ADER-DG on adaptive spacetrees

Dominic Etienne Charrier

July 6, 2016

Contents

1. The local space-time arbitrary derivative discontinuous Galerkin method

1.1. Introduction

In this chapter, we first introduce our model problem in section 1.2. Then, after a short discussion of the notation in Sec. (1.3), we derive an incomplete discrete variational formulation of our model problem in Sec. 1.5. The discrete variational formulation describes an one-step time update scheme. It is incomplete in the sense that we need to specify a proper estimate of intermediate values within the time update interval. Two approaches to compute such an estimate are then introduced in Sec. 1.6. The last two sections then discuss the approximation quality of the scheme.

1.2. Governing equations

Let $\Omega \subset \mathbb{R}$ be a domain with boundary $\partial\Omega$, and let $(0, T)$ be a time interval.

A hyperbolic conservation law (with source term) is a first order partial differential equation of the form

$$\frac{\partial}{\partial t}\mathbf{u} + \mathbf{div}\,\mathbf{F}(\mathbf{u}) = \mathbf{s}(\mathbf{u}) \qquad \text{in } \Omega \times (0,T), \tag{1.1a}$$

where $\mathbf{u} = (u_i)_{0\le i<N_{\text{var}}}$ denotes the vector of *conserved quantities*. The flux tensor is denoted by $\mathbf{F}(\mathbf{u}) = (F_{ij}(\mathbf{u}))_{0\le i<N_{\text{var}},0\le j\le d}$, and $\widetilde{\mathbf{s}} = (\widetilde{\mathbf{s}}_i(\mathbf{u}))_{0\le i<N_{\text{var}}}$ denotes a source. The number of conserved quantities is denoted by N_{var}. More details on the notation including a definition of the differential operators appearing in the equation will be given in the next section.

In order to obtain a well-posed problem, one must additionally specify initial conditions

$$\mathbf{u} = \mathbf{u}_0 \qquad \text{on } \Omega \times \{0\}, \tag{1.1b}$$

as well as suitable boundary conditions on the boundary $\partial\Omega$ of the domain. In this work, we will always assume that boundary conditions are imposed on the flux components normal to the domain boundary.

The well-posedness of (1.1a) – (1.1b) does in general depend upon the considered flux $\mathbf{F}(Q)$, the initial conditions, as well as on a suitable choice of boundary conditions. For nonlinear problems, further assumptions on the solution are necessary; see e.g. [5].

Equation (1.1a) is called the *strong form* of the hyperbolic conservation law since it is stated with respect to strong (or classical) derivatives. Solutions (if they exist) to (1.1a) – (1.1b) and suitable boundary conditions are called *strong solutions*.

1.3. Notation

- Scalar quantities with physical meaning (coordinates, pressures, ...) are written in italic letters. This also includes the components of vectors and tensors with physical meaning: see the next bullet point.
- Vectors and tensors with physical meaning (points, velocities, forces, stresses, vector of conserved quantities ...) are printed in bold and using upright letters, e.g., $\mathbf{A}$, $\mathbf{f}$.
- Linear algebra vectors and matrices are written in upright letters and are underlined, e.g., $\underline{\mathrm{A}}$, $\underline{\mathrm{f}}$. Elements belonging to linear algebra vectors and matrices are also written in an upright font but are not underlined, e.g. A_{ij}, f_i.

1.3.1. Vector notation

Let us introduce some vector notation. The outer product $\mathbf{a} \otimes \mathbf{b}$ of two vectors $\mathbf{a}$ and $\mathbf{b}$ is defined as a second-order tensor with the components

$$(\mathbf{a} \otimes \mathbf{b})_{ij} = a_i \, b_j. \tag{1.2}$$

Let us define the contraction or (inner product) of two second-order tensors $\mathbf{A}, \mathbf{B}$ by

$$\mathbf{A} : \mathbf{B} = \sum_{ij} A_{ij} \, B_{ij}. \tag{1.3}$$

We define the left and right contractions of a fourth-order tensor $\mathbf{A}$ and a second-order tensor $\mathbf{B}$ as the second-order tensors $(\mathbf{A} : \mathbf{B})$ and $(\mathbf{B} : \mathbf{A})$ with the components

$$(\mathbf{A} : \mathbf{B})_{ij} = \sum_{kl} A_{ijkl} \, B_{kl}, \qquad (\mathbf{B} : \mathbf{A})_{kl} = \sum_{ij} B_{ij} \, A_{ijkl}. \tag{1.4}$$

For a sufficiently regular vector-valued function $\mathbf{v}$, the classical gradient tensor $\nabla \mathbf{v}$ is defined by

$$(\nabla \mathbf{v})_{ij} = \partial_{x_j} v_i, \tag{1.5}$$

The classical gradient tensor is thus a second-order tensor-valued function. Finally, the classical divergence $\mathbf{div}\ \mathbf{A}$ (a vector-valued function) of a sufficiently regular second-order tensor-valued function $\mathbf{A}$ is defined by

$$(\mathbf{div}\, \mathbf{A})_i = \sum_j \partial_{x_j} A_{ij}. \tag{1.6}$$

1.3.2. Sobolev spaces

Let M be a bounded Lipschitz domain in $\mathbb{R}^d$, $d = 2, 3$.

We denote by $L^2(M)$ the set of measurable real-valued functions that are square integrable in M. We recall that $L^2(M)$ is a Hilbert space (cf. e.g. [6]) with respect to the inner product

$$(u, v)_{L^2(M)} = \int_M u\, v \,\mathrm{d}\mathbf{x}, \tag{1.7}$$

and that the induced norm is given by

$$\|v\|^2_{L^2(M)} = (v, v)_{L^2(M)}. \tag{1.8}$$

In order to keep the notation compact, we will also denote the inner products and norms on the spaces $L^2(M)^{N_{\mathrm{var}}}$ and $L^2(M)^{N_{\mathrm{var}} \times d}$ by the same symbols.

Lastly, we will denote the space of measurable, square integrable mappings of the time interval $(0, T)$ to a normed space X by $L^2(0, T; X)$ (cf. e.g [6]).

1.4. Computational mesh

Let $\mathcal{T}_h$ be a partition on the domain $\Omega \subset \mathbb{R}^d$ consisting of affine quadrilateral/hexahedral elements. We refer to the disjoint open sets $K \in \mathcal{T}_h$ as mesh cells and denote by h_K the diameter of each cell. Let $\mathbf{n}^K$ denote the outward directed normal to the cell boundary ∂K.

An interior face of $\mathcal{T}_h$ is the $d-1$ dimensional intersection $\partial K^+ \cap \partial K^-$, where K^+ and K^- are two adjacent elements of $\mathcal{T}_h$. Similarly, a boundary face of $\mathcal{T}_h$ is the $d-1$ dimensional intersection $\partial K \cap \partial\Omega$ which consists of entire faces of ∂K. Here and in the following, we refer generically to a "face" even in the case $d = 2$.

Finally, let us assign to each element a discrete time line consisting of time steps t_i^K, $i = 0, 1 \ldots, i^K_{\max}$, such that:

$$0 = t_0^K < t_1^K < \ldots < t^K_{i^K_{\max}} = T, \tag{1.9}$$

and let us further define the time step sizes

$$\Delta t_i^K = t_{i+1}^K - t_i^K,\ i = 0, 1 \ldots, i^K_{\max} - 1. \tag{1.10}$$

1.5. An incomplete discrete variational problem

We will now proceed to transform the strong formulation (1.1a) – (1.1b) into a cellwise variational formulation. Let us multiply (1.1a) by any smooth spatial test function $\mathbf{v}$ with compact support in Ω and integrate over $K \times (t_i^K, t_i^K + \Delta t_i^K)$:

$$\int_{t_i^K}^{t_i^K + \Delta t_i^K} \int_K \left(\frac{\partial}{\partial t}\mathbf{u} + \mathbf{div}\, \mathbf{F}(\mathbf{u}) \right) \mathbf{v} \,\mathrm{d}\mathbf{x}\,\mathrm{d}t = \int_{t_i^K}^{t_i^K + \Delta t_i^K} \int_K \mathbf{s}(\mathbf{u})\, \mathbf{v} \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \qquad \text{in } (0, T).$$

From the product rule, it follows that

$$\begin{aligned}\int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \frac{\partial}{\partial t}\mathbf{u}\,\mathbf{v}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t - \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \mathbf{F}(\mathbf{u}) : \nabla\mathbf{v}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ + \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \mathbf{div}\,(\mathbf{F}(\mathbf{u})\mathbf{v})\,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ = \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \mathbf{s}(\mathbf{u})\,\mathbf{v}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t\end{aligned}$$

for any $\mathbf{v}$ as above, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$. From applying a Green's theorem to the third term, we then obtain

$$\begin{aligned}\int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \frac{\partial}{\partial t}\mathbf{u}\,\mathbf{v}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t - \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \mathbf{F}(\mathbf{u}) : \nabla\mathbf{v}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ + \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_{\partial K} \mathbf{F}(\mathbf{u}) : (\mathbf{v}\otimes\mathbf{n}^K)\,\mathrm{d}s\,\mathrm{d}t \\ = \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \mathbf{s}(\mathbf{u})\,\mathbf{v}\,\mathrm{d}\mathbf{x}\,\mathrm{d}t \qquad (1.11)\end{aligned}$$

for any $\mathbf{v}$ as above, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$.

In order to approximate (1.11), we introduce a space of cellwise polynomial, spatial test functions

$$\mathbf{V}_h = \{\mathbf{v}_h \in L^2(\Omega)^{N_{\mathrm{var}}} : \mathbf{v}_h|_K \in Q_N(K)^{N_{\mathrm{var}}},\ \forall K \in \mathcal{T}_h\}, \qquad (1.12)$$

with $Q_N(K)$ denoting the space of polynomials of maximum degree N in each variable on K. We further introduce a space of cellwise polynomial, space-time ansatz functions:

$$\begin{aligned}\widetilde{\mathbf{V}}_h = \Big\{\widetilde{\mathbf{v}}_h \in L^2(0,T;\mathbf{V}_h) : \widetilde{\mathbf{v}}_h|_{K\times(t_i^K, t_i^K+\Delta t_i^K)} \in Q_N(K\times(t_i^K, t_i^K+\Delta t_i^K))^{N_{\mathrm{var}}}), \\ \forall K \in \mathcal{T}_h, \forall i \in \{0,\ldots,i_{\max}-1\}\Big\}. \qquad (1.13)\end{aligned}$$

We then consider the following discrete equation as approximation of (1.11):

$$\begin{aligned}\int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \frac{\partial}{\partial t}\widetilde{\mathbf{u}}_h\,\mathbf{v}_h\,\mathrm{d}\mathbf{x}\,\mathrm{d}t - \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \widetilde{\mathbf{F}}(\widetilde{\mathbf{u}}_h) : \nabla\mathbf{v}_h\,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ + \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_{\partial K} \widetilde{\mathbf{G}}(\widetilde{\mathbf{u}}_h^-, \widetilde{\mathbf{u}}_h^+) : (\mathbf{v}_h\otimes\mathbf{n}^K)\,\mathrm{d}s\,\mathrm{d}t \\ = \int_{t_i^K}^{t_i^K+\Delta t_i^K}\int_K \widetilde{\mathbf{s}}(\widetilde{\mathbf{u}}_h)\mathbf{v}_h\,\mathrm{d}\mathbf{x}\,\mathrm{d}t, \qquad (1.14)\end{aligned}$$

for all $\mathbf{v}_h \in \mathbf{V}_h$, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i^K_{\max} - 1\}$. Let us denote the restriction of functions $\widetilde{\mathbf{v}}_h \in \widetilde{\mathbf{V}}_h$ to a cell $K \in \mathcal{T}_h$ by $\widetilde{\mathbf{v}}^K = \widetilde{\mathbf{v}}|_K$. From integrating the first term of (1.14) by parts in time and using the fact that functions $\mathbf{v}_h \in \mathbf{V}_h$ are constant in time, we obtain:

$$\begin{aligned}\int_K (\widetilde{\mathbf{u}}^K_h(t^K_i + \Delta t^K_i) - \widetilde{\mathbf{u}}^K_h(t^K_i))\, \widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x} = &\int_{t^K_i}^{t^K_i + \Delta t^K_i} \int_K \widetilde{\mathbf{F}}(\widetilde{\mathbf{u}}_h) : \nabla \mathbf{v}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ &- \int_{t^K_i}^{t^K_i + \Delta t^K_i} \int_{\partial K} \widetilde{\mathbf{G}}(\widetilde{\mathbf{u}}^-_h, \widetilde{\mathbf{u}}^+_h) : (\mathbf{v}_h \otimes \mathbf{n}^K) \,\mathrm{d}s\,\mathrm{d}t \\ &+ \int_{t^K_i}^{t^K_i + \Delta t^K_i} \int_K \widetilde{\mathbf{s}}(\widetilde{\mathbf{u}}_h) \mathbf{v}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t, \end{aligned} \tag{1.15}$$

for all $\mathbf{v}_h \in \mathbf{V}_h$, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i^K_{\max} - 1\}$.

Equation (1.15) describes an one-step time update of the cell-local time discrete solution $\widetilde{\mathbf{u}}^K_h(t^K_i)$, $K \in \mathcal{T}_h$, $i \in \{0, 1, \ldots, i^K_{\max}\}$. Note however that the equation is incomplete since we do not know the values of $\widetilde{\mathbf{u}}^K_h$ within the intervals $(t^K_i,\, t^K_i + \Delta t^K_i)$, $i \in \{0, 1, \ldots, i_{\max} - 1\}$. In the local space-time arbitrary derivative discontinuous Galerkin method, those intermediate values are thus replaced by estimates. Two approaches that can be used to obtain such estimates are discussed in the next section.

Our choice of cellwise discontinuous ansatz and test functions implies that the traces of $\widetilde{\mathbf{u}}_h$ and $\mathbf{v}_h$ are not uniquely defined on the interfaces between cells. As it is usual in the discontinuous Galerkin method, we have defined at these interfaces a unique value, the so-called *numerical flux*. We introduce the numerical flux as $\widetilde{\mathbf{G}}(\widetilde{\mathbf{u}}^-_h, \widetilde{\mathbf{u}}^+_h)\mathbf{n}$, where $\mathbf{n} \in \mathbb{R}^d$ denotes the positively signed unit normal vector to the cell boundary ∂K. Note that $\mathbf{n}^K$ denotes the outward directed unit vector and differs generally from $\mathbf{n}$ in its sign. Above, $\widetilde{\mathbf{u}}^+_h$ indicates the solution in the cell adjacent to the interface for which $\mathbf{n}$ and $\mathbf{n}^K$ coincide.

The numerical flux $\widetilde{\mathbf{G}}(\widetilde{\mathbf{u}}^-_h, \widetilde{\mathbf{u}}^+_h)\mathbf{n}$ is obtained by solving a so-called Riemann problem at the interface between two cells. For a range of hyperbolic conservation laws like the compressible Euler equations, exact Riemann solvers exist. In case no exact Riemann solver is known for a problem or the exact solver is too expensive, one can employ approximate Riemann solvers. For an overview of state-of-the-art approximate Riemann solvers see [7].

Note further that the time step sizes Δt^K_i are limited by a Courant-Friedrichs-Lewy (CFL) condition [3]:

$$\Delta t^K_i \leq \frac{1}{d} \frac{1}{2\,N + 1} \frac{h_K}{|\lambda_{\max,K}|}, \tag{1.16}$$

where d is again the space dimension, N is the approximation order, h_K is the diameter of the mesh cell, and $\lambda_{\max,K}$ is the maximum eigenvalue of the (linearised) flux tensor.

Lastly, note that the scheme requires a time step synchronisation between neighbouring cells. Otherwise the time integration of the numerical flux $\widetilde{\mathbf{G}}(\mathbf{u}^-_h, \mathbf{u}^+_h)\mathbf{n}$ cannot be performed since generally the computed solution in one of the neighbouring cells is more

advanced in time than the other. A simple time step synchronisation is realised by fixing $\Delta t_i^K = \Delta t_i$, in every time step $i \in \{0, 1, \ldots\ldots, i_{\max}^K - 1 = i_{\max} - 1\}$, $\forall K \in \mathcal{T}_h$, where the common time step size is chosen as the smallest time step size, i.e.,

$$\Delta t_i = \min_{K \in \mathcal{T}_h} \Delta t_i^K, \qquad i \in \{0, 1, \ldots, i_{\max} - 1\}. \tag{1.17}$$

1.6. The local space-time discontinuous Galerkin predictor

As mentioned in the previous chapter, we proceed by replacing $\widetilde{\mathbf{u}}_h^K$, $K \in \mathcal{T}_h$ within each interval $(t_i^K,\, t_i^K + \Delta t_i^K)$, $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$ by an estimate $\widetilde{\mathbf{q}}_h \in \widetilde{\mathbf{V}}_h$, which we name the *space-time predictor*.

Inserting the space-time predictor in (1.15), then leads to the problem of finding a solution $\widetilde{\mathbf{u}}_h^K(t_i^K + \Delta t_i^K)$ such that:

$$\begin{aligned}\int_K (\widetilde{\mathbf{u}}_h^K(t_i^K + \Delta t_i^K) - \widetilde{\mathbf{u}}_h^K(t_i^K))\, \widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x} - \int_{t_i^K}^{t_i^K + \Delta t_i^K} \int_K \widetilde{\mathbf{F}}(\widetilde{\mathbf{q}}_h) : \nabla \mathbf{v}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ + \int_{t_i^K}^{t_i^K + \Delta t_i^K} \int_{\partial K} \widetilde{\mathbf{G}}(\widetilde{\mathbf{q}}_h^-, \widetilde{\mathbf{q}}_h^+) : (\mathbf{v}_h \otimes \mathbf{n}^K)\,\mathrm{d}s\,\mathrm{d}t \\ = \int_{t_i^K}^{t_i^K + \Delta t_i^K} \int_K \widetilde{\mathbf{s}}(\widetilde{\mathbf{q}}_h) \mathbf{v}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t,\end{aligned} \tag{1.18}$$

for all $\mathbf{v}_h \in \mathbf{V}_h$, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$.

The original arbitrary derivative finite volume and discontinuous Galerkin algorithms introduce the space-time predictor $\widetilde{\mathbf{q}}_h$ as a Taylor expansion in space and time around the space-time point $(\mathbf{x}_K, t_i^K)$ in every cell $K \in \mathcal{T}_h$, $i = \{0, 1, \ldots, i_{\max}^K\}$, where $\mathbf{x}_K$ denotes the cell center, and then apply a Cauchy-Kowalewsky procedure (CK) to compute time derivatives in terms of the spatial derivatives. This procedure can be implemented in a generic fashion for linear problems but becomes rather cumbersome for nonlinear problems. For nonlinear problems, the time derivatives must be computed by hand.

The CK approach is regarded as the fastest approach for linear hyperbolic equations; see [4] for a detailed discussion of this approach and a comparison to other ADER approaches utilised in the literature.

In this work, we will consider the rather novel local space-time discontinuous Galerkin predictor which was introduced in [1]. This approach is considered as less efficient than existing CK implementations for linear hyperbolic problems; see again [4]. However, it has proven to be a very general and robust method for both, linear and nonlinear problems.

In order to compute the local space-time discontinuous Galerkin predictor, we start again from equation (1.1a). However this time, we directly replace $\mathbf{u}$ by a discrete space-time ansatz function $\widetilde{\mathbf{q}}_h \in \widetilde{\mathbf{V}}_h$, and test with discrete space-time test functions $\widetilde{\mathbf{v}}_h \in \widetilde{\mathbf{V}}_h$.

Integration over a space-time cell $K \times (t_i^K, t_i^K + \Delta t_i^K)$ then leads to:

$$\begin{aligned}\int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \frac{\partial}{\partial t}\widetilde{\mathbf{q}}_h \, \widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t - \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{F}}(\widetilde{\mathbf{q}}_h) : \nabla\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ + \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_{\partial K} \widetilde{\mathbf{G}}(\widetilde{\mathbf{q}}_h^+, \widetilde{\mathbf{q}}_h^-) : (\widetilde{\mathbf{v}}_h \otimes \mathbf{n}^K) \,\mathrm{d}s\,\mathrm{d}t \\ = \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{s}}(\widetilde{\mathbf{q}}_h)\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t, \end{aligned} \tag{1.19}$$

for all $\widetilde{\mathbf{v}}_h \in \widetilde{\mathbf{V}}_h$, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$. We further assume that inflow and outflow are balanced in the weak sense in every space-time cell $K \times (t_i^K, t_i^K + \Delta t_i^K)$. We thus obtain:

$$\begin{aligned}\int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \frac{\partial}{\partial t}\widetilde{\mathbf{q}}_h \, \widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t - \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{F}}(\widetilde{\mathbf{q}}_h) : \nabla\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ = \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{s}}(\widetilde{\mathbf{q}}_h)\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t, \end{aligned} \tag{1.20}$$

for all $\widetilde{\mathbf{v}}_h \in \widetilde{\mathbf{V}}_h$, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$. Integrating the first term in time by parts finally leads to:

$$\begin{aligned}\int_K \widetilde{\mathbf{q}}_h(t_i^K + \Delta t_i^K)\, \widetilde{\mathbf{v}}_h(t_i^K + \Delta t_i^K) \,\mathrm{d}\mathbf{x} - \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{q}}_h \frac{\partial}{\partial t}\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ = \int_K \widetilde{\mathbf{q}}_h(t_i^K)\, \widetilde{\mathbf{v}}_h(t_i^K)\mathrm{d}\mathbf{x} \\ + \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{F}}(\widetilde{\mathbf{q}}_h) : \nabla\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t \\ + \int_{t_i^K}^{t_i^K+\Delta t_i^K} \int_K \widetilde{\mathbf{s}}(\widetilde{\mathbf{q}}_h)\widetilde{\mathbf{v}}_h \,\mathrm{d}\mathbf{x}\,\mathrm{d}t, \end{aligned} \tag{1.21}$$

for all $\widetilde{\mathbf{v}}_h \in \widetilde{\mathbf{V}}_h$, for every $K \in \mathcal{T}_h$ and every $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$.

The space-time predictor computation thus requires solving the *locally implicit* problem (1.21). If we specify an initial guess for the space-time predictor $\widetilde{\mathbf{q}}_i^K$ at time t_i^K, $K \in \mathcal{T}_h$, $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$, and further choose a basis of the spaces $\mathbf{V}_h$ and $\widetilde{\mathbf{V}}_h$. Then, solving problem (1.21) results in solving a fixed-point problem for every cell $K \in \mathcal{T}_h$ in every time step $i \in \{0, 1, \ldots, i_{\max}^K - 1\}$; see [3] and App. A.7 for further details. A straightforward choice of the initial guess is the previous solution [3], i.e., we set

$$\widetilde{\mathbf{q}}_h^K(t_i^K) = \widetilde{\mathbf{u}}^K(t_i^K), \qquad K \in \mathcal{T}_h, i \in \{0, 1, \ldots, i_{\max}^K - 1\}. \tag{1.22}$$

Employing a Picard iteration for solving the fixed-point problem leads to fast convergence to the of the local spacetime predictor solution, see [3] and the references therein.

Lastly, we want to emphasise that we did not make assumptions on the basis functions of the two spaces $\mathbf{V}_h$ and $\widetilde{\mathbf{V}}_h$ at any point in this chapter. The local space-time ADER-DG formulation was derived within a completely abstract framework up to the employed tensor product elements $Q_N(K)^{N_{\text{var}}}$ and $Q_N(K \times (t_i^K, t_i^K + \Delta t_i^K))^{N_{\text{var}}}$ which can be readily replaced by other finite element spaces. A matrixfree, quadraturefree discretisation of ADER-DG discretisation is derived in Appendix A. The discretisation utilises Lagrange basis polynomials that use the nodes of a Gauss-Legendre quadrature as interpolation points. In Appendix B, we further derive prolongation and restriction operators that are required in order to use the aforementioned discretisation on adaptive meshes. On adaptive meshes, prolongation operators are used to represent coarse level degrees of freedom on finer levels of the mesh while restriction operators are used to represent fine level degrees of freedom on coarser levels of the mesh. We further show how to construct restriction and prolongation operators in a recursive fashion.

1.7. Error estimates

To the best of our knowledge, no a-priori error estimates for the ADER-DG method for particular hyperbolic conservation laws have been published to this date. The high-order approximation ability of the method is usually verified by numerical experiments.

For hyperbolic conservation laws that admit a sufficiently regular solution $\widetilde{\mathbf{u}}$, the convergence of the ADER-DG method has been numerically verified to be of optimal order, i.e., it has been demonstrated that

$$\|\widetilde{\mathbf{u}}_h(t_i) - \widetilde{\mathbf{u}}(t_i)\|_{L^2(\Omega)} \leq C\,(h^{N+1} + \Delta t_{i-1}^{N+1}) \tag{1.23}$$

for these problems, with a constant C independent of the mesh size h, the time step size Δt_{i-1}, and the approximation order N. The time discrete numerical solution is above again denoted by $\widetilde{\mathbf{u}}_h(t_i)$. In order for the above bound to be valid, we further require that a family of shape-regular meshes is employed, and that the time steps t_i, $i = 1, 2, \ldots$, are chosen identically on every cell of the mesh and further do not harm the CFL condition.

See e.g. [4] for a verification of the convergence rate for the linear advection equation, and for the nonlinear, compressible Euler equations; see e.g. [2] for a verification for elastic waves.

1.8. Discontinuous solutions and limiters

If we solve hyperbolic problems that are subject to discontinuous initial conditions ("shocks") using a higher order discontinuous Galerkin method, we observe the well-known Gibbs phenomenon; see e.g. [5]. For nonlinear problems, such shocks can further develop over time, even from smooth initial conditions; see e.g. [5].

Shocks have three unfortunate effects on the numerically computed solution[5]:

- The reduction to first order pointwise accuracy away from the point of discontinuity.
- The loss of pointwise convergence at the point of discontinuity
- The introduction of artificial and persistent oscillations around the point of discontinuity.

While the first two effects can be tackled in principle by adaptive mesh refinement around the discontinuity, the last effect might lead to positively signed quantities such as the mass density attaining non-physical negative values. Large non-physical oscillations also pose a problem in multi physics applications. Here, coupled processes might become accelerated or slowed down locally or the corresponding solvers might even crash due to negatively signed coefficients obtained as solution from a hyperbolic PDE solver.

An important component of hyperbolic PDE solvers that deal with shocks or nonlinear solvers is thus a limiting of non-physical oscillations. An overview of different limiting approaches for the discontinuous Galerkin method and the application of a a-posteriori subcell limiting algorithm to the ADER-DG method can be found in [3].

A combination of the local space-time ADER-DG method with adaptive mesh refinement and a-posteriori subcell limiting then leads to a numerical algorithm with very desirable approximation properties even for problems that consider discontinuous initial data or shock formation [8].

A. A matrixfree, quadraturefree local space-time ADER-DG discretisation

A.1. Introduction

This chapter considers a matrixfree, quadrature discretisation of the ADER-DG scheme discussed in Ch. 1. The chapter is organised as follows: After a short introduction to mappings between a reference cell and mesh cells in Sec. A.2, we introduce the nodal basis polynomials which are utilised to discretise the discrete variational problems (1.18) and (1.21) in Sec. A.3. The matrix and quadrature free discretisations of (1.21) and (1.18) are then derived in Sec. A.7, and Sec. A.9, respectively.

A.2. Mappings

In this section, we introduce mappings between the mesh cells $K \in \mathcal{T}_h$ and a reference cell. Mappings allow us to treat integrals over mesh cells with non-uniform extent in an uniform manner. To ease the presentation, we only consider the two-dimensional case here.

Let us define the *reference cell* $\hat{K} = (0,1)^2$ and let us denote by $\hat{\mathbf{x}} = (\hat{x}, \hat{y})^{\mathrm{T}}$ a point belonging to $\hat{K}$. Let $K \in \mathcal{T}_h$ denote a nondegenerate quadrilateral cell with center $\mathbf{P}_0$ and cell size $(\Delta x, \Delta y)^T$.

We can express K according to

$$K = \boldsymbol{\mathcal{F}}_K(\hat{K}),$$

where we define the mapping $\boldsymbol{\mathcal{F}}_K : \hat{K} \cup \partial\hat{K} \to K \cup \partial K$ as

$$\mathbf{x} = \boldsymbol{\mathcal{F}}_K(\hat{\mathbf{x}}) = \begin{pmatrix} \mathcal{F}_{K,x}(\hat{\mathbf{x}}) \\ \mathcal{F}_{K,y}(\hat{\mathbf{x}}) \end{pmatrix} = \mathbf{P}_0 + \begin{pmatrix} \Delta x & 0 \\ 0 & \Delta y \end{pmatrix} \begin{pmatrix} \hat{x} - 0.5 \\ \hat{y} - 0.5 \end{pmatrix} \tag{A.1}$$

We will further denote the inverse mapping by $\boldsymbol{\mathcal{F}}_K^{-1} \colon K \cup \partial K \to \hat{K} \cup \partial\hat{K}$, which can be easily constructed for the considered affine mesh cells.

The considered quadrilateral cells have a constant Jacobian matrix, i.e.,

$$\mathrm{D}\boldsymbol{\mathcal{F}}_K(\hat{\mathbf{x}}) = \begin{pmatrix} \frac{\partial \mathcal{F}_{K,x}}{\partial \hat{x}} & \frac{\partial \mathcal{F}_{K,x}}{\partial \hat{y}} \\ \frac{\partial \mathcal{F}_{K,y}}{\partial \hat{x}} & \frac{\partial \mathcal{F}_{K,y}}{\partial \hat{y}} \end{pmatrix} (\hat{\mathbf{x}}) = \begin{pmatrix} \Delta x & 0 \\ 0 & \Delta y \end{pmatrix},$$

and thus also a constant Jacobian determinant:

$$\begin{aligned} J_K(\hat{\mathbf{x}}) &= \det\left(\mathrm{D}\boldsymbol{\mathcal{F}}_K\right)(\hat{\mathbf{x}}) \\ &= \Delta x\,\Delta y. \end{aligned} \tag{A.2}$$

Let $\hat{f}$ be a sufficiently regular function on the reference cell $\hat{K}$. Further let f be sufficiently regular on K and such that

$$f(\mathbf{x}) = (\hat{f} \circ \boldsymbol{\mathcal{F}}_K^{-1})(\mathbf{x}) = \hat{f}(\hat{\mathbf{x}}), \qquad \hat{\mathbf{x}} \in \hat{K},\, \mathbf{x} = \boldsymbol{\mathcal{F}}_K(\hat{\mathbf{x}}) \in K. \tag{A.3}$$

Then, it holds that

$$\nabla f(\mathbf{x}) = \left(\mathrm{D}\boldsymbol{\mathcal{F}}_K^{-\mathrm{T}} \cdot \hat{\nabla} \hat{f}\right)(\hat{\mathbf{x}}) = \begin{pmatrix} \frac{1}{\Delta x} & 0 \\ 0 & \frac{1}{\Delta y} \end{pmatrix} \hat{\nabla} \hat{f}(\hat{\mathbf{x}}) \tag{A.4}$$

where

$$\hat{\nabla} = \left(\frac{\partial}{\partial \hat{x}}, \frac{\partial}{\partial \hat{y}}\right)^{\mathrm{T}}$$

denotes the gradient with respect to the reference coordinates. This follows from (A.2.1) and using the chain rule. A similar identity can be derived for vector-valued functions.

We can also express integrals over the volume of cell K with respect to the reference coordinates and the invertible mapping $\boldsymbol{\mathcal{F}}_K$. Let $\hat{f}$ and f be defined as in (A.2.1), we have

$$\begin{aligned} \int_K f(\mathbf{x})\,\mathrm{d}\mathbf{x} &= \int_{\hat{K}} \hat{f}(\hat{\mathbf{x}})\,|J_K(\hat{\mathbf{x}})|\,\mathrm{d}\hat{\mathbf{x}} = \int_0^1 \int_0^1 \hat{f}(\hat{x},\hat{y})\,|J_K(\hat{x},\hat{y})|\,\mathrm{d}\hat{x}\,\mathrm{d}\hat{y} \\ &\overset{(A.2)}{=} J_K \int_0^1 \int_0^1 \hat{f}(\hat{x},\hat{y})\;\mathrm{d}\hat{x}\,\mathrm{d}\hat{y} \\ &= \Delta x\,\Delta y \int_0^1 \int_0^1 \hat{f}(\hat{x},\hat{y})\;\mathrm{d}\hat{x}\,\mathrm{d}\hat{y}. \end{aligned}$$

Let us indicate the faces of the reference cell using a tuple (ξ, f), $\xi \in \{0, 1, \ldots, d-1\}$, $f \in \{0, 1\}$, where the first index corresponds to the fixed coordinate direction:

$$\begin{aligned} \hat{e}^{1,0} &= \{\hat{\mathbf{x}} \in \hat{K} : \hat{x} = 0\}, \qquad \hat{e}^{1,1} = \{\hat{\mathbf{x}} \in \hat{K} : \hat{y} = 0\}, \\ \hat{e}^{2,0} &= \{\hat{\mathbf{x}} \in \hat{K} : \hat{x} = 1\}, \qquad \hat{e}^{2,1} = \{\hat{\mathbf{x}} \in \hat{K} : \hat{y} = 1\}, \end{aligned}$$

Then, we can define the four faces belonging to the boundary of K according to:

$$e^{1,0} = \boldsymbol{\mathcal{F}}_K(\hat{e}^{1,0}), \qquad e^{1,1} = \boldsymbol{\mathcal{F}}_K(\hat{e}^{1,1}), \qquad e^{2,0} = \boldsymbol{\mathcal{F}}_K(\hat{e}^{2,0}), \qquad e^{2,1} = \boldsymbol{\mathcal{F}}_K(\hat{e}^{2,1}).$$

Now we can express integrals over faces belonging to the boundary of K with respect to the reference coordinates and the mapping $\boldsymbol{\mathcal{F}}_K$. Let $\hat{f}$ and f be defined as in (A.2.1), we have

$$\int_{e^{1,0}} f(\mathbf{x})\,\mathrm{d}s(\mathbf{x}) = \int_0^1 \hat{f}(0,\hat{y}) \left|\frac{\partial \boldsymbol{\mathcal{F}}_K(0,\hat{y})}{\partial\,\hat{y}}\right| \mathrm{d}\hat{y}$$
$$\overset{(A.1)}{=} \Delta y \int_0^1 \hat{f}(0,\hat{y})\ \mathrm{d}\hat{y}, \tag{A.5}$$
$$\int_{e^{1,1}} f(\mathbf{x})\,\mathrm{d}s(\mathbf{x}) = \int_0^1 \hat{f}(1,\hat{y}) \left|\frac{\partial \boldsymbol{\mathcal{F}}_K(1,\hat{y})}{\partial\,\hat{y}}\right| \mathrm{d}\hat{y}$$
$$= \Delta y \int_0^1 \hat{f}(1,\hat{y})\ \mathrm{d}\hat{y}, \tag{A.6}$$
$$\int_{e^{2,0}} f(\mathbf{x})\,\mathrm{d}s(\mathbf{x}) = \int_0^1 \hat{f}(\hat{x},0) \left|\frac{\partial \boldsymbol{\mathcal{F}}_K(\hat{x},0)}{\partial\,\hat{x}}\right| \mathrm{d}\hat{x}$$
$$\overset{(A.1)}{=} \Delta x \int_0^1 \hat{f}(\hat{x},0)\ \mathrm{d}\hat{x}, \tag{A.7}$$
$$\int_{e^{2,1}} f(\mathbf{x})\,\mathrm{d}s(\mathbf{x}) = \int_0^1 \hat{f}(\hat{x},1) \left|\frac{\partial \boldsymbol{\mathcal{F}}_K(\hat{x},1)}{\partial\,\hat{x}}\right| \mathrm{d}\hat{x}$$
$$= \Delta x \int_0^1 \hat{f}(\hat{x},1)\ \mathrm{d}\hat{x}, \tag{A.8}$$

A.2.1. Mappings in arbitrary dimensions

All concepts considered in the previous section naturally extend to higher dimensions. Let us consider in this sections a reference cell $\hat{K} \in (0,1)^d$, $d \geq 1$.

For the Jacobian determinant, we obtain in this case:

$$J_K = \prod_{\xi=1}^{d} \Delta x_\xi. \tag{A.9}$$

For a volume integral, we obtain:

$$\int_K f(\mathbf{x})\,\mathrm{d}\mathbf{x} = J_K \int_{\hat{K}} f(\hat{\mathbf{x}})\,\mathrm{d}\mathbf{x}$$
$$= \prod_{\xi=1}^{d} \Delta x_\xi \int_{\hat{K}} \hat{f}(\hat{\mathbf{x}})\,\mathrm{d}\hat{\mathbf{x}} \tag{A.10}$$

And for an integral over a face $e^{\xi,f}$, $\xi \in \{1,2,\ldots,d\}$, $f \in \{0,1\}$, we obtain:

$$\int_{e^{\xi,f}} f(\mathbf{x})\,\mathrm{d}s(\mathbf{x}) = \prod_{\zeta=1,\zeta\neq\xi}^{d} \Delta x_\zeta \int_{\hat{e}^{\xi,f}} \hat{f}(\hat{\mathbf{x}})\,\mathrm{d}\hat{s}(\hat{\mathbf{x}})$$
$$= \frac{J_K}{\Delta x_\xi} \int_{\hat{e}^{\xi,f}} \hat{f}(\hat{\mathbf{x}})\,\mathrm{d}\hat{s}(\hat{\mathbf{x}}) \tag{A.11}$$

Let $\hat{f}$ be a sufficiently regular function on the reference cell $\hat{K}$, and let f be sufficiently regular on K and such that

$$f(\mathbf{x}) = (\hat{f} \circ \boldsymbol{\mathcal{F}}_K^{-1})(\mathbf{x}) = \hat{f}(\hat{\mathbf{x}}), \qquad \hat{\mathbf{x}} \in \hat{K},\, \mathbf{x} = \boldsymbol{\mathcal{F}}_K(\hat{\mathbf{x}}) \in K.$$

Then, partial derivatives of f can be expressed with respect to the reference coordinates according to:

$$\frac{\partial f(\mathbf{x})}{\partial x_\xi} = \frac{1}{\Delta x_\xi} \frac{\partial \hat{f}(\hat{\mathbf{x}})}{\partial \hat{x}_\xi}, \tag{A.12}$$

where $\xi \in \{1, 2, \ldots, d\}$.

A.2.2. Space-time mappings

We can express the space-time cell $K \times (t_0, t_0 + \Delta t)$ according to

$$K \times (t_0, t_0 + \Delta t) = \boldsymbol{\mathcal{F}}_{K,t} \hat{K} \times (0, 1),$$

where we have introduced the space-time mapping

$$\boldsymbol{\mathcal{F}}_{K,t} \colon\ (\hat{K} \cup \partial\hat{K}) \times [0, 1] \to (K \cup \partial K) \times [t^K, t^K + \Delta t^K] \tag{A.13}$$

which is defined according to:

$$(\mathbf{x}, t) = \boldsymbol{\mathcal{F}}_{K,t}(\hat{\mathbf{x}}, \hat{t}) = (\mathbf{P}_0, t_0) + \begin{pmatrix} \Delta x & 0 & 0 \\ 0 & \Delta y & 0 \\ 0 & 0 & \Delta t \end{pmatrix} \begin{pmatrix} \hat{x} - 0.5 \\ \hat{y} - 0.5 \\ \hat{t} \end{pmatrix}. \tag{A.14}$$

A.3. Basis functions

A.3.1. Preliminaries

Let be f be a sufficiently regular univariate function such that it can be approximated by a polynomial of order N, i.e., the leading approximation error term is of order $\mathcal{O}(N+1)$.

Given a set of support point and function value pairs $\{(x_i, f(x_i))\}_{0 \le i \le N}$, the corresponding interpolation polynomial in the Lagrange form can be constructed according to:

$$f_N(x) = \sum_{i=0}^{N} f(x_i)\, L_i(x), \tag{A.15}$$

where the Lagrange basis polynomials are defined as

$$L_i(x) = \left(\prod_{\substack{0 \le l \le N \\ l \ne i}} \frac{x - x_l}{x_i - x_l} \right), \qquad i = 0, \ldots, N,$$

Since we exclude the $(x - x_i)$ term in the product, the basis functions have the property

$$L_i(x_j) = \delta_{ij} = \begin{cases} 1 & i = j, \\ 0 & \text{else.} \end{cases} \qquad i, j = 0, \ldots, N, \tag{A.16}$$

Let the interval $I \in \mathbb{R}$ be open or closed. With respect to the Gauss-Legendre quadrature of degree $2\,(N+1) - 1$, we define a discrete scalar product $\langle \cdot, \cdot \rangle_{L^2(I)}$ according to:

$$\langle f, g \rangle_{L^2(I)} = \sum_{n=0}^{N} w_n \, f(x_n) \, g(x_n), \tag{A.17}$$

where x_n denotes a quadrature node and w_n denotes a quadrature weight, $n = 0, 1, \ldots, N$.

Lemma A.1: *Let us denote by $\{\varphi_i\}_{i=0,1,\ldots,N}$ a Lagrange basis utilising basis polynomials located at the nodes of a Gauss-Legendre quadrature of degree $2\,(N+1)-1$. Let the nodes lie in the interval I. Then, $\{\varphi_i\}_{i=0,1,\ldots,N}$ is a orthogonal basis with respect to the $(\cdot, \cdot)_{L^2(I)}$ scalar product. Furthermore, it is a orthogonal basis with respect to all Gauss-Legendre quadratures with at least a degree of $2\,(N+1) - 1$ that are used to evaluate $(\cdot, \cdot)_{L^2(I)}$.*

Proof: We only need to proof that the basis functions are orthogonal with respect to the Gauss-Legendre quadrature of degree $2\,(N+1) - 1$. The rest follows from the fact that the Gauss-Legendre quadrature of degree $2\,(N+1) - 1$ is exact for all polynomials of degree $2\,(N+1) - 1 > 2\,N$, where $2\,N$ is the maximum total degree of the product of two basis polynomials.

Select two basis functions φ_i and φ_j,$i, j = 0, 1, \ldots, N$. Since both functions are Lagrange polynomials, we have

$$\varphi_i(x_j) = \delta_{ij} = \begin{cases} 1 & i = j, \\ 0 & \text{else,} \end{cases} \tag{A.18}$$

where x_j denotes a support point, and δ_{ij} denotes the Kronecker delta. An analogous condition holds for φ_j.

We want to evaluate the scalar product $(\cdot, \cdot)_{L^2(I)}$ by using the Gauss-Legendre quadrature of degree $2\,(N+1) - 1$. We obtain:

$$\begin{aligned} (\varphi_i, \, \varphi_j)_{L^2(I)} &= \langle \varphi_i, \, \varphi_j \rangle_{L^2(I)} = \sum_{n=0}^{N} w_n \, \varphi_i(x_n) \, \varphi_j(x_n) \\ &\overset{(A.16)}{=} \sum_{n=0}^{N} w_n \, \delta_{in} \, \delta_{jn} = w_i \, \delta_{ij}. \end{aligned} \tag{A.19}$$

□

A.3.2. Definition of the basis functions

Let us introduce the space of polynomials of order at most N with support in $(0,\,1)$:

$$Q_N((0,\,1)) = \operatorname{span}\left\{\hat{x}^n \;\text{ such that } 0 \leq n \leq N \text{ and } \hat{x} \in (0,\,1)\right\}.$$

As a basis of the space $Q_N((0,1))$, we use a set of $N+1$ Lagrange basis polynomials. The support points of the basis polynomials are chosen such that they coincide with the nodes of a Gauss-Legendre quadrature of degree $2\,(N+1)-1$ which considers the open interval $(0,\,1)$ as domain of integration. We denote the scalar-valued univariate reference basis functions by $\hat{\varphi}_n(\hat{x})$, $n = 0, 1, \ldots, N$, and further introduce scalar-valued univariate basis functions on the affine mesh cells $K \in \mathcal{T}_h$:

$$\varphi^K_{n,\xi}(x_\xi) = \begin{cases} (\hat{\varphi}_n \circ \mathcal{F}^{-1}_{K,\xi})(x_\xi) & \mathbf{x} \in K, \\ 0 & \text{else.} \end{cases} \tag{A.20}$$

for every coordinate direction $\xi \in \{1, 2, \ldots, d\}$.

Let us further define scalar-valued multivariate polynomials on the reference cell $\hat{K} = (0,1)^d$, and on each mesh cell $K \in \mathcal{T}_h$ according to:

$$\hat{\phi}_n(\hat{\mathbf{x}}) = \prod_{\xi=1}^{d} \hat{\varphi}_{n_\xi}(\hat{x}_\xi), \tag{A.21}$$

$$\phi^K_n(\mathbf{x}) = \begin{cases} (\hat{\phi}_n \circ \boldsymbol{\mathcal{F}}^{-1}_K)(\mathbf{x}) & \mathbf{x} \in K, \\ 0 & \text{else,} \end{cases} \tag{A.22}$$

for $n_\xi = 0, 1, \ldots, N$, $n_\xi \in \{1, \ldots, d\}$, and a linearised index $n = 0, 1, \ldots, (N+1)^d - 1$ that is constructed according to:

$$n = \sum_{\xi=1}^{d} N_\xi\, n_\xi,$$

using the one-dimensional basis indices $n_\xi = 0, 1, \ldots, N$ and some strides $N_\xi \in \{0, N+1\}$, $\xi \in \{1, 2, \ldots, d\}$ that define an unique order of the polynomials.

Notice that (A.22) implies that

$$\mathbf{x} = \boldsymbol{\mathcal{F}}_K \hat{\mathbf{x}} \qquad \Leftrightarrow \qquad \phi^K_n(\mathbf{x}) = \hat{\phi}_n(\hat{\mathbf{x}}),$$

for $\mathbf{x} \in K$, $\hat{\mathbf{x}} \in \hat{K}$, and $n = 0, 1, \ldots, (N+1)^d - 1$.

The corresponding d-dimensional support points (d-dimensional Gauss-Legendre quadrature nodes) on the reference cell are constructed as tensor products of the one-dimensional support points:

$$\hat{\mathbf{x}}_n = (\hat{x}_{n,1}, \ldots, \hat{x}_{n,d})^{\mathrm{T}},$$

with $n = 0, 1, \ldots, (N+1)^d - 1$.

One can show that the polynomials $\{\hat{\phi}_n(\hat{\mathbf{x}})\}_{n=0,1,\ldots,(N+1)^d-1}$ form a basis of the space

$$Q_N(\hat{K}) = \text{span}\left\{\Pi_{i=1}^d \hat{x}_i^{n_i} \text{ such that } 0 \le n_i \le N,\, i = 1,\ldots,d, \text{ and } \hat{\mathbf{x}} \in \hat{K}\right\}$$

The d-dimensional support points for the multivariate basis functions on the mesh cells $K \in \mathcal{T}_h$ are then constructed by means of the mapping $\boldsymbol{\mathcal{F}}_K$:

$$(x_{n,1},\ldots,x_{n,d})^{\mathrm{T}} = \boldsymbol{\mathcal{F}}_K \hat{\mathbf{x}}_n,$$

with $n = 0, 1 \ldots, (N+1)^d - 1$ denoting the global index.

Since we only consider affine reference cell to mesh cell mappings, we can further show that the polynomials $\{\phi_n^K(\mathbf{x})\}_{n=0,1,\ldots,(N+1)^d-1}$ form a basis of the space

$$Q_N(K) = \text{span}\left\{\Pi_{i=1}^d x_i^{n_i} \text{ such that } 0 \le n_i \le N,\, i = 1,\ldots,d, \text{ and } \mathbf{x} \in K\right\}.$$

Let us introduce another index $v = 0, 1, \ldots, N_{\text{var}} - 1$ that numbers the variables. We construct the basis $\{\boldsymbol{\phi}_n^{K:v}(\mathbf{x})\}_{v=0,1,\ldots,N_{\text{var}}-1:\, n=0,1,\ldots,(N+1)^d-1}$ of the space $Q_N(K)^{N_{\text{var}}}$ according to:

$$\begin{aligned}\{\boldsymbol{\phi}_n^{K:v}(\mathbf{x})\}_{v=0,1,\ldots,N_{\text{var}}-1:\, n=0,1,\ldots,(N+1)^d-1} = \\ \{(1,\,0,\,\ldots,0)_{N_{\text{var}}}^{\mathrm{T}}, (0,\,1,\,\ldots,0)_{N_{\text{var}}}^{\mathrm{T}}, \ldots,, (0,\,0,\,\ldots,1)_{N_{\text{var}}}^{\mathrm{T}}\} \otimes \\ \{\phi_n^K(\mathbf{x})\}_{n=0,1,\ldots,(N+1)^d-1}\end{aligned}$$

Lastly, let us introduce the space-time basis polynomials

$$\begin{aligned}\hat{\theta}_{l:n}(\hat{\mathbf{x}},\hat{t}) &= \hat{\varphi}_l(\hat{t})\,\hat{\phi}_{l:n}(\hat{\mathbf{x}}) \\ \theta_{l:n}^K(\mathbf{x},t) &= \begin{cases}(\hat{\theta}_{l:n} \circ \boldsymbol{\mathcal{F}}_{K,t}^{-1})(\mathbf{x},t) & (\mathbf{x},t) \in K \times (t^K, t^K + \Delta t^K) \\ 0 & \text{else,}\end{cases}\end{aligned}$$

with $l = 0, 1, \ldots, N$, and $n = 0, 1, \ldots, (N+1)^d - 1$. Here, t^K and Δt^K denote a time stamp and a time step increment, respectively, associated with the cell K.

The corresponding $(d+1)$-dimensional support points ($(d+1)$-dimensional Gauss-Legendre quadrature nodes) on the reference cell are constructed as tensor products of the one-dimensional support points:

$$(\hat{\mathbf{x}}_n, \hat{t}_l) = (\hat{x}_{n,1},\ldots,\hat{x}_{n,d},\hat{t}_l)^{\mathrm{T}},$$

with $n \in \{0, 1, \ldots, (N+1)^d - 1\}$ denoting the index of the spatial basis and with $l \in \{0, 1, \ldots, N\}$ denoting the index of the temporal basis.

We can now construct a basis $\{\boldsymbol{\theta}_{l:n}^{K:v}(\mathbf{x})\}_{v=0,1,\ldots,N_{\text{var}}-1:\, l=0,1,\ldots,N:\, n=0,1,\ldots,(N+1)^d-1}$ of the space $Q_N(K \times (t^K, t^K + \Delta t^K))^{N_{\text{var}}}$ according to:

$$\begin{aligned}\{\boldsymbol{\theta}_{l:n}^{K:v}(\mathbf{x})\}_{v=0,1,\ldots,N_{\text{var}}-1:\, l=0,1,\ldots,N:\, n=0,1,\ldots,(N+1)^d-1} = \\ \{(1,\,0,\,\ldots,0)_{N_{\text{var}}}^{\mathrm{T}}, (0,\,1,\,\ldots,0)_{N_{\text{var}}}^{\mathrm{T}}, \ldots,, (0,\,0,\,\ldots,1)_{N_{\text{var}}}^{\mathrm{T}}\} \\ \otimes \{\theta_{l:n}^K(\mathbf{x})\}_{l=0,1,\ldots,N:\, n=0,1,\ldots,(N+1)^d-1}.\end{aligned}$$

A.3.3. Properties of the basis functions

Let us number the d-dimensional Gauss-Legendre quadrature weights on the reference cell with the same linearised index $n = 0, 1, \ldots, (N+1)^d - 1$ that we used to number the basis functions and the corresponding d-dimensional spatial support points, i.e.,

$$w_n = \prod_{\xi=1} w_{n_\xi}, \tag{A.23}$$

with $n_\xi = 0, 1, \ldots, N$, $n_\xi \in \{1, \ldots, d\}$.

Below, let $l, l' \in \{0, 1, \ldots, N\}$, $n, n' \in \{0, 1, \ldots, (N+1)^d - 1\}$, $K, K' \in \mathcal{T}_h$ and $v, v' \in \{0, 1, \ldots, N_{\text{var}} - 1\}$. In the following, we summarise the properties of the basis polynomials that have been introduced in the previous section.

- Lagrange basis property (reference cell):

$$\hat{\varphi}_l(\hat{x}_{l'}) \overset{(A.16)}{=} \delta_{ll'}, \tag{A.24}$$

$$\hat{\phi}_n(\hat{\mathbf{x}}_{n'}) = \prod_{\xi=1}^{d} \hat{\varphi}_{n_\xi}(\hat{x}_{m,\xi}) = \prod_{\xi=1}^{d} \delta_{n_\xi n'_\xi} = \delta_{nn'}, \tag{A.25}$$

$$\hat{\theta}_{l:n}(\hat{\mathbf{x}}_{n'}, \hat{t}_{l'}) = \hat{\varphi}_l(\hat{t}_{l'})\, \hat{\phi}_{n'}(\hat{\mathbf{x}}_{n'}) = \delta_{ll'}\, \delta_{nn'}. \tag{A.26}$$

- Sampling property (reference cell):

$$\langle \hat{f},\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))} \overset{(A.17)}{=} w_{l'}\, \hat{f}(\hat{x}_{l'}), \tag{A.27}$$

$$\begin{aligned} \langle \hat{f},\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} &= \prod_{\xi=1}^{d} \langle \hat{f},\, \hat{\varphi}_{n'_\xi} \rangle_{L^2((0,1))} \\ &= w_n\, \hat{f}(\hat{\mathbf{x}}_n), \end{aligned} \tag{A.28}$$

$$\begin{aligned} \langle \hat{f},\, \hat{\theta}_{l':n'} \rangle_{L^2(\hat{K}\times(0,1))} &= \langle \hat{f},\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))}\, \langle \hat{f},\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\ &= w_l\, w_n\, \hat{f}(\hat{\mathbf{x}}_n, t_l), \end{aligned} \tag{A.29}$$

- Sampling property (mesh cell):

$$\langle f,\, \varphi_{l'} \rangle_{L^2((t^K, t^K+\Delta t^K))} \overset{(A.10)}{=} \Delta t^K\, \langle \hat{f},\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))} \tag{A.30}$$

$$= \Delta t^K\, w_{l'}\, f(t_{l'}), \tag{A.31}$$

$$\begin{aligned} \langle f,\, \phi_{n'} \rangle_{L^2(K)} &\overset{(A.10)}{=} J_K\, \langle \hat{f},\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\ &= J_K\, w_{n'}\, f(\mathbf{x}_{n'}), \end{aligned} \tag{A.32}$$

$$\begin{aligned} \langle f,\, \theta_{l':n'} \rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} &= \langle f,\, \varphi_l \rangle_{L^2((t^K, t^K+\Delta t^K))}\, \langle f,\, \phi_n \rangle_{L^2(K)} \\ &= \Delta t^K\, J_K\, w_{l'}\, w_{n'}\, f(\mathbf{x}_{n'}, t_{l'}), \end{aligned} \tag{A.33}$$

- Discrete orthogonality (reference cell)

$$\langle \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))} \overset{(A.19)}{=} w_l\, \delta_{ll'}, \tag{A.34}$$

$$\begin{aligned} \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} &= \prod_{\xi=1}^{d} \langle \hat{\varphi}_{n_\xi},\, \hat{\varphi}_{n'_\xi} \rangle_{L^2((0,1))} = \prod_{\xi=1}^{d} w_{n_\xi} \delta_{n_\xi n'_\xi} \\ &= w_n\, \delta_{nn'}, \end{aligned} \tag{A.35}$$

$$\begin{aligned} \langle \hat{\theta}_{l:n},\, \hat{\theta}_{l':n'} \rangle_{L^2(\hat{K}\times(0,1))} &= \langle \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))}\, \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\ &= w_l\, w_n\, \delta_{ll'}\, \delta_{nn'}. \end{aligned} \tag{A.36}$$

- Discrete orthogonality (mesh cell)

$$\langle \varphi_l^K,\, \varphi_{l'}^K \rangle_{L^2([t^K, t^K+\Delta t^K))} \overset{(A.20)}{=} \Delta t^K\, \langle \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))} = \Delta t^K\, w_l\, \delta_{ll'}, \tag{A.37}$$

$$\begin{aligned} \langle \phi_n^K,\, \phi_{n'}^K \rangle_{L^2(K)} &= J_K\, \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\ &= J_K\, w_n\, \delta_{nn'}, \end{aligned} \tag{A.38}$$

$$\begin{aligned} \langle \theta_{l:n}^K,\, \theta_{l':n'}^K \rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} &= \Delta t^K\, \langle \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))}\, J_K\, \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\ &= \Delta t^K\, J_K\, w_l\, w_n\, \delta_{ll'}\, \delta_{nn'}. \end{aligned} \tag{A.39}$$

- Compact support:

$$\begin{aligned} \langle \phi_n^K,\, \phi_{n'}^{K'} \rangle_{L^2(K)} &= \langle \phi_n^K,\, \phi_{n'}^K \rangle_{L^2(K)}\, \delta_{KK'} \\ &= J_K\, w_n\, \delta_{KK'}\, \delta_{nn'}, \end{aligned} \tag{A.40}$$

$$\begin{aligned} \langle \theta_{l:n}^K,\, \theta_{l':n'}^{K'} \rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} &= \Delta t^K\, \langle \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))}\, J_K\, \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \delta_{KK'} \\ &= \Delta t^K\, J_K\, w_l\, w_n\, \delta_{KK'}\, \delta_{ll'}\, \delta_{nn'}. \end{aligned} \tag{A.41}$$

  Note that (A.41) also implies compact support in time although this is not manifested by a Kronecker delta.

- Orthogonal variables:

$$\begin{aligned} \langle \boldsymbol{\phi}_n^{K:v},\, \boldsymbol{\phi}_{n'}^{K':v'} \rangle_{L^2(K)} &= \delta_{vv'}\, \langle \phi_n^K,\, \phi_{n'}^K \rangle_{L^2(K)} \\ &= J_K\, w_n\, \delta_{KK'}\, \delta_{vv'}\, \delta_{nn'}, \end{aligned} \tag{A.42}$$

$$\begin{aligned} \langle \boldsymbol{\theta}_{l:n}^{K:v},\, \boldsymbol{\theta}_{l':n'}^{K':v'} \rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} &= \langle \theta_{l:n}^K,\, \theta_{l':n'}^{K'} \rangle_{L^2(K\times(t^K, t^K+\Delta t^K))}\, \delta_{vv'} \\ &= \Delta t^K\, J_K\, w_l\, w_n\, \delta_{KK'}\, \delta_{vv'}\, \delta_{ll'}\, \delta_{nn'}. \end{aligned} \tag{A.43}$$

A.4. Derivative operators on the reference cell

In this section, we introduce elementary operators that find application within the derivation of the ADER-DG matrix operator and vector elements.

Below, let $l, l' \in \{0, 1, \ldots, N\}$, $n, n' \in \{0, 1, \ldots, (N+1)^d - 1\}$, $K, K' \in \mathcal{T}_h$ and $v, v' \in \{0, 1, \ldots, N_{\text{var}} - 1\}$.

First, let us introduce an operator $\hat{P} \in \mathbb{R}^{(N+1)\times(N+1)}$ that projects the coefficients associated with the univariate reference basis functions on the $(N+1)$ nodes of a regular partition of interval $(0, 1)$:

$$\hat{P}_{ij} = \varphi_i\left(\frac{j}{N}\right), \qquad i, j = 0, 1, \ldots, N. \tag{A.44}$$

Let us further introduce the operators

$$\left[\hat{f}, \hat{g}\right]^{\tau}_{L^2(\hat{K})} = \int_{\hat{K}} \hat{f}(\hat{\mathbf{x}}, \tau)\, \hat{g}(\hat{\mathbf{x}}, \tau)\, \mathrm{d}\hat{\mathbf{x}}, \tag{A.45}$$

$$[f, g]^{\tau}_{L^2(K)} = \int_{K} f(\mathbf{x}, t^K + \Delta t^K\, \tau)\, g(\mathbf{x}, t^K + \Delta t^K\, \tau)\, \mathrm{d}\mathbf{x}, \tag{A.46}$$

where $\tau \in \{0,\, 1\}$, $\hat{f}$ and $\hat{g}$ are square integrable on $\hat{K} \times (0,\, 1)$, and f and g are square integrable on $K \times (t^K,\, t^K + \Delta t^K)$.

For our basis polynomials, we obtain

$$\begin{aligned}
\left[\hat{\theta}_{l:n},\, \hat{\phi}_{n'}\right]^{\tau}_{L^2(\hat{K})} &= \hat{\varphi}_l(\tau)\ \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\
&= \hat{\varphi}_l(\tau)\, \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\
&= w_n\, \hat{F}^{\tau}_l\, \delta_{nn'}, && \text{(A.47)} \\
\left[\boldsymbol{\theta}^{K:v}_{l:n},\, \boldsymbol{\phi}^{K:v}_{n'}\right]^{\tau}_{L^2(K)} &= J_K\ \left[\hat{\theta}_{l:n},\, \hat{\phi}_{n'}\right]^{\tau}_{L^2(\hat{K})} \\
&= J_K\, w_n\, \hat{F}^{\tau}_l\, \delta_{nn'}, && \text{(A.48)} \\
\left[\hat{\theta}_{l:n},\, \hat{\theta}_{l':n'}\right]^{\tau}_{L^2(\hat{K})} &= \hat{\varphi}_l(\tau)\, \hat{\varphi}_{l'}(\tau)\ \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\
&= w_n\, \hat{F}^{\tau}_l\, \hat{F}^{\tau}_{l'}\, \delta_{nn'}, && \text{(A.49)} \\
\left[\boldsymbol{\theta}^{K:v}_{l:n},\, \boldsymbol{\theta}^{K:v}_{l':n'}\right]^{\tau}_{L^2(K)} &= J_K\ \left[\hat{\theta}_{l:n},\, \hat{\theta}_{l':n'}\right]^{\tau}_{L^2(\hat{K})} \\
&= J_K\, w_n\, \hat{F}^{\tau}_l\, \hat{F}^{\tau}_{l'}\, \delta_{nn'}, && \text{(A.50)}
\end{aligned}$$

where we have introduced the vectors $\hat{F}^f \in \mathbb{R}^{N+1}$, $f = 0, 1$, with

$$\hat{F}^f_i = \begin{cases} \hat{\varphi}_i(0) & f = 0, \\ \hat{\varphi}_i(1) & f = 1, \end{cases} \tag{A.51}$$

$$= \begin{cases} \hat{P}_{i0} & f = 0, \\ \hat{P}_{iN} & f = 1, \end{cases} \tag{A.52}$$

where $i \in \{0, 1, \ldots, N\}$.

We define the reference stiffness operator $\underline{\hat{\mathrm{K}}} \in \mathbb{R}^{(N+1)^2}$ according to:

$$\begin{aligned}\hat{\mathrm{K}}_{ij} &= \langle \partial_{\hat{x}}\, \hat{\varphi}_i,\, \hat{\varphi}_j \rangle_{L^2((0,1))} \\ &\overset{(A.30)}{=} w_j\, \partial_{\hat{x}}\, \hat{\varphi}_i(\hat{x}_j), \qquad i,j \in \{0,1,\ldots,N\}. \end{aligned} \tag{A.53}$$

Let us further introduce time derivative operators on the space-time reference cell and the space-time mesh cells $K \in \mathcal{T}_h$:

$$\begin{aligned}\left\langle \partial_{\hat{t}} \hat{\theta}_{l:n},\, \hat{\theta}_{l':n'} \right\rangle_{L^2(\hat{K}\times(0,1))} &= \langle \partial_{\hat{x}}\, \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))}\, \langle \hat{\phi}_n,\, \hat{\phi}_{n'} \rangle_{L^2(\hat{K})} \\ &\overset{(I)}{=} w_n\, \hat{K}_{ll'}\, \delta_{nn'}, \end{aligned} \tag{A.54}$$

$$\begin{aligned}\left\langle \partial_{\hat{t}} \boldsymbol{\theta}^{K:v}_{l:n},\, \boldsymbol{\theta}^{K:v}_{l':n'} \right\rangle_{L^2(K\times(t^K,t^K+\Delta t^K))} &= J_K \left\langle \partial_{\hat{t}} \hat{\theta}_{l:n},\, \hat{\theta}_{l':n'} \right\rangle_{L^2(\hat{K}\times(0,1))} \delta_{KK'}\, \delta_{vv'} \\ &= J_K\, w_n\, \hat{K}_{ll'}\, \delta_{nn'}, \end{aligned} \tag{A.55}$$

where we have used the operator (A.53) and the discrete orthogonality property of the basis functions (A.35) in step I.

Let $\xi \in \{1,2,\ldots,d\}$. We also introduce spatial derivative operators on the spatial and space-time reference cells:

$$\begin{aligned}\left\langle \hat{\phi}_n,\, \partial_{\hat{x}_\xi} \hat{\phi}_{n'} \right\rangle_{L^2(\hat{K})} &= \langle \hat{\phi}_n,\, \partial_{\hat{x}_\xi} \hat{\phi}_{n'} \rangle_{L^2(\hat{K})}, \\ &= \langle \hat{\varphi}_{n_\xi},\, \partial_{\hat{x}} \hat{\varphi}_{n'_\xi} \rangle_{L^2(0,1))} \prod_{\zeta=1,\zeta\neq\xi}^{d} \langle \hat{\varphi}_{n_\zeta},\, \hat{\varphi}_{n'_\zeta} \rangle_{L^2((0,1))}, \\ &= \hat{K}_{n'_\xi n_\xi} \prod_{\zeta=1,\zeta\neq\xi}^{d} w_{n_\zeta}\, \delta_{n_\zeta n'_\zeta}, \end{aligned} \tag{A.56}$$

$$\begin{aligned}\left\langle \hat{\theta}_{l:n},\, \partial_{\hat{x}_\xi} \hat{\theta}_{l':n'} \right\rangle_{L^2(\hat{K}\times(0,1))} &= \langle \hat{\varphi}_l,\, \hat{\varphi}_{l'} \rangle_{L^2((0,1))}\, \langle \hat{\phi}_n,\, \partial_{\hat{x}_\xi} \hat{\phi}_{n'} \rangle_{L^2(\hat{K})}, \\ &= w_l\, \hat{K}_{n'_\xi n_\xi}\, \delta_{ll'} \prod_{\zeta=1,\zeta\neq\xi}^{d} w_{n_\zeta}\, \delta_{n_\zeta n'_\zeta}. \end{aligned} \tag{A.57}$$

A.5. Basis representation

Assume that the vector-valued function $\mathbf{f} \in L^2(\Omega)^{N_{\mathrm{var}}}$ is sufficiently regular. Let us denote by $\mathbf{f}^K = \mathbf{f}|_K$ the restriction of f to K, and let us define

$$\mathbf{f}^K_n = \mathbf{f}^K(\mathbf{x}_n) = (f^{K;v}(\mathbf{x}_n))_{0\le v\le N_{\mathrm{var}}}, \tag{A.58}$$

where the $\mathbf{x}_n$, $n = 0, 1, \ldots, (N+1)^d - 1$, denote the Gauss-Legendre nodes within cell K. We can express $\mathbf{f}^K$ with respect to the spatial basis functions according to:

$$\mathbf{f}^K \approx \sum_{v=0}^{N_{\text{var}}} \sum_{n=0}^{(N+1)^d-1} \frac{\left\langle \mathbf{f}^K,\, \boldsymbol{\phi}_n^{K:v} \right\rangle_{L^2(K)}}{\left\langle \boldsymbol{\phi}_n^{K:v}, \boldsymbol{\phi}_n^{K:v} \right\rangle_{L^2(K)}} \boldsymbol{\phi}_n^{K:v} \overset{\text{(I)}}{=} \sum_{v=0}^{N_{\text{var}}} \sum_{n=0}^{(N+1)^d-1} \mathbf{f}_n^{K:v}\, \boldsymbol{\phi}_n^{K:v},$$

where we have used the sampling property (A.32), and the orthogonality property (A.42) of the basis polynomials in step I.

Similarly, assume that the vector-valued function $\widetilde{\mathbf{f}} \in L^2(\Omega \times (0,T))^{N_{\text{var}}}$ is sufficiently regular. Let us denote by $\widetilde{\mathbf{f}}^K = \widetilde{\mathbf{f}}|_K$ the restriction of f to K, and let us define

$$\widetilde{\mathbf{f}}_{l:n}^K = \widetilde{\mathbf{f}}^K(\mathbf{x}_n, t_l) = (\widetilde{f}^{K;v}(\mathbf{x}_n, t_l))_{0 \le v \le N_{\text{var}}}, \tag{A.59}$$

where the $(\mathbf{x}_n, t_l)^{\mathrm{T}}$, $l = 0, 1, \ldots, N$, $n = 0, 1, \ldots, (N+1)^d - 1$, denote the Gauss-Legendre nodes within the space-time cell $K \times (t^K, t^K + \Delta t^K)$.

We can express the components $\widetilde{\mathbf{f}}^K$ with respect to the space-time basis functions according to:

$$\widetilde{\mathbf{f}}^K \approx \sum_{v=0}^{N_{\text{var}}} \sum_{l=0}^{N} \sum_{n=0}^{(N+1)^d-1} \frac{\left\langle \widetilde{\mathbf{f}}^K,\, \boldsymbol{\theta}_{l:n}^{K:v} \right\rangle_{L^2(K \times (t_K, t_K + \Delta t^K))}}{\left\langle \boldsymbol{\theta}_{l:n}^{K:v}, \boldsymbol{\theta}_{l:n}^{K:v} \right\rangle_{L^2(K \times (t_K, t_K + \Delta t^K))}} \boldsymbol{\theta}_{l:n}^{K:v} \overset{\text{(I)}}{=} \sum_{v=0}^{N_{\text{var}}} \sum_{l=0}^{N} \sum_{n=0}^{(N+1)^d-1} \widetilde{\mathbf{f}}_{l:n}^{K:v}\, \boldsymbol{\theta}_{l:n}^{K:v},$$

where we have used the sampling property (A.33), and the orthogonality property (A.43) of the basis polynomials in step I.

A.6. Degrees of freedom

We express the cell-local parts of the quantities that are involved in the ADER-DG method introduced in Ch. 1 in terms of the space-time and spatial basis functions and further introduce other quantities that find application in the implementation.

Since the implementation of the scheme is usually performed via a loop over the cells $K \in \mathcal{T}_h$, we adopt a cell-based viewpoint in the following derivations. We further drop the time step index i and assign each cell a time step t^K and a time step increment Δt^K which are both updated in every iteration of an outer time update loop.

- Cell-local space-time predictor $\widetilde{\mathbf{q}}_h^K \in Q_N(K \times (t^K, t^K + \Delta t^K))^{N_{\text{var}}}$:

$$\widetilde{\mathbf{q}}_h^K = \sum_{v=0}^{N_{\text{var}}} \sum_{l=0}^{N} \sum_{n=0}^{(N+1)^d-1} \widetilde{q}_{l:n}^{K:v}\, \boldsymbol{\theta}_{l:n}^{K:v}. \tag{A.60}$$

We store the space-time predictor coefficients in the vector $\underline{\widetilde{\mathrm{q}}}^K \in \mathbb{R}^{N_{\mathrm{var}}(N+1)^{d+1}}$.

- Cell-local space-time volume flux $\widetilde{\mathbf{F}}_h^K \in Q_N(K \times (t^K, t^K + \Delta t^K))^{N_{\mathrm{var}} \times d}$:

$$\begin{aligned}\widetilde{\mathbf{F}}_h^K &= \sum_{v=0}^{N_{\mathrm{var}}} \sum_{l=0}^{N} \sum_{n=0}^{(N+1)^d-1} \widetilde{\mathbf{F}}_{l:n}^{K:v} \, \boldsymbol{\theta}_{l:n}^{K:v} \\ &= \sum_{v=0}^{N_{\mathrm{var}}} \sum_{l=0}^{N} \sum_{n=0}^{(N+1)^d-1} (\widetilde{F}_{l:n,1}^{K:v}, \ldots, \widetilde{F}_{l:n,d}^{K:v}) \, \boldsymbol{\theta}_{l:n}^{K:v}. \end{aligned} \tag{A.61}$$

- Cell-local space-time source term $\widetilde{\mathbf{s}}_h^K \in Q_N(K \times (t^K, t^K + \Delta t^K))^{N_{\mathrm{var}}}$:

$$\widetilde{\mathbf{s}}_h^K = \sum_{v=0}^{N_{\mathrm{var}}} \sum_{l=0}^{N} \sum_{n=0}^{(N+1)^d-1} \widetilde{S}_{l:n}^{K:v} \, \boldsymbol{\theta}_{l:n}^{K:v}. \tag{A.62}$$

- Cell-local predictor $\mathbf{q}_h^K \in Q_N(K)^{N_{\mathrm{var}}}$:

$$\mathbf{q}_h^K = \sum_{v=0}^{N_{\mathrm{var}}} \sum_{n=0}^{(N+1)^d-1} \widetilde{\mathbf{q}}_n^{K:v} \, \boldsymbol{\phi}_n^{K:v} \tag{A.63}$$

We store the predictor coefficients in the vector $\underline{\mathrm{q}}^K \in \mathbb{R}^{N_{\mathrm{var}}(N+1)^d}$.

- We introduce a cell-local solution $\mathbf{u}_h \in Q_N(K)^{N_{\mathrm{var}}}$:

$$\mathbf{u}_h^K = \sum_{v=0}^{N_{\mathrm{var}}} \sum_{n=0}^{(N+1)^d-1} u_n^{K:v} \, \boldsymbol{\phi}_n^{K:v} \tag{A.64}$$

We store the solution coefficients in the vector $\underline{\mathrm{u}}^K \in \mathbb{R}^{N_{\mathrm{var}}(N+1)^d}$.

- We introduce a cell-local source term $\mathbf{s}^K \in Q_N(K)^{N_{\mathrm{var}}}$:

$$\mathbf{s}_h^K = \sum_{v=0}^{N_{\mathrm{var}}} \sum_{n=0}^{(N+1)^d-1} s_n^{K:v} \, \boldsymbol{\phi}_n^{K:v}. \tag{A.65}$$

- We introduce a cell-local volume flux $\mathbf{F}_h^K \in Q_N(K)^{N_{\mathrm{var}} \times d}$:

$$\mathbf{F}_h^K = \sum_{v=0}^{N_{\mathrm{var}}} \sum_{n=0}^{(N+1)^d-1} \mathbf{F}_n^{K:v} \, \boldsymbol{\phi}_n^{K:v} \tag{A.66}$$

$$= \sum_{v=0}^{N_{\mathrm{var}}} \sum_{n=0}^{(N+1)^d-1} (F_{n,1}^{K:v}, \ldots, F_{n,d}^{K:v}) \, \boldsymbol{\phi}_n^{K:v}. \tag{A.67}$$

A.7. Space-time predictor computation

Let us insert $\widetilde{\mathbf{q}}_h^K$, $\mathbf{u}_h^K$, $\widetilde{\mathbf{F}}_h^K$, $\widetilde{\mathbf{s}}_h^K$ defined as in (A.60), (A.64), (A.61), and (A.62) into the discrete variational problem (1.21) – (1.22).

The space-time predictor computation is a completely cell-local computation due to the compact support of the spatial and space-time basis functions (cf. (A.40) and (A.41)). No data from neighbouring cells are required.

Furthermore, the computation of space-time volume fluxes and the space-time source term couples the individual variables $v = 0, 1, \ldots, N_{\text{var}} - 1$ of the space-time predictor. After the computation, the orthogonality properties (A.43) apply to the space-time volume fluxes and the space-time source term.

For every $K \in \mathcal{T}_h$, the space-time predictor computation is thus reduced to perform N_{iter} Picard iterations in the form given below to obtain the cell-local space-time predictor coefficients $\widetilde{q}_{l:n}^{K:v}$, $v = 0, 1, \ldots, N_{\text{var}} - 1$, $l = 0, 1, \ldots, N$, $n = 0, 1, \ldots, (N+1)^d - 1$, where we omit an additionally loop over each, the cells and the variables, due to the reasons given above:

$$
\begin{aligned}
&\sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \left(\left[\boldsymbol{\theta}_{l:n}^{K:v},\, \boldsymbol{\theta}_{l':n'}^{K:v} \right]_{L^2(K)}^1 - \left\langle \partial_t \boldsymbol{\theta}_{l:n}^{K:v},\, \boldsymbol{\theta}_{l':n'}^{K:v} \right\rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} \right) \widetilde{q}_{l':n'}^{K:v:(r+1)} \\
&\qquad = \sum_{n'=0}^{(N+1)^d-1} \left[\boldsymbol{\theta}_{l:n}^{K:v},\, \boldsymbol{\phi}_{n'}^{K:v} \right]_{L^2(K)}^0 u_{n'}^{K:v} \\
&\qquad - \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \widetilde{\mathbf{F}}_{l':n'}^{K:v}(\widetilde{q}_h^{K:(r)}) \left\langle \boldsymbol{\theta}_{l':n'}^{K:v},\, \nabla \boldsymbol{\theta}_{l:n}^{K:v} \right\rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} \\
&\qquad + \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \widetilde{s}_{l':n'}^{K:v}(\widetilde{q}_h^{K:(r)}) \left\langle \boldsymbol{\theta}_{l':n'}^{K:v},\, \boldsymbol{\theta}_{l:n}^{K:v} \right\rangle_{L^2(K\times(t^K, t^K+\Delta t^K))}, \qquad \text{(A.68)}
\end{aligned}
$$

where $r \in \{0, 1, \ldots, N_{\text{iter}}\}$ denotes the current iteration. In matrix notation, we obtain:

$$
\begin{aligned}
\underline{\mathrm{L}}^{KK}\, \underline{\widetilde{\mathrm{q}}}^{K:(r+1)} &= \underline{\mathrm{v}}^K(\underline{\mathrm{u}}^K) - \underline{\mathrm{w}}^K(\underline{\widetilde{\mathrm{q}}}^{K:(r)}) && \text{(A.69)} \\
\Rightarrow \underline{\widetilde{\mathrm{q}}}^{K:(r+1)} &= (\underline{\mathrm{L}}^{KK})^{-1} \left(\underline{\mathrm{v}}^K(\underline{\mathrm{u}}^K) - \underline{\mathrm{w}}^K(\underline{\widetilde{\mathrm{q}}}^{K:(r)}) \right), && \text{(A.70)}
\end{aligned}
$$

where we identify the left-hand side operator $\underline{\mathrm{L}}^{KK} \in \mathbb{R}^{N_{\text{var}}\,(N+1)^{d+1} \times N_{\text{var}}\,(N+1)^{d+1}}$, and the right-hand side vectors $\underline{\mathrm{v}}^K \in \mathbb{R}^{N_{\text{var}}\,(N+1)^{d+1}}$ depending on $\underline{\mathrm{u}}^K$, and $\underline{\mathrm{w}}^{K:(r)} \in \mathbb{R}^{N_{\text{var}}\,(N+1)^{d+1}}$ depending on $\underline{\widetilde{\mathrm{q}}}^{K:(r)}$.

A.7.1. Left-hand side operator

The matrix operator $\underline{\mathrm{L}}^{KK:vv} \in \mathbb{R}^{N_{\mathrm{var}}\,(N+1)^{d+1} \times N_{\mathrm{var}}\,(N+1)^{d+1}}$ has the elements:

$$\begin{aligned}
\mathrm{L}^{KK:vv}_{ll':nn'} &= \left[\boldsymbol{\theta}^{K:v}_{l:n},\, \boldsymbol{\theta}^{K:v}_{l':n'}\right]^1_{L^2(K)} - \left\langle \partial_t \boldsymbol{\theta}^{K:v}_{l:n},\, \boldsymbol{\theta}^{K:v}_{l':n'} \right\rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} \\
&\overset{\text{(I)}}{=} J_K\, w_n\, (\hat{F}^1_l\, \hat{F}^1_{l'} - \hat{K}_{ll'})\, \delta_{nn'} \\
&= J_K\, w_n\, \hat{L}_{ll'}\, \delta_{nn'},
\end{aligned} \tag{A.71}$$

where we have introduced the reference cell operator

$$\hat{L}_{ll'} = \hat{F}^1_l\, \hat{F}^1_{l'} - \hat{K}_{ll'}, \tag{A.72}$$

and where $v \in \{0, 1, \ldots, N_{\mathrm{var}}-1\}$ $l, l' \in \{0, 1, \ldots, N\}$, and $n, n' \in \{0, 1, \ldots, (N+1)^d-1\}$. In step I of the above derivations, we used (A.48) and (A.55).

A.7.2. Constant right-hand side term

The elements of vector $\underline{\mathrm{v}}^K \in \mathbb{R}^{N_{\mathrm{var}}(N+1)^{d+1}}$ do not change between the Picard iterations and are computed according to:

$$\begin{aligned}
\mathrm{v}^{K:v}_{l:n} &= \sum_{n'=0}^{(N+1)^d-1} \left[\boldsymbol{\theta}^{K:v}_{l:n},\, \boldsymbol{\phi}^{K:v}_{n'}\right]^0_{L^2(K)}\, u^{K:v}_{n'} \\
&\quad + \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} S^{K:v}_{l':n'}(\widetilde{q}^{K:(r)})\, \left\langle \boldsymbol{\theta}^{K:v}_{l':n'},\, \boldsymbol{\theta}^{K:v}_{l:n} \right\rangle_{L^2(K\times(t^K, t^K+\Delta t^K))} \\
&\overset{\text{(I)}}{=} \sum_{n'=0}^{(N+1)^d-1} J_K\, w_n\, \hat{F}^0_l\, \delta_{nn'}\, u^{K:v}_{n'} \\
&\quad + \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} S^{K:v}_{l':n'}\, \Delta t^K\, J_K\, w_l\, w_n\, \delta_{ll'}\, \delta_{nn'} \\
&= J_K\, w_n\, \hat{F}^0_l\, u^{K:v}_n + \Delta t^K\, J_K\, w_l\, w_n\, S^{K:v}_{l:n},
\end{aligned}$$

with $v = 0, 1, \ldots, N_{\mathrm{var}} - 1$, $l = 0, 1, \ldots, N$, and $n = 0, 1, \ldots, (N+1)^d - 1$. We have used (A.48) and (A.43) in step I of the above derivations.

A.7.3. Space-time volume flux integral

The elements of vector $\underline{\mathrm{w}}^K$ are computed according to:

$$\begin{aligned}
\mathrm{w}_{l:n}^{K:v} &= \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \widetilde{\mathbf{F}}_{l':n'}^{K:v} \left\langle \boldsymbol{\theta}_{l':n'}^{K:v},\, \nabla \boldsymbol{\theta}_{l:n}^{K:v} \right\rangle_{L^2(K\times(t^K,t^K+\Delta t^K))} \\
&\overset{\text{(I)}}{=} \Delta t^K\, J_K \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \widetilde{\mathbf{F}}_{l':n'}^{K:v}\, \mathrm{D}\boldsymbol{\mathcal{F}}_K^{-\mathrm{T}} \left\langle \hat{\theta}_{l':n'},\, \hat{\nabla}\hat{\theta}_{l:n} \right\rangle_{L^2(\hat{K}\times(0,1))} \\
&\overset{\text{(II)}}{=} J_K \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \sum_{\xi=1}^{d} \frac{\Delta t^K}{\Delta x_\xi} \left\langle \hat{\theta}_{l':n'},\, \partial_{\hat{x}_\xi}\hat{\theta}_{l:n} \right\rangle_{L^2(\hat{K}\times(0,1))} \widetilde{F}_{l':n',\xi}^{K:v} \\
&\overset{\text{(III)}}{=} J_K \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \sum_{\xi=1}^{d} \frac{\Delta t^K}{\Delta x_\xi}\, \hat{\mathrm{K}}_{n_\xi n'_\xi}\, \widetilde{F}_{l':n',\xi}^{K:v}\, w_l\, \delta_{ll'} \prod_{\zeta=1,\zeta\neq\xi}^{d} w_{n_\zeta}\, \delta_{n'_\zeta n_\zeta},
\end{aligned} \tag{A.73}$$

with $v = 0, 1, \ldots, N_{\mathrm{var}} - 1$, $l = 0, 1, \ldots, N$, and $n = 0, 1, \ldots, (N+1)^d - 1$. In step I of the above derivations, we applied a scaling argument. In step II, we expanded the d-dimensional scalar product and wrote the integral over the space-time reference cell as a discrete scalar product. In step III, we used (A.57).

A.7.4. Time averaging of space-time quantities

In the current implementation, we compute a time average of the space-time predictor and space-time volume flux values before performing a Riemann solve at the interfaces between cells.

Let us express a component of the local space-time predictor in terms of the local space-time basis

$$\widetilde{q}_h^K = \sum_{n'=0}^{N_{\mathrm{var}}} \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \widetilde{q}_{l':n'}^{K:v}\, \boldsymbol{\theta}_{l':n'}^{K:v}$$

We want to compute the predictor in $(t^K, t^K + \Delta t^K)$ as the time average of the space-time predictor over the same interval, i.e.,

$$
\begin{aligned}
q_h^K &= \sum_{n'=0}^{(N+1)^d-1} q_{n'}^{K:v}\, \boldsymbol{\phi}_{n'}^{K:v} \\
&= \frac{1}{\Delta t^K} \int_{t^K}^{t^K+\Delta t^K} \sum_{l'=0}^{N} \sum_{n'=0}^{(N+1)^d-1} \widetilde{q}_{l':n'}^{K:v}\, \boldsymbol{\theta}_{l':n'}^{K:v}\, \mathrm{d}t \\
&= \sum_{n'=0}^{(N+1)^d-1} \boldsymbol{\phi}_{n'}^{K:v}\, \frac{1}{\Delta t^K} \int_{t^K}^{t^K+\Delta t^K} \sum_{l'=0}^{N} \widetilde{q}_{l':n'}^{K:v}\, \varphi_{l'}^{K:v}\, \mathrm{d}t \\
&= \sum_{n'=0}^{(N+1)^d-1} \boldsymbol{\phi}_{n'}^{K:v}\, \frac{1}{\Delta t^K} \sum_{l'=0}^{N} \widetilde{q}_{l':n'}^{K:v}\, \langle 1,\, \varphi_{l'}^{K:v} \rangle_{L^2(t^K, t^K+\Delta t^K)} \\
&\overset{(A.30)}{=} \sum_{n'=0}^{(N+1)^d-1} \boldsymbol{\phi}_{n'}^{K:v} \sum_{l'=0}^{N} w_{l'}\, \widetilde{q}_{l':n'}^{K:v}.
\end{aligned}
$$

Thus,

$$
q_{n'}^{K:v} = \sum_{l'=0}^{N} w_{l'}\, \widetilde{q}_{l':n'}^{K:v}, \qquad n = 0, 1, \ldots, (N+1)^d - 1. \tag{A.74}
$$

We analogously define the volume flux as time average of the space-time volume flux:

$$
\mathbf{F}_{n'}^{K:v} = \sum_{l'=0}^{N} w_{l'}\, \widetilde{\mathbf{F}}_{l':n'}^{K:v}, \qquad n = 0, 1, \ldots, (N+1)^d - 1, \tag{A.75}
$$

and the source term as time average of the space-time source term:

$$
s_{n'}^{K:v} = \sum_{l'=0}^{N} w_{l'}\, \widetilde{s}_{l':n'}^{K:v}, \qquad n = 0, 1, \ldots, (N+1)^d - 1. \tag{A.76}
$$

A.8. Boundary extrapolation

In the ADER-DG method, we need to solve a Riemann problem on the interfaces between mesh cells. The Riemann solver requires from both cells involved in the Riemann solve at a interface, a extrapolation of the cell-local predictor and of the normal component of the cell-local volume flux to the interface.

Such a extrapolation is equivalent to projecting volume based degrees of freedom onto degrees of freedoms located on the interface. Note that this operation which we call *boundary extrapolation* can in some sense be interpreted as the "transposed" operation to the surface integral discussed in the Sec. A.9.3.

Let us define on the reference element $2\,d$ sets of support points

$$\{\hat{\mathbf{x}}_k\}^{\xi,f}_{k=0,1,\ldots,(N+1)^{d-1}-1} = \left\{\hat{\mathbf{x}}_n\colon\ \hat{x}_{n,\xi} = 0 \text{ if } f = 0 \text{ or } \hat{x}_{n,\xi} = 1 \text{ if } f = 1,\ n = 0, 1, \ldots, (N+1)^d - 1\right\},$$

where $\xi = 1, 2, \ldots, d$, and $f = 0, 1$. We construct the corresponding support points on the boundary of the mesh cells $K \in \mathcal{T}_h$ by applying a mapping:

$$\{\mathbf{x}_k\}^{\xi,f}_{k=0,1,\ldots,(N+1)^{d-1}-1} = \boldsymbol{\mathcal{F}}_K\{\hat{\mathbf{x}}_k\}^{\xi,f}_{k=0,1,\ldots,(N+1)^{d-1}-1}$$

The boundary extrapolation requires us to sum the contributions of the basis functions at the quadrature points on each face $e^{\xi,f}$.

Let us start with the extrapolation of the normal flux components $(\mathbf{F}_h^{K:v}\,\mathbf{n}^\xi) \in Q_N(K)^{\text{var}}$, $v = 0, 1, \ldots, N_{\text{var}} - 1$, $\xi \in \{1, 2, \ldots, d\}$, which are scalar-valued quantities. The flux tensor components $\mathbf{F}_h^{K:v}$, $v = 0, 1, \ldots, N_{\text{var}} - 1$, are here defined as in (A.75).

Using the quadrature nodes $\mathbf{x}_k^{\xi,f} \in \{\hat{\mathbf{x}}_k\}^{\xi,f}_{k=0,1,\ldots,(N+1)^{d-1}-1}$,$k = 0, 1, \ldots, (N+1)^{d-1} - 1$, located on face $e^{\xi,f}$, $\xi = 1, 2, \ldots, d$, $f = 0, 1$, we obtain for the components of the normal flux vector $\underline{\mathrm{g}}^{K:\xi,f} \in \mathbb{R}^{N_{\text{var}}\,(N+1)^{d-1}}$:

$$\begin{aligned}
\mathrm{g}_k^{K:\xi,f:v} &= \sum_{n'=0}^{(N+1)^d-1} (\mathbf{F}_h^{K:v}\,\mathbf{n}^\xi)_{n'}\,\phi_{n'}^K(\mathbf{x}_k^{\xi,f}) \\
&\overset{\text{(I)}}{=} \sum_{n'=0}^{(N+1)^d-1} \mathbf{F}_{n',\xi}^{K:v}\,\hat{\phi}_{n'}(\hat{\mathbf{x}}_k^{\xi,f}) \\
&\overset{\text{(II)}}{=} \sum_{n'=0}^{(N+1)^d-1} \mathbf{F}_{n',\xi}^{K:v}\,\hat{\varphi}_{n'_\zeta}(\hat{x}_{k,\xi}^{\xi,f}) \prod_{\zeta=1,\zeta\neq\xi}^{d} \hat{\varphi}_{n'_\zeta}(\hat{x}_{k,\zeta}^{\xi,f}) \\
&\overset{\text{(III)}}{=} \sum_{n'=0}^{(N+1)^d-1} \mathbf{F}_{n',\xi}^{K:v}\,\hat{\varphi}_{n'_\xi}(\hat{x}_{k,\xi}^{\xi,f}) \prod_{\zeta=1,\zeta\neq\xi}^{d} \delta_{n'_\zeta k_\zeta} \\
&= \sum_{n'=0}^{(N+1)^d-1} \mathbf{F}_{n',\xi}^{K:v}\,\hat{F}_{n'_\xi}^f \prod_{\zeta=1,\zeta\neq\xi}^{d} \delta_{n'_\zeta k_\zeta},
\end{aligned} \tag{A.77}$$

where $k = 0, 1, \ldots, (N+1)^{d-1} - 1$, and where the vectors $\hat{F}^f \in \mathbb{R}^{N+1}$, $f = 0, 1$, are defined by (A.51). In step I of the above calculations, we have made use of the definition of the multivariate basis functions: cf. (A.22). In step II, we have split the multivariate reference basis functions into the univariate ones: cf. (A.21). In step III, we have used the Lagrange basis property (A.24).

We obtain for the elements of the extrapolated predictor vectors $\underline{\mathrm{e}}^{K:\xi f} \in \mathbb{R}^{N_{\text{var}}\,(N+1)^{d-1}}$

$\xi = 1, 2, \ldots, d,\ f = 0, 1$:

$$\begin{aligned} \mathrm{e}_k^{K:\xi,f:v} &= \sum_{n'=0}^{(N+1)^d-1} q_{n'}^{K:v}\, \boldsymbol{\phi}_{n'}^{K:v}(\mathbf{x}_k^{\xi,f}) \\ &= \sum_{n'=0}^{(N+1)^d-1} q_{n'}^{K:v}\, \hat{F}_{n'_\xi}^{f} \prod_{\zeta=1,\zeta\neq\xi}^{d} \delta_{n'_\zeta k_\zeta}, \end{aligned} \tag{A.78}$$

where $k = 0, 1, \ldots, (N+1)^{d-1} - 1$.

The arrays $\underline{\mathrm{g}}^{K:\xi,f} \in \mathbb{R}^{N_{\mathrm{var}}\,(N+1)^{d-1}}$ and $\underline{\mathrm{e}}^{K:\xi f} \in \mathbb{R}^{N_{\mathrm{var}}\,(N+1)^{d-1}}$ $\xi = 1, 2, \ldots, d,\ f = 0, 1$, then serve as input of the Riemann solver.

A.9. Correction

Let us insert $\widetilde{\mathbf{q}}_h^K$, $\widetilde{\mathbf{F}}_h^K$, $\widetilde{\mathbf{s}}_h^K$ defined as in (A.60), (A.60), and (A.62) into the discrete variational problem (1.18). In the current implementation, we then immediately integrate $\widetilde{\mathbf{q}}_h^K$, $\widetilde{\mathbf{F}}_h^K$, $\widetilde{\mathbf{s}}_h^K$ in time and introduce the time averaged space-time quantities defined in A.7.4, as well as the time independent numerical flux $\mathbf{G}_h(\mathbf{q}_h^-, \mathbf{q}_h^+)\mathbf{n}^K$.

For every $K \in \mathcal{T}_h$, we then solve a system of equations of the following form for the coefficients $u_n^{K:v}$, $n = 0, 1, \ldots, (N+1)^d - 1$, of the cell-local solution:

$$\begin{aligned} &\sum_{n'=0}^{(N+1)^d-1} \left\langle \boldsymbol{\phi}_n^{K:v},\, \boldsymbol{\phi}_{n'}^{K:v} \right\rangle_{L^2(K)} (u_n^{K:v} - u_n^{K:v(\mathrm{old})}) \\ &\qquad + \Delta t^K \int_{\partial K} \mathbf{G}_h(\mathbf{q}_h^-, \mathbf{q}_h^+) : (\boldsymbol{\phi}_n^{K:v} \otimes \mathbf{n}^K)\, \mathrm{d}\mathbf{x} \\ &\qquad - \Delta t^K \int_K \mathbf{F}_h^K\, \nabla \boldsymbol{\phi}_n^{K:v}\, \mathrm{d}\mathbf{x} \\ &\qquad = \Delta t^K \int_K \mathbf{s}_h^K\, \nabla \boldsymbol{\phi}_n^{K:v}\, \mathrm{d}\mathbf{x} \end{aligned} \tag{A.79}$$

In matrix notation, this is equivalent to solving the following system for $\underline{\mathrm{u}}^K$:

$$\underline{\mathrm{M}}^{KK}\, (\underline{\mathrm{u}}^K - \underline{\mathrm{u}}^{K:(\mathrm{old})}) = \Delta t^K\, \left(\underline{\mathrm{a}}^K - \underline{\mathrm{b}}^K + \underline{\mathrm{s}}^K\right)$$

All the operators and vectors introduced above will be discussed in detail in the next sections. Lastly, note that after solving the Riemann problem at the cell interfaces, the correction operation is reduced to a completely cell-local computation.

A.9.1. Mass operator

The cell-local mass operator $\underline{\mathrm{M}}^{KK:vv} \in \mathbb{R}^{(N+1)^d \times (N+1)^d}$ has the elements:

$$\begin{aligned} \mathrm{M}_{nn'}^{KK:vv} &= \langle \phi_{n'}^{K:v}, \phi_n^{K:v} \rangle_{L^2(K)} \\ &\overset{(A.42)}{=} J_K\, w_n\, \delta_{nn'}, \end{aligned} \tag{A.80}$$

where $n, n' = 0, 1, \ldots, (N+1)^d - 1$. The cell-local mass operator is thus diagonal.

A.9.2. Volume integral

The evaluation of the volume integral is very similar to the evaluation of the space-time volume flux integral (Sec. A.7.3).

The elements of the cell-local volume flux vector $\underline{\mathrm{a}}^K \in \mathbb{R}^{N_{\mathrm{var}}\,(N+1)^d}$ are computed according to:

$$\begin{aligned}
\mathrm{a}_n^{K:v} &= \sum_{n'=0}^{(N+1)^d-1} \mathbf{F}_{n'}^{K:v} \left\langle \boldsymbol{\phi}_{n'}^{K:v},\, \nabla \boldsymbol{\phi}_n^{K:v} \right\rangle_{L^2(K)} \\
&\overset{\text{(I)}}{=} J_K \sum_{n'=0}^{(N+1)^d-1} \mathbf{F}_{n'}^{K:v}\, \mathrm{D}\boldsymbol{\mathcal{F}}_K^{-\mathrm{T}} \left\langle \hat{\phi}_{n'},\, \hat{\nabla} \hat{\phi}_n \right\rangle_{L^2(\hat{K})} \\
&\overset{\text{(II)}}{=} J_K \sum_{\xi=1}^{d} \frac{1}{\Delta x_\xi} \sum_{n'=0}^{(N+1)^d-1} \langle \hat{\phi}_{n'},\, \partial_{\hat{x}_\xi} \hat{\phi}_n \rangle_{L^2(\hat{K})}\, F_{n',\xi}^{K:v} \\
&\overset{\text{(III)}}{=} J_K \sum_{\xi=1}^{d} \frac{1}{\Delta x_\xi} \sum_{n'=0}^{(N+1)^d-1} \hat{\mathrm{K}}_{n_\xi n'_\xi}\, F_{n',\xi}^{K:v} \prod_{\zeta=1,\zeta\neq\xi}^{d} w_{n'_\zeta}\, \delta_{n'_\zeta\, n_\zeta},
\end{aligned} \tag{A.81}$$

where $n = 0, 1, \ldots, (N+1)^d - 1$. In step I of the above derivations, we used a scaling argument. In step II, we expanded the d-dimensional scalar product and wrote the integral over the reference element as a discrete scalar product. In step III, we used (A.56).

A.9.3. Surface integral

Let us introduce positively signed normal vectors $\mathbf{n}^\xi \in \mathbb{R}^d$ that have their only non-zero at position ξ. Let us further introduce outward directed normal vectors $\mathbf{n}^{\xi f} \in \mathbb{R}^d$ that also have their only non-zero at position ξ. Their sign is however negative if $f = 0$ and positive if $f = 1$.

In our computations, the numerical flux at face (ξ, f) is defined with respect to the direction $\mathbf{n}^\xi \in \mathbb{R}^d$. From this definition of the numerical flux, it follows that the numerical flux with respect to the outward directed normal vectors is computed according to:

$$\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\, \mathbf{n}^{\xi,f} = \begin{cases} -(\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\, \mathbf{n}^\xi) & f = 0, \\ +(\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\, \mathbf{n}^\xi) & f = 1. \end{cases}, \tag{A.82}$$

where $\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\, \mathbf{n}^\xi$ denotes the numerical flux as computed by the Riemann solver at face (ξ, f), and $\mathbf{n}^{\xi,f}$ denotes the outward directed normal vector at face (ξ, f), $\xi \in \{1, 2, \ldots, d\}$, $f \in \{0, 1\}$.

Let us express the outward directed numerical flux $\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\,\mathbf{n}^{\xi,f}$ in terms of scalar-valued univariate basis functions located on the face (ξ, f):

$$\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\,\mathbf{n}^{\xi,f} = \sum_{k'=0}^{(N+1)^{d-1}-1} \mathbf{G}_{k'}^{K:v:\xi,f} \prod_{\zeta=1,\zeta\neq\xi}^{d} \varphi_{k'_\zeta}^K, \tag{A.83}$$

where $k \in \{0, 1, \ldots, (N+1)^{d-1} - 1\}$ denotes the linearised index.

The cell-local vector $\underline{\mathrm{b}}^K \in \mathbb{R}^{N_{\mathrm{var}}\,(N+1)^d}$ has the elements:

$$\begin{aligned}
\mathrm{b}_n^{K:v} &= \sum_{\xi=1}^{d}\sum_{f=0}^{1} \int_{e^{\xi,f}} (\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)) : (\boldsymbol{\phi}_n^{K:v} \otimes \mathbf{n}^{\xi,f})\,\mathrm{d}s \\
&= \sum_{\xi=1}^{d}\sum_{f=0}^{1} \int_{e^{\xi,f}} \boldsymbol{\phi}_n^{K:v}\,\mathbf{G}_h^{K:v:\xi,f}(\mathbf{q}_h^-, \mathbf{q}_h^+)\,\mathbf{n}^{\xi,f}\,\mathrm{d}s \\
&\overset{\text{(I)}}{=} \sum_{\xi=1}^{d}\sum_{f=0}^{1} \hat{F}^f_{n_\xi} \sum_{k'=0}^{(N+1)^{d-1}-1} \mathbf{G}_{k'}^{K:v:\xi,f} \prod_{\zeta=1,\zeta\neq\xi}^{d} \Delta x_\zeta\, \langle \hat{\varphi}_{n_\zeta},\ \hat{\varphi}_{k'_\zeta} \rangle_{L^2((0,1))} \\
&\overset{\text{(II)}}{=} \sum_{\xi=1}^{d}\sum_{f=0}^{1} \hat{F}^f_{n_\xi} \sum_{k'=0}^{(N+1)^{d-1}-1} \mathbf{G}_{k'}^{K:v:\xi,f} \prod_{\zeta=1,\zeta\neq\xi}^{d} \Delta x_\zeta\, w_{n_\zeta}\, \delta_{n_\zeta k'_\zeta},
\end{aligned}$$

where $n = 0, 1, \ldots, (N+1)^d - 1$. In step I, we have introduced the vectors $\hat{F}^f \in \mathbb{R}^{N+1}$, $f = 0, 1$, defined as in (A.51). We further have used the relation (A.37) in step I and II.

A.9.4. Source terms

We obtain for the elements of the cell-local source term vector $\underline{\mathrm{s}}^K \in \mathbb{R}^{N_{\mathrm{var}}\,(N+1)^d}$:

$$\begin{aligned}
\mathrm{s}_n^{K:v} = \int_K \mathbf{s}^{K:v}\,\boldsymbol{\phi}_n^{K:v}\,\mathrm{d}\mathbf{x} &= \langle\, \mathbf{s}^{K:v},\, \boldsymbol{\phi}_n^{K:v} \,\rangle_{L^2(K)} \\
&= \sum_{n'=1}^{(N+1)^d-1} s_{n'}^{K:v}\, \langle \boldsymbol{\phi}_{n'}^{K:v},\, \boldsymbol{\phi}_n^{K:v} \rangle_{L^2(K)} \\
&\overset{\text{(A.38)}}{=} \sum_{n'=1}^{(N+1)^d-1} \delta_{nn'}\, J_K\, w_{n'}\, S_{n'}^{K:v} \\
&= w_n\, J_K\, S_n^{K:v},
\end{aligned}$$

where $v = 0, 1 \ldots, N_{\mathrm{var}} - 1$, and $n = 0, 1, \ldots, (N+1)^d - 1$.

A.9.5. Solution update

The correction step requires us to solve the following equation for $\underline{\mathrm{u}}^K$

$$\begin{aligned}
\underline{\mathrm{M}}^{KK}\,(\underline{\mathrm{u}}^K - \underline{\mathrm{u}}^{K:(\mathrm{old})}) &= \Delta t^K\, \left(\underline{\mathrm{a}}^K - \underline{\mathrm{b}}^K + \underline{\mathrm{s}}^K\right) \\
\Rightarrow J_K\, w_n\, (\mathrm{u}_n^{K:v} - \mathrm{u}_n^{K:v:(\mathrm{old})}) &= \Delta t^K\, \left(\underline{\mathrm{a}}^K - \underline{\mathrm{b}}^K + \underline{\mathrm{s}}^K\right),
\end{aligned}$$

for $n = 0, 1, \ldots, (N+1)^d - 1$. From rearranging the equation, we finally obtain

$$\mathrm{u}_n^{K:v} = \mathrm{u}_n^{K:v:(\mathrm{old})} + \frac{\Delta t^K}{w_n \, J_K} \left(\mathrm{a}_n^{K:v} - \mathrm{b}_n^{K:v} + \mathrm{s}_n^{K:v} \right),$$

where $v = 0, 1, \ldots, N_{\mathrm{var}} - 1$ and $n = 0, 1, \ldots, (N+1)^d - 1$. Notice the division by J_K on the right-hand side.

A.10. Projecting the ADER-DG degrees of freedom on a uniform subgrid

The DG degrees of freedom can be projected on a uniform partition of a cell $K \in \mathcal{T}_h$ using a d-dimensional tensor product of the 1-d uniform subgrid projector defined by (A.44). A projection of the nodal degrees of freedom on a uniform partition is interesting for visualisation purposes since the Gauss-Legendre quadrature nodes tend to accumulate near the boundary of the domain of integration for an increasing number of quadrature nodes, i.e., for higher order approximations.

Let us denote by $\mathbf{u}_h^{K:(\mathrm{reg})} \in Q_N(K)^d$ the projection of the cell-solution $\mathbf{u}_h^K$ on a uniform subgrid on K. The basis coefficients of $\mathbf{u}_h^{(\mathrm{reg})}$ are computed according to:

$$u_{h:n}^{(\mathrm{reg})} \overset{(\mathrm{A.22})}{=} \sum_{n'=0}^{(N+1)^d-1} u_{h:n'} \prod_{\xi=1}^{d} \hat{P}_{n'_\xi, n_\xi}, \qquad n = 0, 1, \ldots, (N+1)^d - 1, \tag{A.84}$$

where $\hat{P}$ denotes the uniform subgrid projector defined by (A.44).

B. Prolongation and restriction operators for matrix free, quadrature free discontinuous Galerkin methods on adaptive Cartesian meshes

B.1. Introduction

This chapter is outlined as follows: In the first section, we introduce subinterval projectors for adaptive meshes in one space dimension. We require that the subintervals are constructed by recursive k-partitioning of the original root interval. For the sake of simplicity, we assume set $k = 3$ here and below. The introduced projectors are used to realise the prolongation of degrees of freedom specified on larger intervals down to the finer intervals in these adaptive meshes. Throughout this chapter, we consider matrix free, quadrature free DG methods that utilise nodal (Lagrange) basis functions that use the nodes of a Gauss-Legendre quadrature as support points.

In the sections B.3 and B.4, we then introduce prolongation and restriction operators that are utilised to represent degrees of freedom which are located within the volume or on the boundary of a mesh cell on finer and coarser levels of an adaptive mesh.

B.1.1. Contributions

We introduce a recursive way to create multilevel prolongation and restriction operators based on $k = 3$ single level operators.

We propose to encode the relative position of subintervals with respect to a tertiary basis. Decoding theses numbers then enables us to determine in which way we have to apply the single level operators in order to obtain the same action as from applying the respective multilevel operator.

B.2. Prolongation and restriction operators in one dimension

Let $\mathcal{T}_h$ denote an uniform mesh on the unit interval $\hat{C} = (0,1)$ that was constructed by recursively tripartitioning the original interval l times. We can describe any of the 3^l l^{th}-level subintervals as a shifted and scaled version of the original interval $\hat{C} = (0,1)$.

Let us assign the refinement level $l = 0$ to the unit interval and further introduce an index $i^{0,l} = 0, 1, \ldots, 3^l - 1$, $l \geq 0$. The open subintervals are constructed according to:

$$C^{0,\hat{l}:i^{0:l}} = \frac{1}{3^l}(i^{0,l}, i^{0,l} + 1). \tag{B.1}$$

Let us further define $N+1$ quadrature nodes on each subinterval by projecting quadrature nodes located on the unit interval onto the subinterval using the same shifting and scaling:

$$\hat{x}_k^{0,l:i^{0:l}} = \frac{1}{3^l}(\hat{x}_k + i^{0,l}), \tag{B.2}$$

where the points $\hat{x}_k \in (0,1)$, $k = 0, 1 \ldots, N$, denote the Gauss-Legendre nodes on the unit interval.

We further define $N+1$ Lagrange polynomials $\hat{\varphi}_k^{0,l;i^{0,l}}$ on each subinterval as well as $N+1$ Lagrange polynomials $\hat{\varphi}_k$ on the original interval, $k = 0, 1, \ldots, N$ that use the Gauss-Legendre nodes as support points. Associating nodal values with the Lagrange polynomials, we can construct any polynomial with order N or less on the subintervals and the original unit interval. The Lagrange polynomials thus form a basis of the polynomial spaces $Q_k(C^{0,\hat{l}:i^{0:l}})$ on the subintervals, and $Q_k(\hat{C})$ on the unit interval, respectively.

Let us denote the 0^{th}-level nodal values by g_k, and the nodal values associated with subinterval $C^{0,l:i^{0:l}}$ by $g_k^{l:i^{0:l}}$, where $k = 0, 1, \ldots, N$. We can project l^{th}-level nodal values associated with the original interval onto the nodal values associated with subinterval $C^{0,l;i^{0,l}}$ by means of the operator $\hat{\Pi}^{0,l:i^{0,l}} \in \mathbb{R}^{(N+1)\times(N+1)}$, where

$$\hat{\Pi}_{mn}^{0,l:i^{0,l}} = \hat{\varphi}_m\left(\frac{1}{3^l}(\hat{x}_n + i^{0,l})\right), \quad m, n = 0, 1, \ldots, N, \tag{B.3}$$

The nodal values in the subintervals are thus obtained by, first, reconstructing the 0-th polynomial using the 0-th basis functions and nodal values and, second, (over-)sampling the 0-th level polynomial (see Fig. B.1). At the same time, $\hat{\Pi}^{0,l:i^{0,l}}$ is the L^2 projector that projects the 0^{th}-level nodal values on the nodal values associated with subinterval

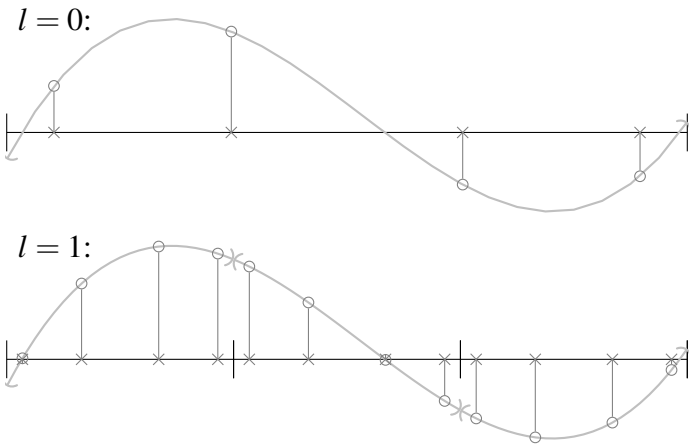

Figure B.1.: The nodal values associated with the 1st-level subintervals are obtained by oversampling the 0th-level polynomial.

$C^{0,l;i^{0,l}}$. Indeed,

$$\begin{aligned} g_k^{l:i^{0:l}} &= \sum_{k'=0}^{N} g_{k'} \frac{\left\langle \hat{\varphi}_{k'},\, \hat{\varphi}_k^{l:i^{0:l}} \right\rangle_{L^2(\hat{C}^{0,l:i^{0:l}})}}{\left\langle \hat{\varphi}_k^{l:i^{0:l}},\, \hat{\varphi}_k^{l:i^{0:l}} \right\rangle_{L^2(\hat{C}^{0,l:i^{0:l}})}} \\ &\overset{\text{(I)}}{=} \sum_{k'=0}^{N} g_{k'} \frac{\frac{1}{3^l}\, w_k\, \hat{\varphi}_{k'}\left(\frac{1}{3^l}(\hat{x}_k + i^{0,l})\right)}{\frac{1}{3^l}\, w_k} \\ &= \sum_{k'=0}^{N} g_{k'}\, \hat{\varphi}_{k'}\left(\frac{1}{3^l}(\hat{x}_k + i^{0,l})\right) \\ &= \sum_{k'=0}^{N} g_{k'}\, \hat{\Pi}_{k'k}^{0,l;i^{0,l}}, \end{aligned} \tag{B.4}$$

where $k = 0, 1, \ldots, N$, and where we have used the sampling property (A.30) and the discrete orthogonality (A.34) of the basis functions in step I.

In this work, we refer to the operation of projecting the 0th-level nodal values (or degrees of freedom) onto to the basis functions associated with the l^{th}-level subintervals as *prolongation*.

We further want to emphasise that interpolation and L^2 projection coincide for the chosen Lagrange basis functions.

B.2.1. Restriction

The *restriction* operation reconstructs a polynomial on the unit interval according to the polynomials on the subintervals. The corresponding projector needs to gather information from all subintervals.

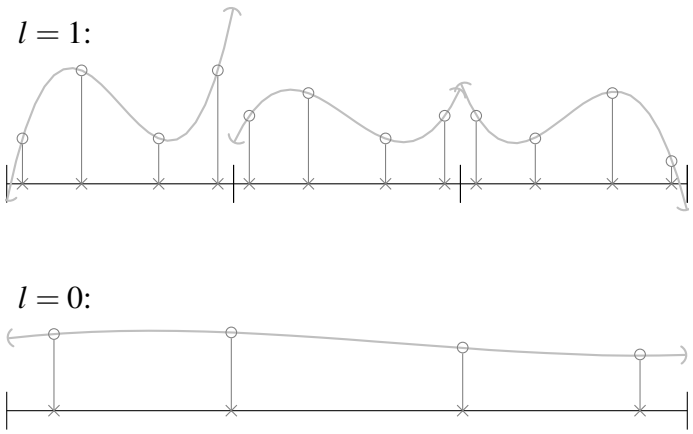

Figure B.2.: The nodal values associated with the basis functions in the 0^{th}-level interval are obtained using a L^2 projection of the 1^{st}-level polynomials.

Here, we define the contribution of each subinterval to the restriction as the L^2 projection of the subinterval coefficients on the unit interval coefficients:

$$g_k = \sum_{i^{0,l}=0}^{3^l-1} \sum_{k'=0}^{N} \frac{\left\langle \hat{\varphi}_k, \hat{\varphi}_{k'}^{l:i^{0:l}} \right\rangle_{L^2(\hat{C})}}{\langle \hat{\varphi}_k, \hat{\varphi}_k \rangle_{L^2(\hat{C})}} \, g_{k'}^{l:i^{0,l}} \tag{B.5}$$

$$\overset{\text{(I)}}{=} \sum_{i^{0,l}=0}^{3^l-1} \sum_{k'=0}^{N} \frac{\left\langle \hat{\varphi}_k, \hat{\varphi}_{k'}^{l:i^{0:l}} \right\rangle_{L^2(\hat{C}^{0,l;i^{0:l}})}}{\langle \hat{\varphi}_k, \hat{\varphi}_k \rangle_{L^2(\hat{C})}} \, g_{k'}^{l:i^{0,l}} \tag{B.6}$$

$$\overset{\text{(II)}}{=} \sum_{i^{0,l}=0}^{3^l-1} \sum_{k'=0}^{N} \frac{\frac{1}{3^l} \, w_{k'} \, \hat{\varphi}_k \left(\frac{1}{3^l} (\hat{x}_{k'} + i^{0,l}) \right)}{w_k} \, g_{k'}^{l:i^{0,l}} \tag{B.7}$$

$$= \sum_{i^{0,l}=0}^{3^l-1} \sum_{k'=0}^{N} \frac{1}{3^l} \left(\frac{w_{k'}}{w_k} \, \hat{\Pi}_{kk'}^{0,l:i^{0,l}} \right) \, g_{k'}^{l:i^{0,l}} \tag{B.8}$$

where we have used the compact support of the basis functions in step I and the sampling property of the basis functions (A.30) in step II.

The restriction leads to a damping of higher frequencies (Fig. B.2)

B.2.2. Recursive construction of the subinterval projectors

The projectors $\hat{\Pi}^{0,l:i_l} \in \mathbb{R}^{(N+1)\times(N+1)}$, $l > 0$, can be constructed using a cascade of the single-level projectors $\hat{\Pi}^{0,1:0}$, $\hat{\Pi}^{0,1:1}$, and $\hat{\Pi}^{0,1:2}$ (Figure B.3). This follows directly from the fact that the Lagrange interpolation utilising polynomials of order N is exact for polynomials of the same order. The original polynomial is thus exactly represented by the subinterval polynomials in the corresponding subinterval.

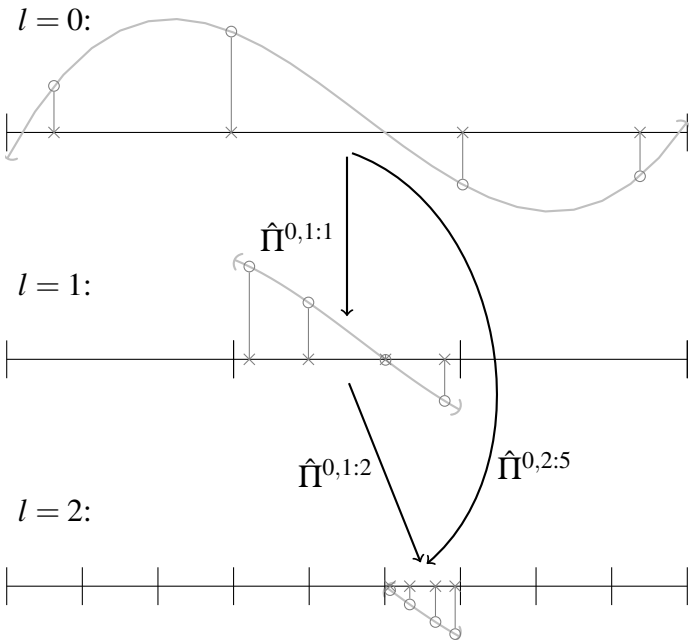

Figure B.3.: Prolongation of 0^{th}-level nodal values down to level 2. The operation can be performed by applying two single level projectors ($\hat{\Pi}^{0,1:1}$ and $\hat{\Pi}^{0,1:2}$ in the example) or by applying a two level projector ($\hat{\Pi}^{0,2:5}$ in the example).

To this end, let us express the indices $i^{0,l} = 0, 1, \ldots, 3^l - 1$ in terms of a tertiary basis:

$$i^{0,l} = \sum_{\beta=1}^{l} 3^{l-\beta}\, j_\beta, \qquad j_\beta \in \{0, 1, 2\}. \tag{B.9}$$

The support points and the corresponding Lagrange basis functions on the sub intervals can now be constructed in a recursive fashion by downscaling and shifting the parent intervals:

$$\hat{x}_k^{l:i^{0,l}} = \frac{1}{3^l}(\hat{x}_k + i^{0,l}) = \frac{1}{3}\,(\hat{x}_k^{(l-1):i^{0,l-1}} + j_l), \tag{B.10}$$

$$\hat{\varphi}_k^{l:i^{0,l}}(\hat{x}) = \left(\prod_{\substack{0 \le n \le N \\ n \neq k}} \frac{\hat{x} - \hat{x}_n^{l:i^{0,l}}}{x_k^{l:i^{0,l}} - \hat{x}_n^{l:i^{0,l}}} \right), \qquad k = 0, \ldots, N, \tag{B.11}$$

where $j_{l-1} \in \{0, 1, 2\}$ denotes the sub interval index with respect to the parent interval.

The 0^{th} nodal values are then projected onto the $i^{0,l}$-th subinterval in a recursive way

similar to the way the subintervals are constructed, i.e.:

$$\hat{\Pi}^{0,l:i^{0,l}}_{m_0\,m_l} = \sum_{m_1=0,\,m_2=0,\,\ldots,m_{l-1}=0}^{N} \hat{\Pi}^{0,1:j_1}_{m_0\,m_1} \,\ldots\, \hat{\Pi}^{(l-2),(l-1):j_{l-1}}_{m_{l-2}\,m_{l-1}}\, \hat{\Pi}^{(l-1),l:j_l}_{m_{l-1}\,m_l} \tag{B.12}$$

$$= \sum_{m_1=0,\,m_2=0,\,\ldots,m_{l-1}=0}^{N} \hat{\Pi}^{0,1:j_1}_{m_0\,m_1}\, \hat{\Pi}^{0,1:j_2}_{m_1\,m_2} \,\ldots\, \hat{\Pi}^{0,1:j_{l-1}}_{m_{l-2}\,m_{l-1}}\, \hat{\Pi}^{0,1:j_l}_{m_{l-1}\,m_l}, \tag{B.13}$$

where $m_0, m_l = 0, 1, \ldots, N$.

B.3. Subcell prolongation and restriction operators

Subcell prolongation and restriction operators are important for the prolongation and restriction of volume data in the context of dynamic adaptive mesh refinement.

These operators are applied if a mesh cell is refined and prolongates its solution degrees of freedom down to the newly created children, or if child cells are erased and need to restrict their solution degrees of freedom up to the newly introduced parent cell beforehand.

Since we only consider affine mesh cells here that are refined according to tripartitioning, the actual dimensions of the cells are not important in these projections but only the relative size of the fine level mesh cell with respect to the coarse level mesh cell.

The prolongation operator can thus be simply constructed as tensor product of the subinterval prolongation operators introduced for unit interval in the previous section.

Let us split the reference cell $\hat{K} = (0,1)^d$ into $3^{l\,d}$, $l > 0$, subcells

$$\hat{K}^{l:i_l} = \frac{1}{3^l}[i_{l,1},\, i_{l,1}+1] \times \frac{1}{3^l}[i_{l,2}, i_{l,2}+1] \times \ldots \times \frac{1}{3^l}[i_{l,d},\, i_{l,d}+1], \tag{B.14}$$

and let us further introduce a linear index

$$i^{0,l} = \sum_{\xi=1}^{d} I_{l,\xi}\, i_{l,\xi}, \tag{B.15}$$

where the $I_{l,\xi} \in \{1, 3^l\}$, $\xi \in \{1, 2, \ldots, d\}$ denote strides that define a unique, continuous index.

We can project coefficients $u_{n'}$, $n' = 0, \ldots, (N+1)^d$, associated with basis polynomials that use Gauss-Legendre quadrature nodes within the volume of $\hat{K}$ as interpolation points onto basis polynomials associated with quadrature nodes within the volume of the subcells $\hat{K}^{l:i^{0,l}}$ by using a tensor product of the subinterval projectors $\hat{\Pi}^{0,l:i^{0:l}_\xi}$, $\xi \in \{0, 1, \ldots, d\}$ defined in the previous section. Indeed, this follows directly from a L^2

projection:

$$
\begin{aligned}
u_n^{l:i^{0,l}} &= \sum_{k'=0}^{(N+1)^d-1} u_{n'} \frac{\left\langle \hat{\phi}_{n'}, \hat{\phi}_n^{l:i^{0,l}} \right\rangle_{L^2(\hat{K}^{l:i^{0,l}})}}{\left\langle \hat{\phi}_n^{l:i^{0,l}} \hat{\phi}_n^{l:i^{0,l}} \right\rangle_{L^2(\hat{K}^{l:i^{0,l}})}} \\
&= \sum_{k'=0}^{(N+1)^d-1} u_{n'} \prod_{\xi=1}^{d} \frac{\left\langle \hat{\varphi}_{n'_\xi}, \hat{\varphi}_{n_\xi}^{l:i^{0,l}} \right\rangle_{L^2(\hat{K}^{l:i^{0,l}})}}{\left\langle \hat{\varphi}_{n_\xi}^{l:i^{0,l}} \hat{\varphi}_{n_\xi}^{l:i^{0,l}} \right\rangle_{L^2(\hat{K}^{l:i^{0,l}})}} \\
&= \sum_{k'=0}^{(N+1)^d-1} u_{n'} \prod_{\xi=1}^{d} \hat{\Pi}^{0,l:i_\xi^{0,l}}_{n'_\xi,n_\xi}, \qquad n = 0, 1, \ldots, (N+1)^d - 1.
\end{aligned} \tag{B.16}
$$

B.3.1. Restriction

We define the restriction operation here as L^2 projection of the fine level coefficients onto the coarse level coefficients:

$$
\begin{aligned}
u_n &= \sum_{i^{0,l}=0}^{3^{l\,d}-1} \sum_{n'=0}^{(N+1)^d-1} \prod_{\xi=1}^{d} \frac{\left\langle \hat{\varphi}_{n_\xi}, \hat{\varphi}_{n'_\xi}^{l:i_\xi^{0:l}} \right\rangle_{L^2(\hat{K}^{0,l:i^{0:l}})}}{\left\langle \hat{\varphi}_{n_\xi}, \hat{\varphi}_{n_\xi} \right\rangle_{L^2(\hat{K})}} u_{n'}^{l:i^{0,l}} \\
&= \sum_{i^{0,l}=0}^{3^{l\,d}-1} \sum_{n'=0}^{(N+1)^d-1} \prod_{\xi=1}^{d} \frac{\left\langle \hat{\varphi}_{n_\xi}, \hat{\varphi}_{n'_\xi}^{l:i_\xi^{0:l}} \right\rangle_{L^2(\hat{K}^{0,l:i^{0:l}})}}{\left\langle \hat{\varphi}_{n_\xi}, \hat{\varphi}_{n_\xi} \right\rangle_{L^2(\hat{K})}} u_{n'}^{l:i^{0,l}} \\
&= \sum_{i^{0,l}=0}^{3^{l\,d}-1} \sum_{n'=0}^{(N+1)^d-1} \prod_{\xi=1}^{d} \frac{\frac{1}{3^l} w_{n'_\xi} \hat{\varphi}_{n_\xi}\left(\frac{1}{3^l}(\hat{x}_{n',\xi} + i_\xi^{0,l})\right)}{w_{n_\xi}} u_{n'}^{l:i^{0,l}} \\
&= \sum_{i^{0,l}=0}^{3^{l\,d}-1} \sum_{n'=0}^{(N+1)^d-1} \frac{1}{3^{l\,d}} \left(\prod_{\xi=1}^{d} \frac{w_{n'_\xi}}{w_{n_\xi}} \hat{\Pi}^{0,l:i_\xi^{0,l}}_{n_\xi n'_\xi} \right) u_{n'}^{l:i^{0,l}},
\end{aligned} \tag{B.17}
$$

for $n = 0, \ldots, (N+1)^d$.

B.4. Subface prolongation and restriction operators

The numerical flux computation in the DG method requires us to evaluate a surface integral of the following form over every face $e \subset \partial K$ of a mesh cell K:

$$
\int_e \mathbf{G}_h(\mathbf{u}_h^-, \mathbf{u}_h^+) : (\boldsymbol{\phi}_n^{K:v} \otimes \mathbf{n}^K)\, \mathrm{d}\mathbf{x}, \tag{B.18}
$$

where $\mathbf{G}_h(\mathbf{u}_h^-, \mathbf{u}_h^+)\mathbf{n}$ denotes the so-called numerical flux.

Subface projectors are important for performing the numerical flux computation on the interface between two mesh cells that differ in their refinement level. In particular, they are used to represent coarse level degrees of freedom on finer levels, as well as to represent fine level degrees of freedom on coarser levels of the mesh. The first operation can be again classified as prolongation operation, and the second as restriction operation.

Since we only consider affine mesh cells here that are refined according to tripartitioning, the actual dimensions of the cells are not important in these projections but only the relative size of the fine level mesh cell with respect to the coarse level mesh cell. The above integral can thus be reformulated as integral over the reference face $\hat{C}$.

Let us split the reference face $\hat{C} = (0,1)^{d-1}$ into $3^{l\,(d-1)}$ sub intervals,

$$\hat{C}^{l:i_l} = \frac{1}{3^l}[i_{l,1},\, i_{l,1}+1] \times \frac{1}{3^l}[i_{l,2}, i_{l,2}+1] \times \ldots \times \frac{1}{3^l}[i_{l,d-1},\, i_{l,d-1}+1], \tag{B.19}$$

where the subinterval indices $i_{l,\zeta} = 0, 1, \ldots, 3^l$, $\zeta \in \{0, 1, d-1\}$ are defined by (B.9), and where we have introduced the index

$$i^{0,l} = \sum_{\zeta=1}^{d-1} I_{l,\zeta}\, i_{l,\zeta}. \tag{B.20}$$

Here, the $I_{l,\zeta} \in \{1, 3^l\}$, $\zeta \in \{1, 2, \ldots, d-1\}$ denote strides that define a unique, continuous index.

Similar to the previous section, we can project coefficients associated with basis polynomials that use Gauss-Legendre quadrature nodes within the reference face $\hat{C}$ as interpolation points onto the respective coefficients associated with the subfaces $C^{l:i^{0,l}}$ by using a tensor product of the subinterval projectors $\hat{\Pi}^{0,l:i_\zeta^{0:l}}$, $\zeta \in \{0, 1, \ldots, d-1\}$:

$$g_k^{l:i^{0,l}} = \sum_{k'=0}^{(N+1)^{d-1}-1} g_{k'} \prod_{\zeta=1}^{d-1} \hat{\Pi}^{0,l:i_\zeta^{0,l}}_{k'_\zeta, k_\zeta}, \qquad k = 0, 1, \ldots, (N+1)^{d-1} - 1, \tag{B.21}$$

where the values $g_{k'}$, $k = 0, \ldots, (N+1)^{d-1} - 1$, denote coefficients associated with the reference face $\hat{C}$.

B.4.1. Restriction

By demanding that the evaluation of (B.18) leads to the same result independent of if we sum the contributions of the $3^{l\,(d-1)}$ subfaces or if we perform an integration over the reference face $\hat{C}$, we obtain the following L^2-projection to reconstruct the coarse level

(ξ, f)	$(0,0)$	$(0,1)$	$(1,0)$	$(1,1)$	$(2,0)$	$(2,1)$	$\cdots$
c_1	0	2	$\in \{0,1,2\}$	$\in \{0,1,2\}$	$\in \{0,1,2\}$	$\in \{0,1,2\}$	$\cdots$
c_2	$\in \{0,1,2\}$	$\in \{0,1,2\}$	0	2	$\in \{0,1,2\}$	$\in \{0,1,2\}$	$\cdots$
c_3	$\in \{0,1,2\}$	$\in \{0,1,2\}$	$\in \{0,1,2\}$	$\in \{0,1,2\}$	0	2	$\cdots$
$\vdots$	$\vdots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$	$\ddots$

Table B.1.: Indices $(c_1, c_2, \ldots, c_d)$ of subcells that are adjacent to the parent cell face with index (ξ, f). The difference in levels of the subcells and the parent cell is 1.

coefficients:

$$\begin{aligned} g_k &= \sum_{i^{0,l}=0}^{3^{l\,(d-1)}-1} \sum_{k'=0}^{(N+1)^{d-1}-1} \prod_{\zeta=1}^{d-1} \frac{\left\langle \hat{\varphi}_{k_\zeta}, \hat{\varphi}_{k'_\zeta}^{l:i_\zeta^{0:l}} \right\rangle_{L^2(\hat{C}^{0,l:i^{0:l}})}}{\left\langle \hat{\varphi}_{k_\zeta}, \hat{\varphi}_{k_\zeta} \right\rangle_{L^2(\hat{C})}} g_{k'}^{l:i^{0,l}} \\ &= \sum_{i^{0,l}=0}^{3^{l\,(d-1)}-1} \sum_{k'=0}^{(N+1)^{d-1}-1} \prod_{\zeta=1}^{d-1} \frac{\frac{1}{3^l}\, w_{k'_\zeta}\, \hat{\varphi}_{k_\zeta}\left(\frac{1}{3^l}(\hat{x}_{k',\zeta} + i_\zeta^{0,l})\right)}{w_{k_\zeta}} g_{k'}^{l:i^{0,l}} \\ &= \sum_{i^{0,l}=0}^{3^{l\,(d-1)}-1} \sum_{k'=0}^{(N+1)^{d-1}-1} \frac{1}{3^{l\,(d-1)}} \left(\prod_{\zeta=1}^{d-1} \frac{w_{k'_\zeta}}{w_{k_\zeta}}\, \hat{\Pi}_{k_\zeta k'_\zeta}^{0,l:i_\zeta^{0,l}} \right) g_{k'}^{l:i^{0,l}}, \end{aligned} \tag{B.22}$$

for $k = 0, 1, \ldots, (N+1)^{d-1} - 1$.

B.4.2. Determining the subface index from a subcell index

For a given face (ξ, f) of a cell on level $l-1$ and a single-level subcell index $(c_1, c_2, \ldots, c_d)$, $c_\xi \in \{0,1,2\}$, $\xi \in \{0,1,\ldots,d\}$ we can reconstruct the single-level subface indices $j_{l,\zeta}$, $\zeta \in \{0,1,\ldots,d-1\}$ according to Table B.1.

With respect to the topmost face at level $l = 0$ that contains the subface at level l, we can then construct a projector for the respective subface by recursively (bottom-up) determining the subface indices of all intermediate levels and then computing an index (top-down) according to (B.20) and (B.9)

Bibliography

[1] Michael Dumbser, Cedric Enaux, and Eleuterio F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227(8):3971–4001, April 2008.

[2] Michael Dumbser and Martin Käser. An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - II. The three-dimensional isotropic case. *Geophysical Journal International*, 167(1):319–336, October 2006.

[3] Michael Dumbser, Olindo Zanotti, Raphal Loubre, and Steven 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.

[4] Gregor Gassner, Michael Dumbser, Florian Hindenlang, and Claus-Dieter Munz. Explicit one-step time discretizations for discontinuous Galerkin and finite volume schemes based on local predictors. *Journal of Computational Physics*, 230(11):4232–4247, May 2011.

[5] Jan S. Hesthaven and Tim Warburton. *Nodal discontinuous Galerkin methods: algorithms, analysis, and applications*. Number 54 in Texts in applied mathematics. Springer, New York, 2008. OCLC: ocn191889938.

[6] Beatrice. Riviere. *Discontinuous Galerkin methods for solving elliptic and parabolic equations: theory and implementation*. SIAM, Society for Industrial and Applied Mathematics, Philadelphia, PA, 2008.

[7] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer, Dordrecht ; New York, 3rd ed edition, 2009. OCLC: ocn401321914.

[8] Olindo Zanotti, Francesco Fambri, Michael Dumbser, and Arturo Hidalgo. Spacetime adaptive ADER discontinuous Galerkin finite element schemes with a posteriori subcell finite volume limiting. *Computers & Fluids*, 118:204–224, 2015.

List of Figures

List of Tables