1/2}\mathrm{U}^+_{i-1/2}) - \int_{I_i} \mathrm{f}(\mathrm{U})\partial_x \mathrm{U}\, dx = 0\,. \tag{2.28}$$

We now rewrite the last term as

$$\int_{I_i} \mathrm{f}(\mathrm{U})\partial_x \mathrm{U}\, dx = \int_{\mathrm{U}^+_{i-1/2}}^{\mathrm{U}^-_{i+1/2}} \mathrm{f}(\mathrm{U})d\,\mathrm{U} = g(\mathrm{U}^-_{i+1/2}) - g(\mathrm{U}^+_{i-1/2}) \tag{2.29}$$

with the definition

$$g = \int \mathrm{f}(\mathrm{U})d\,\mathrm{U}\,. \tag{2.30}$$

Therefore, (2.28) takes the form

$$\int_{I_i} \partial_t \left(\frac{\mathrm{U}^2}{2}\right) dx + (\mathrm{f}_{i+1/2}\mathrm{U}^-_{i+1/2} - g(\mathrm{U}^-_{i+1/2})) - (\mathrm{f}_{i-1/2}\mathrm{U}^+_{i-1/2} - g(\mathrm{U}^+_{i-1/2})) = 0\,, \tag{2.31}$$

which we can also rewrite as

$$\begin{aligned}
\int_{I_i} \partial_t \left(\frac{\mathrm{U}^2}{2}\right) dx \quad &+ \quad (\mathrm{f}_{i+1/2}\mathrm{U}^-_{i+1/2} - g(\mathrm{U}^-_{i+1/2})) - (\mathrm{f}_{i-1/2}\mathrm{U}^-_{i-1/2} - g(\mathrm{U}^-_{i-1/2})) \\
&- \quad (\mathrm{f}_{i-1/2}\mathrm{U}^+_{i-1/2} - g(\mathrm{U}^+_{i-1/2})) + (\mathrm{f}_{i-1/2}\mathrm{U}^-_{i-1/2} - g(\mathrm{U}^-_{i-1/2})) \\
\int_{I_i} \partial_t \left(\frac{\mathrm{U}^2}{2}\right) dx \quad &+ \quad \hat{F}_{i+1/2} - \hat{F}_{i-1/2} + \hat{R}_{i-1/2} = 0\,,
\end{aligned} \tag{2.32}$$

where

$$\hat{F}_{i+1/2} = (\mathrm{f}_{i+1/2}\mathrm{U}^-_{i+1/2} - g(\mathrm{U}^-_{i+1/2})) \tag{2.33}$$

$$\hat{F}_{i-1/2} = (\mathrm{f}_{i-1/2}\mathrm{U}^-_{i-1/2} - g(\mathrm{U}^-_{i-1/2})) \tag{2.34}$$

$$\hat{R}_{i-1/2} = -(\mathrm{f}_{i-1/2}\mathrm{U}^+_{i-1/2} - g(\mathrm{U}^+_{i-1/2})) + (\mathrm{f}_{i-1/2}\mathrm{U}^-_{i-1/2} - g(\mathrm{U}^-_{i-1/2})) \tag{2.35}$$

We now notice that the terms $\hat{F}_{i+1/2}$ and $\hat{F}_{i-1/2}$ are discrete entropy fluxes consistent with the continuous entropy flux of Eq. (2.21). The term $\hat{R}_{i-1/2}$ can be written more coincisely as

$$\hat{R}_{i-1/2} = \int_{\mathrm{U}^-_{i-1/2}}^{\mathrm{U}^+_{i-1/2}} \left(\mathrm{f}(\mathrm{U}) - \mathrm{f}_{i-1/2}(\mathrm{U}^-_{i-1/2}, \mathrm{U}^+_{i-1/2})\right) d\mathrm{U} \tag{2.36}$$

Due to the property (2.25), we know that $\hat{R}_{i-1/2} \geq 0$, and therefore, from (2.32) it follows that the inequality (2.27) of the theorem is proved. □

Corollary *The Discontinuous Galerkin scheme* (2.26) *is L_2 stable.*

In fact, summing (2.27) over all elements I_i and imposing either periodic boundary conditions or zero fluxes at the borders of the domain Ω we get

$$\int_{\Omega} \partial_t \left(\frac{\mathrm{U}^2}{2}\right) dx \leq 0\,. \tag{2.37}$$

The L_2 stability property (2.37) holds for arbitary high order semi-discrete DG schemes and for any non-linear hyperbolic conservation law. The only requirements are that the space of the approximating solution and of the test functions are the same, and that $f_{i\pm1/2}$ are e-fluxes.

2.6 ADER-DG schemes

A one-step time-update high order DG scheme can be obtained by exploiting the solution of the Local Discontinuous Galerkin predictor that we have introduced in the context of FV methods. To this extent, let us go back to Eq. (2.5) and integrate it in time, obtaining

$$\int_{t^n}^{t^{n+1}} \int_{I_i} \Phi_k \frac{\partial \mathbf{U}_h}{\partial t} d\mathbf{x}dt + \int_{t^n}^{t^{n+1}} \int_{\partial I_i} \Phi_k \, \mathbf{F}\left(\mathbf{U}_h\right) \cdot \mathbf{n} \, dSdt - \int_{t^n}^{t^{n+1}} \int_{I_i} \nabla\Phi_k \cdot \mathbf{F}\left(\mathbf{U}_h\right) d\mathbf{x}dt = 0\,. \tag{2.38}$$

Whatever numerical flux function (Riemann solver) is chosen, denoted as $\tilde{\mathbf{f}}_{\mathrm{RP}}$ below, the time integration of the second and of the third term of Eq. (2.39) must be performed to the desired order of accuracy. To this extent, we use the local space-time predictor $\mathbf{q}_h$ to compute the numerical flux function of the second term as $\tilde{\mathbf{f}}_{\mathrm{RP}}\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right)$

and the physical flux of the third term as $\mathbf{F}\left(\mathbf{q}_h\right)$. We emphasize that $\mathbf{q}_h^-$ and $\mathbf{q}_h^+$ are the left and right states of the Riemann problem. On the other hand, after inserting $\mathbf{U}_h$, as given by (2.4), in the first term of (2.39) we find the following arbitrary high order accurate one-step discontinuous Galerkin (ADER-DG) scheme:

$$\left(\int_{I_i} \Phi_k \Phi_l d\mathbf{x}\right)\left(\hat{\mathbf{U}}_l^{n+1}-\hat{\mathbf{U}}_l^n\right)+\int_{t^n}^{t^{n+1}} \int_{\partial I_i} \Phi_k \tilde{\mathbf{f}}_{\mathrm{RP}}\left(\mathbf{q}_h^-, \mathbf{q}_h^+\right) \cdot \mathbf{n}\, dS dt-\int_{t^n}^{t^{n+1}} \int_{I_i} \nabla \Phi_k \cdot \mathbf{F}\left(\mathbf{q}_h\right) d\mathbf{x} dt=0 . \quad (2.39)$$

The effective order of accuracy of the ADER-DG scheme resulting from (2.39) is $N+1$, both in space and in time, as long as the solution remains smooth. In spite of its great ability in achieving sub-cell resolution even on very coarse grids, the ADER-DG scheme, as well any other unlimited DG scheme, will fail at discontinuities due to the Gibbs phenomenon. For this reason it is necessary to introduce some sort of limiter, which should ideally preserve the typical sub-cell resolution properties of the DG method. To this extent, a peculiar approach involving sub-cell finite volume schemes has been recently proposed by [26] as an alternative to slope limiters techniques (borrowed from finite-volume methods), which normally destroy the sub-cell resolution properties of the DG method [14, 45, 5].

The overall scheme can be schematically described as follows:

- *a predictor step*, in which Eq. (2.2) is solved within each element *in the small* by means of a locally implicit space-time discontinuous Galerkin scheme (see Lecture 2, [20, 18]);
- *a pure discontinuous Galerkin (DG) scheme*, which, by exploiting the information obtained by the predictor, allows to compute the solution at the next time level through a single one-step corrector;
- *an a posteriori sub-cell limiter*, which recomputes the solution of the troubled cells needing a limiter through an ADER-WENO finite volume scheme acting at the sub-cell level [26, 63];
- *an adaptive mesh refinement* (AMR) approach, which is implemented according to a cell-by-cell strategy and must be properly nested within the sub-cell philosophy (not done in these lectures but see [63]).

2.7 A sub-cell limiter for DG

Should we implement the ADER-DG scheme as it is described above, we would obtain a numerical scheme capable of resolving smooth solutions with an order of accuracy equal to $N+1$, where N is the degree of the chosen polynomials, but totally inadequate for discontinuous solutions, for which the Gibbs phenomenon would quickly lead to spurious oscillations and even to a breakdown of the scheme.

The procedures that have been adopted to overcome this difficulty can be roughly divided in two classes. From one side, it is possible to introduce additional numerical dissipation, either in the form of artificial viscosity [51, 43, 8], or by means of filtering [48]. From another side, it is possible to isolate the so-called *troubled cells*, namely those affected by spurious oscillations, and adopt for them some sort of nonlinear finite-volume-type slope-limiting procedure [16, 46, 5, 42], either based on nonlinear WENO or HWENO reconstruction or by applying a TVB limiter to the higher order moments of the discrete solution. The drawback of this strategy is that in most cases the subcell resolution properties of the DG scheme are immediately lost. A new idea for an a-posteriori limiter has been recently proposed by [26] and it works as follows.

- The *unlimited* ADER-DG scheme (2.39) is first used to evolve the solution from time t^n to t^{n+1}, producing a so-called candidate solution $\mathbf{U}_h^*(\mathbf{x}, t^{n+1})$ inside each cell.
- The candidate solution $\mathbf{U}_h^*(\mathbf{x}, t^{n+1})$ is then checked against two different criteria to verify its validity, namely
    1. *Physical admissibility detection:* if the conversion from conservative to primitive variables fails, or if either the pressure or the rest mass density drops below a threshold value, or if we encounter superluminal velocities, then the cell is flagged as *troubled*.

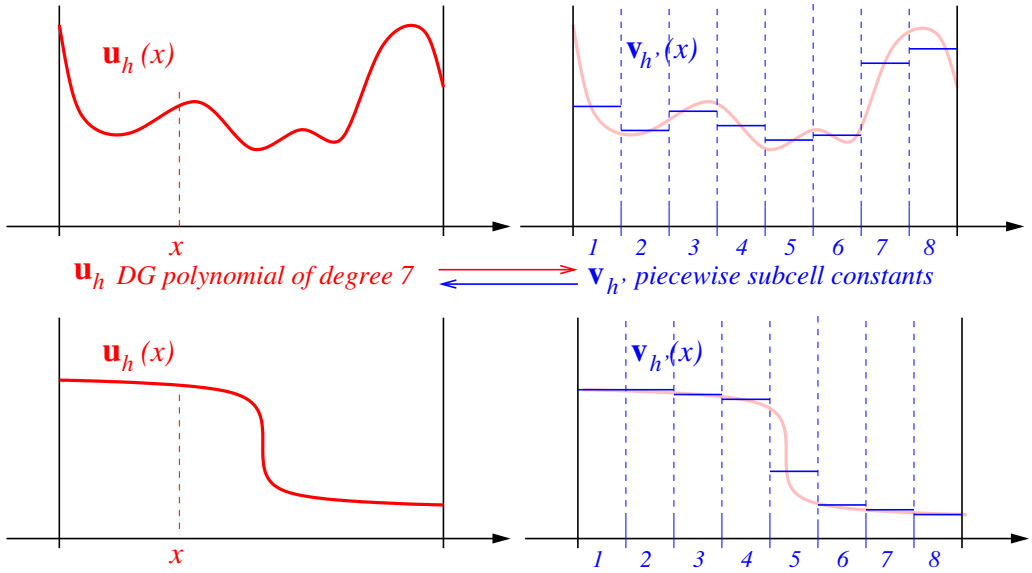

Figure 2.2: Examples of DG polynomials $\mathbf{U}_h$ on a cell (red) and associated projection $\mathbf{v}_h = \mathcal{P}(\mathbf{U}_h)$ onto subcell averages (blue). The information contained in $\mathbf{U}_h$ can be recovered from $\mathbf{v}_h$ via the subcell reconstruction operator for $N_s \geq N+1$. Figure taken from [26].

2. *Numerical admissibility detection:* if the polynomial representing the candidate solution does not lie between the minimum and the maximum of the polynomials representing the solution at the previous time step in the set $\mathcal{V}_i$, then the cell is flagged as *troubled*. The set $\mathcal{V}_i$ contains the cell I_i and all its Voronoi neighbor cells that share a common node with I_i. This second detection criterion is specifically designed to remove spurious Gibbs oscillations from the solution and it works as follows: A candidate solution $\mathbf{U}_h^*(\mathbf{x}, t^{n+1})$ is said to fulfill the numerical admissibility detection criterion in cell T_i if the following relation is fulfilled componentwise for all conserved variables:

$$\min_{\mathbf{y}\in\mathcal{V}_i}(\mathbf{U}_h(\mathbf{y}, t^n)) - \boldsymbol{\delta} \leq \mathbf{U}_h^*(\mathbf{x}, t^{n+1}) \leq \max_{\mathbf{y}\in\mathcal{V}_i}(\mathbf{U}_h(\mathbf{y}, t^n)) + \boldsymbol{\delta}, \qquad \forall \mathbf{x} \in T_i, \tag{2.40}$$

where $\mathcal{V}_i$ is a set containing element T_i together with its Voronoi neighbor cells that share a common node with T_i. We see from (2.40) that the discrete maximum principle is now applied in the sense of polynomials. The polynomial that represents the candidate solution on element T_i must remain between the minimum and the maximum of the polynomials that have represented the discrete solution at the old time step in the neighborhood $\mathcal{V}_i$ of cell T_i. The quantity $\boldsymbol{\delta}$ is used to relax the strict maximum principle in order to allow some very small overshoots and undershoots and to avoid problems with roundoff errors. In practice we set

$$\boldsymbol{\delta} = \epsilon \cdot \left(\max_{\mathbf{y}\in\mathcal{V}_i}(\mathbf{U}_h(\mathbf{y}, t^n)) - \min_{\mathbf{y}\in\mathcal{V}_i}(\mathbf{U}_h(\mathbf{y}, t^n)) \right), \tag{2.41}$$

with $\epsilon = 10^{-3}$.

- As soon as a cell is flagged as troubled at the future time t^{n+1}, it generates a local sub-grid formed by $N_s = 2N+1$ cells per space dimension, each of which is assigned a subcell average $\mathbf{v}_h(\mathbf{x}, t^n)$ by means of a L_2 projection obtained from the DG polynomial at the *previous time level* t^n, i.e.

$$v_{i,j}^n = \frac{1}{|S_{i,j}|}\int_{S_{i,j}} \mathbf{u}_h(\mathbf{x}, t^n)\, d\mathbf{x} = \frac{1}{|S_{i,j}|}\int_{S_{i,j}} \hat{\mathbf{u}}_l^n \phi_l(\mathbf{x})\, d\mathbf{x}, \qquad \forall S_{i,j} \in \mathcal{S}_i\,, \tag{2.42}$$

where $\mathcal{S}_i = \bigcup_j S_{i,j}$ is the set of the sub-grid cells. In this way the high accuracy of the DG polynomial is transferred to the subgrid level *before* the spurious oscillations arise. We choose $N_s = 2N + 1$ in order to guarantee that the maximum timestep of the ADER-DG scheme on the main grid

$$\Delta t < \frac{1}{d} \frac{1}{(2N+1)} \frac{h}{|\lambda_{\max}|} \,, \tag{2.43}$$

matches the maximum possible time step of the ADER finite volume scheme on the sub-grid.

- The alternative data representation, provided by Eq. (2.42), is now used as initial condition to evolve the discrete solution with a more robust finite volume scheme on the sub-grid. This is done by resorting to either an ADER-WENO finite volume scheme, or to an even more robust second order TVD shock capturing scheme. For more details see [25, 62]. In practice, on the sub-grid a new evolution from time t^n to t^{n+1} is performed combining a third order WENO finite volume scheme with the spacetime discontinuous Galerkin predictor described in Sect. 1.9.

- The last step requires that the new solution at time t^{n+1} over the sub-grid is projected back to the main grid. This is done imposing that

$$\int_{S_{i,j}} \mathbf{U}_h(\mathbf{x}, t^{n+1}) d\mathbf{x} = \int_{S_{i,j}} \mathbf{v}_h(\mathbf{x}, t^{n+1}) d\mathbf{x}, \qquad \forall S_{i,j} \in \mathcal{S}_i \,. \tag{2.44}$$

  which is a standard reconstruction problem in high order finite volume methods [6, 53, 54] and spectral finite volume schemes [60, 41].

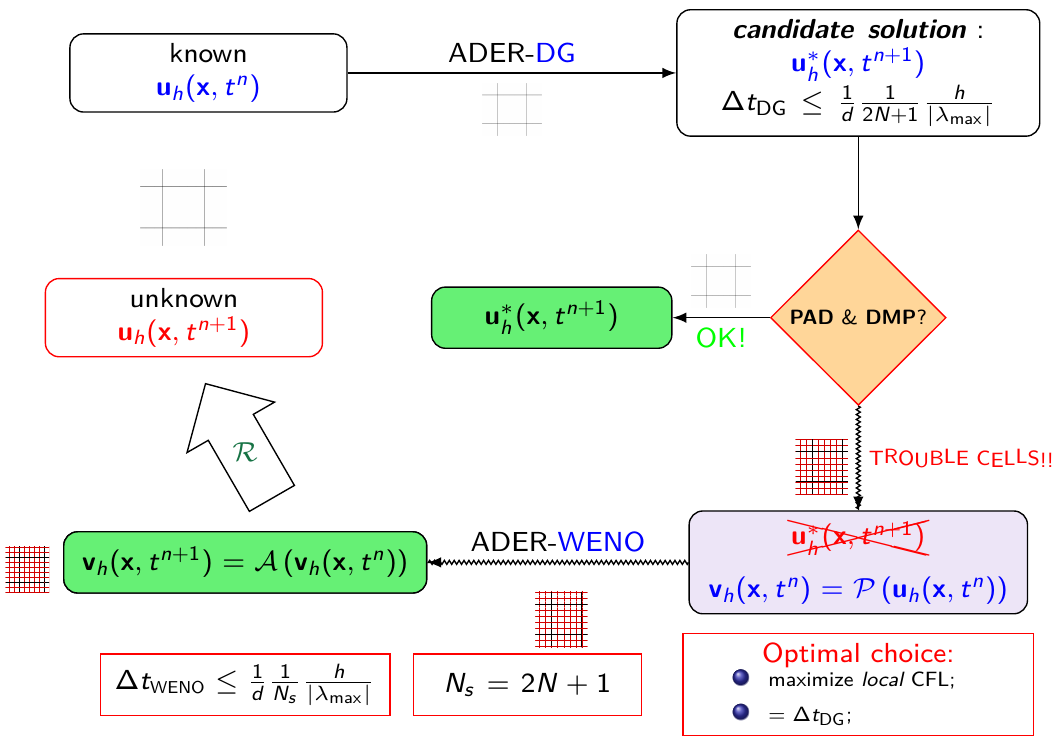

Figure 2.3: A schematic diagram illustrating the sub-cell limiter.

Appendix A

Conservation form of the 1D Euler equations

The Newtonian hydrodynamics equations for a one-dimensional, non self-gravitating fluid

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} = 0 \,, \tag{A.1}$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2 + p)}{\partial x} = 0 \,, \tag{A.2}$$

$$\frac{\partial E}{\partial t} + \frac{\partial[u(E+p)]}{\partial x} = 0 \,, \tag{A.3}$$

$$\tag{A.4}$$

offer a simple example of a system of equations that can be written in conservative form like in (2.2) with a vector of conserved variables given by

$$\mathbf{U} = \begin{bmatrix} \rho \\ \rho u \\ E \end{bmatrix} \tag{A.5}$$

primitive variables given by

$$\mathbf{V} = \begin{bmatrix} \rho \\ u \\ p \end{bmatrix} \tag{A.6}$$

and a flux given by

$$\mathbf{F} = \begin{bmatrix} \rho u \\ \rho u^2 + p \\ u(E+p) \end{bmatrix} \,, \tag{A.7}$$

where $E = \frac{1}{2}\rho u^2 + \rho\epsilon$ is the total energy, ϵ the specific internal energy. In the case of the Euler equations the conservative formulation reflects the physical conservation of specific and well defined quantities, i.e. the mass, the momentum and the energy. However, for other system of equations it may happen that the conservative formulation does not reflect any true conserved physical quantity.

Bibliography

[1] D. Alic, C. Bona, and C. Bona-Casas. Towards a gauge-polyvalent numerical relativity code. *Phys. Rev. D*, 79(4):044026, February 2009.

[2] A. M. Anile. *Relativistic Fluids and Magneto-fluids*. Cambridge University Press, February 1990.

[3] A. M. Anile and O. Muscato. Improved hydrodynamical model for carrier transport in semiconductors. *Phys. Rev. B*, 51:16728–16740, June 1995.

[4] A.M. Anile, S. Pennisi, and M. Sammartino. Covariant radiation hydrodynamics. *Ann. Inst. Henri Poincaré, Phys. Théor.*, 56(1):49–74, 1992.

[5] D. S. Balsara, C. Altmann, C.-D. Munz, and M. Dumbser. A sub-cell based indicator for troubled zones in RKDG schemes and a novel class of hybrid RKDG+HWENO schemes. *Journal of Computational Physics*, 226:586–620, September 2007.

[6] T.J. Barth and P.O. Frederickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. *AIAA paper no. 90-0013*, 28th Aerospace Sciences Meeting January 1990.

[7] W. Boscheri and M. Dumbser. A Direct Arbitrary-Lagrangian-Eulerian ADER-WENO Finite Volume Scheme on Unstructured Tetrahedral Meshes for Conservative and Nonconservative Hyperbolic Systems in 3D. *Journal of Computational Physics*, 275:484–523, 2014.

[8] J. Cesenek, M. Feistauer, J. Horacek, V. Kucera, and J. Prokopova. Simulation of compressible viscous flow in time–dependent domains. *Applied Mathematics and Computation*, 219:7139–7150, 2013.

[9] B. Cockburn. The Runge Kutta Discontinuous Galerkin Method for Conservation Laws V Multidimensional Systems. *Journal of Computational Physics*, 141:199–224, April 1998.

[10] 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*, 54:545–581, 1990.

[11] B. Cockburn, S. How, and C.-W. Shu. TVB Runge Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws IV: The Multidimensional Case. *Math. Comp.*, 54:545, 1990.

[12] 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*, 84:90–113, 1989.

[13] B. Cockburn and C. W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: general framework. *Mathematics of Computation*, 52:411–435, 1989.

[14] B. Cockburn and C.-W. Shu. TVB Runge Kutta Local Projection Discontinuous Galerkin Finite Element Method for Scalar Conservation Laws II: General Framework. *Math. Comp.*, 52:411, 1989.

[15] B. Cockburn and C. W. Shu. The Runge-Kutta local projection P1-Discontinuous Galerkin finite element method for scalar conservation laws. *Mathematical Modelling and Numerical Analysis*, 25:337–361, 1991.

[16] B. Cockburn and C. W. Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. *Journal of Computational Physics*, 141:199–224, 1998.

[17] Richard Courant, Eugene Isaacson, and Mina Rees. On the solution of nonlinear hyperbolic differential equations by finite differences. *Communications on Pure and Applied Mathematics*, 5(3):243–255, 1952.

[18] M. Dumbser, D. S. Balsara, E. F. Toro, and C.-D. Munz. A unified framework for the construction of one-step finite volume and discontinuous Galerkin schemes on unstructured meshes. *Journal of Computational Physics*, 227:8209–8253, September 2008.

[19] M. Dumbser and W. Boscheri. High-order unstructured Lagrangian one–step WENO finite volume schemes for non–conservative hyperbolic systems: Applications to compressible multi–phase flows. *Computers and Fluids*, 86:405–432, 2013.

[20] M. Dumbser, C. Enaux, and E.F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227:3971–4001, 2008.

[21] M. Dumbser and M. Kaeser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221:693–723, February 2007.

[22] M. Dumbser, M. Kaeser, V. A. Titarev, and E. F. Toro. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *Journal of Computational Physics*, 226:204–243, September 2007.

[23] M. Dumbser and M. Käser. Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems. *Journal of Computational Physics*, 221:693–723, 2007.

[24] M. Dumbser, M. Käser, V.A Titarev, and E.F. Toro. Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems. *Journal of Computational Physics*, 226:204–243, 2007.

[25] M. Dumbser, O. Zanotti, A. Hidalgo, and D.S. Balsara. ADER-WENO Finite Volume Schemes with Space-Time Adaptive Mesh Refinement. *Journal of Computational Physics*, 248:257–286, 2013.

[26] M. Dumbser, O. Zanotti, R. Loubère, and S. Diot. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *Journal of Computational Physics*, 278:47–75, December 2014.

[27] S. K. Godunov. A difference method for numerical calculations of discontinuous solutions of the equations of hydrpdynamics. *Mat. Sb.*, 47:271, 1959. in Russian.

[28] J. P. H. Goedbloed and S. Poedts. *Principles of Magnetohydrodynamics*. August 2004.

[29] D. Gottlieb and E. Tadmor. The CFL condition for spectral approximations to hyperbolic initial-boundary value problems. *Mathematics of Computation*, 56:565–588, April 1991.

[30] J.S. Hesthaven and T. Warburton. *Nodal Discontinuous Galerkin Methods: Algorithms, Analysis, and Applications*. Texts in Applied Mathematics. Springer, 2007.

[31] T. Y. Hou and P. G. LeFloch. Why Nonconservative Schemes Converge to Wrong Solutions: Error Analysis. *Math. Comp.*, 62:497–530, 1994.

[32] W. Israel. Nonstationary irreversible thermodynamics: A causal relativistic theory. *Annals of Physics*, 100:310–331, September 1976.

[33] G. S. Jiang and C.-W. Shu. On a cell entropy inequality for discontinuous Galerkin methods. *Mathematics of Computation*, 62:531–538, 1994.

[34] G.-S. Jiang and C.W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, 126:202–228, 1996.

[35] Guang-Shan Jiang and Chi-Wang Shu. Efficient implementation of weighted eno schemes. *J. Comput. Phys*, 126:202–228, 1996.

[36] V. P. Kolgan. Application of the minimum-derivative principle in the construction of finite-difference schemes for numerical analysis of discontinuous solutions in gas dynamics. *Transactions of the Central Aerohydrodynamics Institute*, 3(6):68–77, 1972. in Russian.

[37] S. S. Komissarov. Multidimensional numerical scheme for resistive relativistic magnetohydrodynamics. *Mon. Not. R. Astron. Soc.*, 382:995–1004, December 2007.

[38] L. Krivodonova and R.Qin. An analysis of the spectrum of the discontinuous galerkin method. *Applied Numerical Mathematics*, 64:1–18, 2013.

[39] P. D. Lax and B. Wendroff. Systems of conservation laws. *Commun. Pure Appl. Math.*, 13:217–237, 1960.

[40] R. J. Leveque. *Numerical Methods for Conservation Laws*. Birkhauser Verlag, Basel, 1992.

[41] Y. Liu, M. Vinokur, and Z.J. Wang. Spectral (Finite) Volume Method for Conservation Laws on Unstructured Grids V: Extension to Three-Dimensional Systems. *Journal of Computational Physics*, 212:454–472, 2006.

[42] L.Krivodonova. Limiters for high-order discontinuous Galerkin methods. *Journal of Computational Physics*, 226:879–896, September 2007.

[43] P.-O. Persson and J. Peraire. Sub-cell shock capturing for discontinuous galerkin methods. *AIAA Paper 2006-112*, 2006.

[44] W. H. Press, B. P. Flannery, Saul A. Teukolsky, and W. T. Vetterling. *Numerical Recipes*. Cambridge University Press, Cambridge, England, 1986.

[45] J. Qiu and C. Shu. Runge Kutta Discontinuous Galerkin Methods using WENO limiters. *SIAM J. Sci. Comp.*, 26:907–929, 2005.

[46] J. Qiu and C.W. Shu. Runge-Kutta discontinuous Galerkin method using WENO limiters. *SIAM Journal on Scientific Computing*, 26:907–929, 2005.

[47] A. Quarteroni and A. Valli. *Numerical Approximation of Partial Differential Equations*. Springer, 1997.

[48] D. Radice and L. Rezzolla. Discontinuous Galerkin methods for general-relativistic hydrodynamics: Formulation and application to spherically symmetric spacetimes. *Phys. Rev. D*, 84(2):024010, July 2011.

[49] W. H Reed and T. R. Hill. Triangular mesh methods for the neutron transport equation. Technical report, Los Alamos Scientific Laboratory, 1973.

[50] L. Rezzolla and O. Zanotti. *Relativistic Hydrodynamics*. Oxford University Press, Oxford UK, 2013.

[51] R.Hartmann and P.Houston. Adaptive discontinuous Galerkin finite element methods for the compressible Euler equations. *J. Comp. Phys.*, 183(2):508–532, 2002.

[52] P. Solin. *Partial Differential Equations And the Finite Element Method*. Pure and Applied Mathematics. Wiley-Interscience, 2006.

[53] V. A. Titarev and E. F. Toro. Finite-volume WENO schemes for three-dimensional conservation laws. *Journal of Computational Physics*, 201:238–260, November 2004.

[54] V.A. Titarev and E.F. Toro. ADER schemes for three-dimensional nonlinear hyperbolic systems. *Journal of Computational Physics*, 204:715–736, 2005.

[55] E. F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer-Verlag, 2009.

[56] E. F. Toro, R. C. Millington, and L. A. M. Nejad. Towards very high-order godunov schemes. In *In Godunov Methods: Theory and Applications. Conference in Honour of S K Godunov*, pages 897–902, New York, Boston and London, 2001. Kluwer Academic/Plenum Publishers.

[57] E. F. Toro and V. A. Titarev. ADER schemes for scalar non-linear hyperbolic conservation laws with source terms in three-space dimensions. *Journal of Computational Physics*, 202:196–215, January 2005.

[58] B. van Leer. Towards the Ultimate Conservative Difference Scheme. IV. A New Approach to Numerical Convection. *Journal of Computational Physics*, 23:276, March 1977.

[59] Bram van Leer. Towards the ultimate conservative difference scheme. v. a second-order sequel to godunov's method. *Journal of Computational Physics*, 32(1):101 – 136, 1979.

[60] Z.J. Wang, L. Zhang, and Y. Liu. Spectral (finite) volume method for conservation laws on unstructured grids iv: Extension to two-dimensional euler equations. *Journal of Computational Physics*, 194:716–741, 2004.

[61] O. Zanotti and M. Dumbser. Numerical simulations of high Lundquist number relativistic magnetic reconnection. *Mon. Not. R. Astron. Soc.*, 418:1004–1011, December 2011.

[62] O. Zanotti and M. Dumbser. A high order special relativistic hydrodynamic and magnetohydrodynamic code with space-time adaptive mesh refinement. *Computer Physics Communications*, 188:110–127, 2015.

[63] O. Zanotti, F. Fambri, M. Dumbser, and A. Hidalgo. Spacetime adaptive ader discontinuous galerkin finite element schemes with a posteriori sub-cell finite volume limiting. *Computers and Fluids*, 118(0):204 – 224, 2015.