



## Technical Report Series Center for Data and Simulation Science

Gero Schnücke, Nico Kraus, Thomas Bolemann, Gregor J. Gassner

Entropy Stable Discontinuous Galerkin Schemes on Moving Meshes  
with Summation-by-Parts Property for Hyperbolic Conservation  
Laws

Technical Report ID: CDS-2019-9

Available at <http://kups.ub.uni-koeln.de/id/eprint/9518>

Submitted on April 11, 2019

# Entropy Stable Discontinuous Galerkin Schemes on Moving Meshes with Summation-by-Parts Property for Hyperbolic Conservation Laws

Gero Schnücke\*, Nico Krais<sup>†</sup>, Thomas Bolemann<sup>‡</sup> and Gregor J. Gassner<sup>§</sup>

December 24, 2018

---

## Abstract

This work is focused on the entropy analysis of a semi-discrete nodal discontinuous Galerkin spectral element method (DGSEM) on moving meshes for hyperbolic conservation laws. The DGSEM is constructed with a local tensor-product Lagrange-polynomial basis computed from Legendre-Gauss-Lobatto (LGL) points. Furthermore, the collocation of interpolation and quadrature nodes is used in the spatial discretization. This approach leads to discrete derivative approximations in space that are summation-by-parts (SBP) operators. On a static mesh, the SBP property and suitable two-point flux functions, which satisfy the entropy condition from Tadmor, allow to mimic results from the continuous entropy analysis on the discrete level. In this paper, Tadmor's condition is extended to the moving mesh framework. Based on the moving mesh entropy condition, entropy conservative two-point flux functions for the homogeneous shallow water equations and the compressible Euler equations are constructed. Furthermore, it will be proven that the semi-discrete moving mesh DGSEM is an entropy conservative scheme when a two-point flux function, which satisfies the moving mesh entropy condition, is applied in the split form DG framework. This proof does not require any exactness of quadrature in the spatial integrals of the variational form. Nevertheless, entropy conservation is not sufficient to tame discontinuities in the numerical solution and thus the entropy conservative moving mesh DGSEM is modified by adding numerical dissipation matrices to the entropy conservative fluxes. Then, the method becomes entropy stable such that the discrete mathematical entropy is bounded at any time by its initial and boundary data when the boundary conditions are specified appropriately.

Besides the entropy stability, the time discretization of the moving mesh DGSEM will be investigated and it will be proven that the moving mesh DGSEM satisfies the free stream preservation property for an arbitrary  $s$ -stage Runge-Kutta method.

The theoretical properties of the moving mesh DGSEM will be validated by numerical experiments for the compressible Euler equations.

**Key Words:** Discontinuous Galerkin, Summation-by-Parts, Moving Meshes, Entropy Stability, Free Stream Preservation

---



---

\* Department for Mathematics and Computer Science, University of Cologne, Email: gschnuec@math.uni-koeln.de

<sup>†</sup> Institute of Aerodynamics and Gas Dynamics (IAG), University of Stuttgart, Email: krais@iag.uni-stuttgart.de

<sup>‡</sup> Department for Mathematics and Computer Science, University of Cologne

Institute of Aerodynamics and Gas Dynamics (IAG), University of Stuttgart, Email: bolemann@iag.uni-stuttgart.de

<sup>§</sup> Department for Mathematics and Computer Science, University of Cologne, Email: ggassner@math.uni-koeln.de  
Center for Data and Simulation Science (CDS), University of Cologne, Url: www.cds.uni-koeln.de

# 1 Introduction

A lot of applications in engineering and physics require the approximation of conservation laws on time-dependent domains, e.g., domains with moving boundaries. For instance, moving mesh discontinuous Galerkin (DG) methods have been investigated in [4, 24, 43, 50, 51]. In particular, moving mesh discontinuous Galerkin spectral element methods (DGSEM) have been constructed and analyzed in [38, 47, 63]. In the literature, there are also moving mesh methods with the capability to change the connectivity of the mesh, e.g. with finite volume (FV) methods [44, 55] and with a DG method [59]. In general, moving mesh methods are well suited to preserve motion related properties like the Galilean-invariance. These properties are necessary to describe physical processes like the formation of disc galaxies [45].

A common way to approximate conservation laws on time-dependent domains is to use the Arbitrary Lagrangian-Eulerian (ALE) approach [17]. In this approach the conservation law is transformed from the time-dependent domain onto a time-independent reference domain. The motion of the mesh on the physical domain is part of the transformation. Thus, the grid velocity field appears as a new quantity in the equation on the reference domain. On the one hand the ALE transformation simplifies the discretization, since a static mesh can be used in the reference domain. On the other hand, the new quantities in the equation on the reference domain complicate the discrete stability analysis, even in the linear case [38]. More precisely, the appearance of the grid velocity field leads to a non-conservative equation. On the continuous level, the geometric quantities and the grid velocity field satisfy an additional balance law. This balance law is often referred to as geometric conservation law (GCL) [18, 30, 41, 42, 46]. The GCL allows to write the transformed equation in conservation form. However, on the discrete level a numerical scheme does not necessarily satisfy a discrete GCL (D-GCL). Farhat et al. [18, 30, 41] proved that the absence of this property has a critical effect on the accuracy and stability of a moving mesh method. In particular, the preservation of constant states is no longer guaranteed.

In this work, moving mesh DGSEM to solve non-linear, symmetrizable and hyperbolic systems of conservation laws are investigated. It is well known that symmetrizable systems are equipped with an entropy/entropy flux pair [29, 48]. For scalar conservation laws, entropy admissible criteria provide the unique physical relevant weak solution [15, 40]. For systems of conservation laws, entropy admissible criteria are not enough to ensure well posedness [12]. Nevertheless, the entropy is an essential quantity to analyze systems of conservation laws. In particular, for gas dynamics a possible mathematical entropy is the scaled negative thermodynamic entropy which shows that the mathematical model correctly captures the second law of thermodynamics [1]. The entropy is conserved for smooth solutions of a conservation law and decays for discontinuous solutions [31, 57].

Therefore, it is desirable to construct numerical schemes for conservation laws which reflect the properties of the entropy on the discrete level. Tadmor [56] developed a discrete entropy criterion to construct a specific class of two-point flux functions for low-order FV methods. FV methods with these class of two-point flux functions preserve entropy on the discrete level. Moreover, these FV methods can be modified by adding dissipation to the numerical fluxes such that the entropy is decreasing for all times. Therefore, two-point fluxes with Tadmor's discrete entropy condition are called entropy conservative fluxes. Fisher and Carpenter [21] proved that low-order entropy conservative fluxes can be used to construct high-order schemes when the derivative approximations in space are summation-by-parts (SBP) operators. A SBP operator provides a discrete analogue of the integration-by-parts formula [20, 25, 39]. It is worth to mention that the derivative matrix in the DGSEM provides a SBP operator, if the tensor-product Lagrange-polynomial basis is computed from Legendre-Gauss-Lobatto (LGL) points and interpolation and quadrature are collocated. Gassner et al. [26, 27] showed that split forms of the partial differential equations can be discretely recovered when specific choices of numerical volume fluxes in the flux form volume integral of Fisher and Carpenter are chosen. Thus, the following restrictions need to be satisfied to construct entropy stable

DGSEM:

- (1) The derivative matrix satisfies the SBP property.
- (2) There are two-point flux functions with Tadmor's discrete entropy condition that can be extended to high-order in a split form DG framework.

This methodology has been used in the construction of high-order entropy stable DGSEM on quadrilateral/hexahedral elements, e.g. [3, 26, 60], or on triangular/tetrahedral elements, e.g. [6, 11, 14]. All these methods are provably entropy stable and the semi-discrete entropy analysis for them is based merely on the properties of the SPB operators and the assumption that the time integration is exact. The exactness of quadrature in the spatial integrals of the variational form is not necessary.

Up to now, the methodology to construct entropy stable DGSEM has been developed for static meshes. The focus of this work is the construction of entropy stable moving mesh DGSEM. The remainder of the paper is organized as follows: In Section 2 the ALE transformation is presented for one dimensional systems of conservation laws. The continuous entropy analysis for the conservation law on the reference domain is given in the Subsection 2.1. Then, in Subsection 2.2 a moving mesh FV scheme is constructed and a natural extension of Tadmor's discrete entropy condition for the ALE framework is presented. The capability of the moving mesh FV scheme to preserve constant states is investigated in the Subsection 2.3. The ALE transformation and continuous entropy analysis is presented in the Subsections 3.1, 3.2 and 3.3. The framework for the spectral element discretization with the SPB operator is given in the Subsection 3.4 and the DG split form framework is presented in the Subsection 3.5. The moving mesh DGSEM is finally presented in the Subsection 3.5. A discrete entropy analysis for the moving mesh DGSEM is given in the Subsections 3.6 and 3.7. Furthermore, in Subsection 3.8 it is proven that the moving mesh DGSEM satisfies the free stream preservation property. In Section 4, numerical examples with the compressible Euler equations are presented to validate our theoretical findings.

## 2 Basic concepts for 1D conservations laws in the ALE framework

In this section, we discuss the ALE transformation for the one dimensional symmetrizable system

$$\frac{\partial \mathbf{u}}{\partial t} + \frac{\partial \mathbf{f}}{\partial x} = \mathbf{0}. \quad (2.1)$$

The system consists of  $p$  equations, is defined on a time dependent interval  $[a(t), b(t)]$  and is equipped with suitable boundary conditions. The time dependent interval has continuous differentiable boundary functions and can be mapped on the reference interval  $[-1, 1]$  by a time-dependent affine linear mapping  $x = \chi(\xi, t)$ . The mapping provides the quantities

$$J(\xi, t) := \frac{\partial}{\partial \xi}(\chi(\xi, t)), \quad \text{and} \quad \nu(\xi, t) := \frac{d}{dt}(\chi(\xi, t)), \quad (2.2)$$

where  $\nu$  is the grid velocity. The chain rule and (2.2) provide

$$J \frac{d\mathbf{u}}{dt} = J \frac{\partial \mathbf{u}}{\partial t} + \nu \frac{\partial \mathbf{u}}{\partial \xi} \quad \Leftrightarrow \quad J \frac{\partial \mathbf{u}}{\partial t} = J \frac{d\mathbf{u}}{dt} - \nu \frac{\partial \mathbf{u}}{\partial \xi}. \quad (2.3)$$

Hence, by applying the chain rule and rearranging terms the conservation law (2.1) becomes

$$J \frac{d\mathbf{u}}{dt} + \frac{\partial \mathbf{f}}{\partial \xi} = \nu \frac{\partial \mathbf{u}}{\partial \xi} \quad (2.4)$$

on the reference element. The product rule allows to write the equation (2.4) as

$$J \frac{d\mathbf{u}}{dt} + \frac{\partial \mathbf{g}}{\partial \xi} = -\frac{\partial \nu}{\partial \xi} \mathbf{u}, \quad (2.5)$$

where  $\mathbf{g} = \mathbf{f} - \nu \mathbf{u}$ . The equations (2.4) and (2.5) are not desirable for a numerical discretization, since the equations are not in conservation form and a source term appears. The discretization of equations with source terms provides additional numerical problems, since the preservation of certain states is not guaranteed [9, 28, 60]. Nevertheless, Schwarz's theorem on the symmetry of second derivatives provides

$$\frac{dJ}{dt} = \frac{d}{dt} \frac{\partial \chi}{\partial \xi} = \frac{\partial}{\partial \xi} \frac{d\chi}{dt} = \frac{\partial \nu}{\partial \xi}. \quad (2.6)$$

The equation (2.6) is typically referred to as the GCL. Thus, equation (2.1) can be written on the reference interval as the conservation law

$$\frac{dJ\mathbf{u}}{dt} + \frac{\partial \mathbf{g}}{\partial \xi} = \mathbf{0}. \quad (2.7)$$

## 2.1 Entropy analysis in one dimension

The well known results by Godunov [29] and Mock [48] state that a system of conservation laws is symmetrizable if and only if it is equipped with a strictly convex entropy function. Hence, it exists an entropy/entropy flux pair  $(s, f^s)$  for the system (2.1). Then the entropy variables are defined by

$$\mathbf{w} := \frac{\partial s}{\partial \mathbf{u}}, \quad (2.8)$$

and the entropy flux satisfies

$$\mathbf{w}^T \left( \frac{\partial \mathbf{f}}{\partial x} \right) = \left( \frac{\partial f^s}{\partial x} \right) \quad \Leftrightarrow \quad \mathbf{w}^T \left( \frac{\partial \mathbf{f}}{\partial \xi} \right) = \left( \frac{\partial f^s}{\partial \xi} \right). \quad (2.9)$$

Thus, the chain rule provides

$$\begin{aligned} \mathbf{w}^T \left( \frac{dJ\mathbf{u}}{dt} \right) &= J \mathbf{w}^T \left( \frac{d\mathbf{u}}{dt} \right) + \left( \frac{dJ}{dt} \right) \mathbf{w}^T \mathbf{u} \\ &= \left( \frac{dJs}{dt} \right) + \left( \frac{dJ}{dt} \right) (\mathbf{w}^T \mathbf{u} - s) \\ &= \left( \frac{dJs}{dt} \right) + \left( \frac{\partial \nu}{\partial \xi} \right) (\mathbf{w}^T \mathbf{u} - s), \end{aligned} \quad (2.10)$$

where we used the GCL (2.6) in the last step. Likewise, the chain rule and equation (2.9) give

$$\begin{aligned} \mathbf{w}^T \left( \frac{\partial \mathbf{g}}{\partial \xi} \right) &= \mathbf{w}^T \left( \frac{\partial \mathbf{f}}{\partial \xi} \right) - \nu \mathbf{w}^T \left( \frac{\partial \mathbf{u}}{\partial \xi} \right) - \left( \frac{\partial \nu}{\partial \xi} \right) \mathbf{w}^T \mathbf{u} \\ &= \left( \frac{\partial f^s}{\partial \xi} \right) - \nu \left( \frac{\partial s}{\partial \xi} \right) - \left( \frac{\partial \nu}{\partial \xi} \right) \mathbf{w}^T \mathbf{u} \\ &= \frac{\partial}{\partial \xi} (f^s - \nu s) - \left( \frac{\partial \nu}{\partial \xi} \right) (\mathbf{w}^T \mathbf{u} - s). \end{aligned} \quad (2.11)$$

For smooth solutions of the conservation law (2.1), the equations (2.10), (2.11), provide the entropy balance law on the reference element

$$0 = \mathbf{w}^T \left( \frac{dJ\mathbf{u}}{dt} \right) + \mathbf{w}^T \left( \frac{\partial \mathbf{g}}{\partial \xi} \right) = \left( \frac{dJs}{dt} \right) + \frac{\partial}{\partial \xi} (f^s - \nu s). \quad (2.12)$$

Next, we integrate the equation (2.12) over the reference element  $[-1, 1]$ , apply the fundamental theorem of calculus and rearrange terms. This provides the identity

$$\frac{d}{dt} \int_{-1}^1 Js \, d\xi = - (f^s - \nu s)|_{-1}^1. \quad (2.13)$$

The equation (2.13) describes the temporal evolution of the total entropy and holds merely for smooth solutions of the conservation law (2.1). Nevertheless, for discontinuous solutions, we obtain the inequality

$$\frac{d}{dt} \int_{-1}^1 J s d\xi \leq - (f^s - \nu s)|_{-1}^1. \quad (2.14)$$

A numerical method, which satisfies a discrete analogue of the equation (2.13), is called entropy conservative and a method, which satisfies a discrete analogue of the equation (2.14), is called entropy stable approximation.

On the discrete level, it is not easy to mimic the chain rule. We focus on SBP operators that allow to recover the integration-by-parts formula on the discrete level [20, 25, 39]. Thus, as an alternative, we re-write (2.11) in such a way that the product rule can be used:

$$\left( \frac{\partial \mathbf{w}}{\partial \xi} \right)^T \mathbf{g} = \left( \frac{\partial \psi}{\partial \xi} \right) - \nu \left( \frac{\partial \phi}{\partial \xi} \right), \quad (2.15)$$

where  $\phi := \mathbf{w}^T \mathbf{u} - s$  and  $\psi := \mathbf{w}^T \mathbf{f} - f^s$  are the entropy potential and the entropy flux potential functions [57].

## 2.2 Discrete entropy conservation for moving mesh FV methods

Next, we consider a FV discretization of the problem (2.1). We divide the domain  $[a(t), b(t)]$  in  $K$  non-overlapping time-dependent elements  $I_k(t) := [x_k(t), x_{k+1}(t)]$ ,  $k = 1, \dots, K$ . Each interval  $I_k(t)$  is mapped onto the reference Interval  $[-1, 1]$  by an affine linear mapping. The approximation for the element mean value of the exact solution  $\mathbf{u}$  in the element  $I_k(t)$  is denoted by  $\mathbf{u}_k(t)$  and the approximation for  $J$  in the element  $I_k(t)$  is denoted by  $J_k$ . The GCL (2.6) needs to be satisfied on the discrete level. Thus, the equation (2.6) is also discretized. Then, the semi-discrete moving mesh FV scheme is given by

$$\frac{dJ_k}{dt} = \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right), \quad (2.16a)$$

$$\frac{d(J_k \mathbf{u}_k)}{dt} = - \left( \mathbf{g}_{k+\frac{1}{2}}^* - \mathbf{g}_{k-\frac{1}{2}}^* \right). \quad (2.16b)$$

The numerical flux functions  $\nu_{k+\frac{1}{2}}^*$  and  $\mathbf{g}_{k+\frac{1}{2}}^*$  in (2.16) satisfy

$$\nu_{k+\frac{1}{2}}^* := \frac{1}{2} \left( \nu_{k+\frac{1}{2}}^L + \nu_{k+\frac{1}{2}}^R \right), \quad \text{and} \quad \mathbf{g}_{k+\frac{1}{2}}^* := \mathbf{g}^* \left( \nu_{k+\frac{1}{2}}^L, \nu_{k+\frac{1}{2}}^R, \mathbf{u}_k, \mathbf{u}_{k+1} \right), \quad (2.17)$$

where  $\nu_{k+\frac{1}{2}}^L$  is the approximation of the grid velocity from the left side in the cell interface point  $x_{k+1}$  and  $\nu_{k+\frac{1}{2}}^R$  is the approximation from the right side. Moreover, the flux function  $\mathbf{g}^*$  needs to satisfy the consistence condition

$$\mathbf{g}^* \left( \nu^L, \nu^R, \mathbf{u}, \mathbf{u} \right) = \mathbf{f}(\mathbf{u}) - \frac{1}{2} \left( \nu^L + \nu^R \right) \mathbf{u}. \quad (2.18)$$

In the FV context, a discrete analogue of the equation (2.15) is given by

$$(\mathbf{w}_{k+1} - \mathbf{w}_k)^T \mathbf{g}_{k+\frac{1}{2}}^* = (\psi_{k+1} - \psi_k) - \nu_{k+\frac{1}{2}}^* (\phi_{k+1} - \phi_k), \quad (2.19)$$

$$\phi_k := \mathbf{w}_k^T \mathbf{u}_k - s(\mathbf{u}_k), \quad \psi_k := \mathbf{w}_k^T \mathbf{f}(\mathbf{u}_k) - f^s(\mathbf{u}_k), \quad (2.20)$$

for all  $k = 1, \dots, K-1$ . We note that for  $\nu = 0$  (static mesh special case) the equation (2.19) becomes Tadmor's discrete entropy condition for entropy conservative numerical flux functions [56]. The following property of the FV method (2.16) is proven in Appendix D.1

**Theorem 2.1.** Suppose the flux function  $\mathbf{g}^*$  satisfies the condition (2.19). Then the semi-discrete moving mesh FV (2.16) scheme is entropy consistent such that

$$\frac{d}{dt} \sum_{k=1}^K J_k s(\mathbf{u}_k) = - \left( \mathbf{w}_K^T \mathbf{g}_{K+\frac{1}{2}}^* - \psi_K + \nu_{K+\frac{1}{2}}^* \psi_K \right) + \left( \mathbf{w}_1^T \mathbf{g}_{\frac{1}{2}}^* - \psi_1 + \nu_{\frac{1}{2}}^* \psi_1 \right). \quad (2.21)$$

The equation (2.21) is a discrete analogue of the identity (2.13). Therefore, in the context of the ALE framework, the condition (2.19) can be seen as a natural extension of Tadmor's discrete entropy condition. Finally, we remark that a flux with the property (2.19) can be constructed from an entropy consistent flux for FV methods on static meshes.

**Remark 2.2.** If a two-point flux function  $\mathbf{f}^*$  for static meshes with the Tadmor condition

$$(\mathbf{w}_{k+1} - \mathbf{w}_k)^T \mathbf{f}_{k+\frac{1}{2}}^* = (\psi_{k+1} - \psi_k), \quad k = 1, \dots, K-1, \quad (2.22)$$

and a state function  $\mathbf{u}^\#$  with the property

$$(\mathbf{w}_{k+1} - \mathbf{w}_k)^T \mathbf{u}_{k+\frac{1}{2}}^\# = (\phi_{k+1} - \phi_k), \quad k = 1, \dots, K-1, \quad (2.23)$$

are available, a flux function  $\mathbf{g}^*$  with the entropy condition (2.19) can be constructed by

$$\mathbf{g}_{k+\frac{1}{2}}^* = \mathbf{f}_{k+\frac{1}{2}}^* - \nu_{k+\frac{1}{2}}^* \mathbf{u}_{k+\frac{1}{2}}^\#. \quad (2.24)$$

State functions  $\mathbf{u}^\#$  with the property (2.23) have been constructed by Friedrich et al. [23] for the shallow water, compressible Euler and ideal magnetohydrodynamic (MHD) equations and are needed for the construction of fully discrete entropy stable space-time DGSEM methods.

### 2.3 Free stream preservation for moving mesh FV methods

The geometric and metric terms are time-dependent in a moving mesh approximation. Thus, it is a priori not clear, if the common used time discretization strategies like Runge-Kutta (RK) methods, e.g. [5], provide a meaningful approximation for these terms. In particular, it is not clear, if the semi-discrete GCL (2.16a) stays true after the time discretization. A simple test to check, if the discretization of the geometric and metric terms is reasonable, is the free stream preservation test. The capability of the numerical scheme to preserve solutions with constant states is investigated in this test.

In the following, the time discretization of the semi-discrete moving mesh FV scheme (2.16) is analyzed. We divide the interval  $[0, T]$  in discrete time levels  $t^n$ . The step size of the time discretization is  $\Delta t$ . The FV solutions  $\mathbf{u}_k$ ,  $J_k$  and the numerical fluxes  $\nu_{k+\frac{1}{2}}^*$ ,  $\mathbf{g}_{k+\frac{1}{2}}^*$  are approximated in the time levels  $t^n$ , e.g.  $\mathbf{u}_k(t^n) \approx \mathbf{u}_k^n$ . Thus, the moving mesh forward Euler FV discretization is given by

$$J_k^{n+1} = J_k^n + \Delta t \left( \nu_{k+\frac{1}{2}}^{*,n} - \nu_{k-\frac{1}{2}}^{*,n} \right), \quad (2.25a)$$

$$(J_k \mathbf{u}_k)^{n+1} = J_k^n \mathbf{u}_k^n - \Delta t \left( \mathbf{g}_{k+\frac{1}{2}}^{*,n} - \mathbf{g}_{k-\frac{1}{2}}^{*,n} \right), \quad (2.25b)$$

$$\mathbf{u}_k^{n+1} = (J_k \mathbf{u}_k)^{n+1} / J_k^{n+1}. \quad (2.25c)$$

The last step is necessary to decouple the solution  $\mathbf{u}_k$  from the metric term  $J_k$ . However, it is also possible to write (2.25) in an equivalent form without the step (2.25c). Therefore, we apply the D-GCL (2.25a) and realize

$$J_k^n \mathbf{u}_k^n = \left( J_k^{n+1} - \Delta t \left( \nu_{k+\frac{1}{2}}^{*,n} - \nu_{k-\frac{1}{2}}^{*,n} \right) \right) \mathbf{u}_k^n. \quad (2.26)$$

Then, we plug (2.26) in (2.25b) and obtain

$$J_k^{n+1} = J_k^n + \Delta t \left( \nu_{k+\frac{1}{2}}^{*,n} - \nu_{k-\frac{1}{2}}^{*,n} \right), \quad (2.27a)$$

$$\mathbf{u}_k^{n+1} = \mathbf{u}_k^n - \frac{\Delta t}{J_k^{n+1}} \left[ \left( \mathbf{g}_{k+\frac{1}{2}}^{*,n} + \nu_{k+\frac{1}{2}}^{*,n} \mathbf{u}_k^n \right) - \left( \mathbf{g}_{k-\frac{1}{2}}^{*,n} + \nu_{k-\frac{1}{2}}^{*,n} \mathbf{u}_k^n \right) \right]. \quad (2.27b)$$

We note that the equation (2.27b) matches the structure of the one dimensional moving mesh FV method from Fazio and LeVeque [19]. Next, we analyze the forward Euler moving mesh FV scheme with respect to the free stream preservation property. For a solution  $\mathbf{u}_k^n = \mathbf{c} = [c_1, \dots, c_p]^T$  with constant components  $c_i \in \mathbb{R}^p$ ,  $i = 1, \dots, p$ , it follows by (2.18)

$$\begin{aligned} \mathbf{u}_k^{n+1} &= \mathbf{c} - \frac{\Delta t}{J_k^{n+1}} \left[ \left( \mathbf{g}_{k+\frac{1}{2}}^{*,n} + \nu_{k+\frac{1}{2}}^{*,n} \mathbf{c} \right) - \left( \mathbf{g}_{k-\frac{1}{2}}^{*,n} + \nu_{k-\frac{1}{2}}^{*,n} \mathbf{c} \right) \right] \\ &= \mathbf{c} - \frac{\Delta t}{J_k^{n+1}} \left[ \left( \left( \mathbf{f}(\mathbf{c}) - \nu_{k+\frac{1}{2}}^{*,n} \mathbf{c} \right) + \nu_{k+\frac{1}{2}}^{*,n} \mathbf{c} \right) - \left( \left( \mathbf{f}(\mathbf{c}) - \nu_{k-\frac{1}{2}}^{*,n} \mathbf{c} \right) + \nu_{k-\frac{1}{2}}^{*,n} \mathbf{c} \right) \right] = \mathbf{c}. \end{aligned} \quad (2.28)$$

Thus, the forward Euler moving mesh FV method satisfies the free stream preservation property.

In the same way, the semi-discrete moving mesh FV scheme (2.16) can be discretized by an explicit  $s$ -stage RK method with the characteristic coefficients  $\{a_{\tau\sigma}\}_{\tau,\sigma=1}^s$ ,  $\{b_\sigma\}_{\sigma=1}^s$ ,  $\{c_\sigma\}_{\sigma=1}^s$ . This provides, the fully-discrete moving mesh RK-FV method:

for  $\tau = 1, \dots, s$ :

$$J_k^{(\tau)} = J_k^n + \Delta t \sum_{\sigma=1}^{\tau-1} a_{\tau\sigma} \left( \nu_{k+\frac{1}{2}}^{*,n+\sigma} - \nu_{k-\frac{1}{2}}^{*,n+\sigma} \right), \quad (2.29a)$$

$$\mathbf{u}_k^{(\tau)} = \mathbf{u}_k^n - \frac{\Delta t}{J_k^{(\tau)}} \sum_{\sigma=1}^{\tau-1} a_{\tau\sigma} \left[ \left( \mathbf{g}_{k+\frac{1}{2}}^{*,n+\sigma} + \nu_{k+\frac{1}{2}}^{*,n+\sigma} \mathbf{u}_k^{(\sigma)} \right) - \left( \mathbf{g}_{k-\frac{1}{2}}^{*,n+\sigma} + \nu_{k-\frac{1}{2}}^{*,n+\sigma} \mathbf{u}_k^{(\sigma)} \right) \right], \quad (2.29b)$$

$$J_k^{n+1} = J_k^n + \Delta t \sum_{j=1}^s b_j \left( \nu_{k+\frac{1}{2}}^{*,n+j} - \nu_{k-\frac{1}{2}}^{*,n+j} \right), \quad (2.29c)$$

$$\mathbf{u}_k^{n+1} = \mathbf{u}_k^n - \frac{\Delta t}{J_k^{n+1}} \sum_{\sigma=1}^s b_\sigma \left[ \left( \mathbf{g}_{k+\frac{1}{2}}^{*,n+\sigma} + \nu_{k+\frac{1}{2}}^{*,n+\sigma} \mathbf{u}_k^{(\sigma)} \right) - \left( \mathbf{g}_{k-\frac{1}{2}}^{*,n+\sigma} + \nu_{k-\frac{1}{2}}^{*,n+\sigma} \mathbf{u}_k^{(\sigma)} \right) \right], \quad (2.29d)$$

where  $\nu_{k+\frac{1}{2}}^{L,n+\sigma} := \nu_{k+\frac{1}{2}}^L(t^n + c_\sigma \Delta t)$ ,  $\nu_{k+\frac{1}{2}}^{R,n+\sigma} := \nu_{k+\frac{1}{2}}^R(t^n + c_\sigma \Delta t)$  and

$$\nu_{k+\frac{1}{2}}^{*,n+\sigma} = \frac{1}{2} \left( \nu_{k+\frac{1}{2}}^{L,n+\sigma} + \nu_{k+\frac{1}{2}}^{R,n+\sigma} \right), \quad (2.30)$$

$$\mathbf{g}_{k+\frac{1}{2}}^{*,n+\sigma} = \mathbf{g}^* \left( \nu_{k+\frac{1}{2}}^{L,n+\sigma}, \nu_{k+\frac{1}{2}}^{R,n+\sigma}, \mathbf{u}_k^{(\sigma)}, \mathbf{u}_k^{(\sigma)} \right). \quad (2.31)$$

The free stream preservation property for the moving mesh RK-FV method (2.29) can be proven by the same calculation as in (2.28). In particular, the following statement can be proven by exactly the same arguments as for the forward Euler step.

**Theorem 2.3.** Let  $\mathbf{u}_k^n = \mathbf{c} := (c_1, \dots, c_p)^T \in \mathbb{R}^p$  be the solution of the fully-discrete FV (2.29) scheme at time level  $t^n$ . Furthermore, the numerical flux  $\mathbf{g}^*$  satisfies (2.18). Then, the constant states  $c_i$ ,  $i = 1, \dots, p$ , are preserved in each Runge-Kutta stage (2.29b). In particular, the solution of the fully-discrete FV method at time level  $t^{n+1}$  is  $\mathbf{u}_k^{n+1} = \mathbf{c}$ .

### 3 Entropy stable DGSEM on moving meshes

In Section 2 an entropy conservative first order moving mesh FV method has been introduced. The main goal of this work is the construction of an entropy stable high order moving mesh DGSEM. On static meshes, it is possible to construct high-order entropy stable DGSEM, if the derivative matrix is an SBP operator and entropy conservative two-point flux functions are available. This methodology has been used in the construction of high-order entropy



stable DGSEM on quadrilateral/hexahedral elements, e.g. [3, 26, 60]. In this section, it will be shown that similar ideas can be used to construct high-order entropy stable moving mesh DGSEM, when there are two-point flux functions with the property (2.19) available for the low-order FV method (2.16). The construction of the entropy stable moving mesh DGSEM will be presented for an arbitrary symmetrizable and hyperbolic system of conservation laws

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^3 \frac{\partial \mathbf{f}_i}{\partial x_i} = \mathbf{0}, \quad (3.1)$$

on a time-dependent domain  $\Omega(t) \subseteq \mathbb{R}^3$ . The vector of conserved variables is  $\mathbf{u}$  and  $\mathbf{f}_i$ ,  $i = 1, 2, 3$ , are the physical flux vectors. The state vectors are of size  $p$  depending on the number of equations in the system under consideration and the conservation law is subjected to appropriate initial and boundary conditions.

The block vector nomenclature in [27] simplifies the analysis of the system (3.1) on curved elements. Thus, we translate the conservation law (3.1) in block vector notation. A block vector is highlighted by the double arrow

$$\overleftrightarrow{\mathbf{f}} := \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \end{bmatrix}. \quad (3.2)$$

The dot product of two block vectors is given by

$$\overleftrightarrow{\mathbf{f}} \cdot \overleftrightarrow{\mathbf{g}} := \sum_{i=1}^3 \mathbf{f}_i^T \mathbf{g}_i. \quad (3.3)$$

Furthermore, the dot product of a vector  $\vec{v}$  in the three dimensional space and a block vector is defined by

$$\vec{v} \cdot \overleftrightarrow{\mathbf{f}} := \sum_{i=1}^3 v_i \mathbf{f}_i. \quad (3.4)$$

We note that the dot product (3.3) is a scalar quantity and the dot product (3.4) is a vector in a  $p$  dimensional space, where the number  $p$  corresponds to the number of conserved variables in the conservation law (3.1). The interaction between a vector  $\vec{v}$  and the conserved variables is defined as the block vector

$$\vec{v} \mathbf{u} := \begin{bmatrix} v_1 \mathbf{u} \\ v_2 \mathbf{u} \\ v_3 \mathbf{u} \end{bmatrix}. \quad (3.5)$$

Thus, in particular, the spatial gradient of the conserved variables is defined by

$$\vec{\nabla}_x \mathbf{u} := \begin{bmatrix} \frac{\partial \mathbf{u}}{\partial x_1} \\ \frac{\partial \mathbf{u}}{\partial x_2} \\ \frac{\partial \mathbf{u}}{\partial x_3} \end{bmatrix}. \quad (3.6)$$

The gradient of a vector valued function  $\vec{g} = [g_1, g_2, g_3]^T$  is a second order tensor, written in matrix form as

$$\vec{\nabla}_x \vec{g} = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \frac{\partial g_1}{\partial x_2} & \frac{\partial g_1}{\partial x_3} \\ \frac{\partial g_2}{\partial x_1} & \frac{\partial g_2}{\partial x_2} & \frac{\partial g_2}{\partial x_3} \\ \frac{\partial g_3}{\partial x_1} & \frac{\partial g_3}{\partial x_2} & \frac{\partial g_3}{\partial x_3} \end{bmatrix}. \quad (3.7)$$

The dot product (3.3) and the spatial gradient (3.6) are used to define the divergence of a block vector flux as

$$\vec{\nabla}_x \cdot \overleftrightarrow{\mathbf{f}} := \sum_{i=1}^3 \frac{\partial \mathbf{f}_i}{\partial x_i}. \quad (3.8)$$

Moreover, for a vector valued function  $\vec{g}$  and the conserved variables, we have the product rule

$$\vec{\nabla}_x \cdot (\vec{g} \mathbf{u}) = (\vec{\nabla}_x \cdot \vec{g}) \mathbf{u} + \vec{g} \cdot (\vec{\nabla}_x \mathbf{u}) \quad (3.9)$$

with respect to the dot products (3.3) and (3.4). These notations allow to write the conservation law (3.1) in the compact form

$$\frac{\partial \mathbf{u}}{\partial t} + \vec{\nabla}_x \cdot \vec{\mathbf{f}} = \mathbf{0}. \quad (3.10)$$

### 3.1 Building blocks of the ALE transformation for hexahedral curved meshes

In order to set up the moving mesh DGSEM in the Section 3.5, we make for all  $t \in [0, T]$  the assumptions:

- (A1) For a fixed number  $K \in \mathbb{N}$  the physical domain  $\Omega(t)$  can be subdivided into  $K$  time-dependent, non-overlapping and conforming hexahedral elements,  $e_\kappa(t)$ ,  $\kappa = 1, \dots, K$ . These elements can have curved faces.
- (A2) The time-dependent elements  $e_\kappa(t)$  are mapped into the spatial computational domain  $E = [-1, 1]^3$  with a bijective isoparametric transfinite mapping. Winters constructed in his PHD thesis [64] a mapping for this set up. Like in [64], it is assumed that the curved faces satisfy for all  $t \in [0, T]$

$$\begin{aligned} \vec{\Gamma}_1(-1, \xi^3, t) &= \vec{\Gamma}_6(-1, \xi^3, t), & \vec{\Gamma}_2(-1, \xi^3, t) &= \vec{\Gamma}_6(1, \xi^3, t), & \vec{\Gamma}_3(-1, \xi^2, t) &= \vec{\Gamma}_6(\xi^2, -1, t), \\ \vec{\Gamma}_1(1, \xi^3, t) &= \vec{\Gamma}_4(-1, \xi^3, t), & \vec{\Gamma}_2(1, \xi^3, t) &= \vec{\Gamma}_4(1, \xi^3, t), & \vec{\Gamma}_3(1, \xi^2, t) &= \vec{\Gamma}_4(\xi^2, -1, t), \\ \vec{\Gamma}_1(\xi^1, -1, t) &= \vec{\Gamma}_3(\xi^1, -1, t), & \vec{\Gamma}_2(\xi^1, -1, t) &= \vec{\Gamma}_3(\xi^1, 1, t), & \vec{\Gamma}_5(-1, \xi^2, t) &= \vec{\Gamma}_6(\xi^2, 1, t), \\ \vec{\Gamma}_1(\xi^1, 1, t) &= \vec{\Gamma}_5(\xi^1, -1, t), & \vec{\Gamma}_2(\xi^1, 1, t) &= \vec{\Gamma}_5(\xi^1, 1, t), & \vec{\Gamma}_5(1, \xi^2, t) &= \vec{\Gamma}_4(\xi^2, 1, t). \end{aligned} \quad (3.11)$$

The location of the curved faces is sketched in Figure 3.1. The curved faces of an element  $e_\kappa(t)$  are approximated

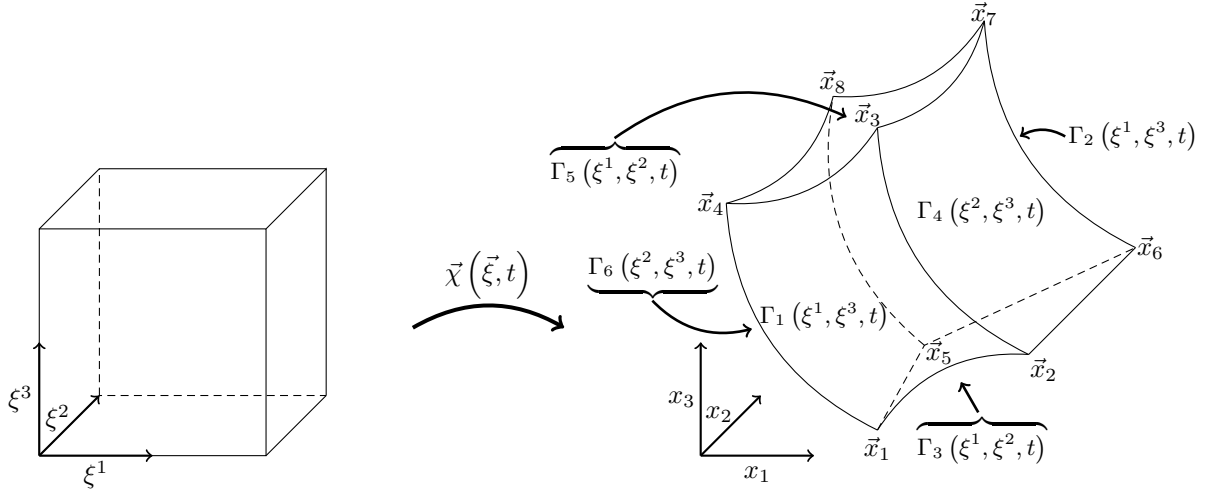


Figure 3.1: Left the reference element  $E = [-1, 1]^3$  and on the right a general hexahedral element  $e_\kappa(t)$  with the curved faces  $\vec{\Gamma}_1(\xi^1, \xi^3, t)$ ,  $\vec{\Gamma}_2(\xi^1, \xi^3, t)$ ,  $\vec{\Gamma}_3(\xi^1, \xi^2, t)$ ,  $\vec{\Gamma}_4(\xi^2, \xi^3, t)$ ,  $\vec{\Gamma}_5(\xi^1, \xi^2, t)$ , and  $\vec{\Gamma}_6(\xi^2, \xi^3, t)$ . The mapping  $\vec{\chi}(\vec{\xi}, t)$  connects  $E$  and  $e_\kappa(t)$ .

as interpolation polynomials up to degree  $N$  such that

$$\mathbb{I}^N(\vec{\Gamma}_i)(\eta, \zeta, t) := \sum_{j,k=0}^N \vec{\Gamma}_i(\eta_j, \zeta_k, t) \ell_j(\eta) \ell_k(\zeta), \quad i = 1, 2, 3, 4, 5, 6, \quad (3.12)$$

where  $\{\ell_j\}_{j=0}^N$ ,  $\{\ell_k\}_{k=0}^N$  are the Lagrange polynomials associated with the interpolation points  $\{\eta_j\}_{j=0}^N$  and  $\{\zeta_k\}_{k=0}^N$ .

(A3) The determinant  $J$  of the Jacobian matrix  $\vec{\nabla}_{\vec{\xi}} \vec{\chi}$  satisfies

$$J := \det \left( \vec{\nabla}_{\vec{\xi}} \vec{\chi} \right) > 0, \quad \forall t \in [0, T]. \quad (3.13)$$

Mesh curving techniques are discussed by Hindenlang et al. [32] and methodologies to construct a moving mesh with the properties (A1)-(A3) are given in the literature e.g. the book of Huang and Russell [33] Chapter 6, Chapter 7]. The mapping  $\vec{\chi} = [\chi_1, \chi_2, \chi_3]^T$  provides the grid velocity field

$$\vec{\nu} = [\nu_1, \nu_2, \nu_3]^T := \left[ \frac{d\chi_1}{dt}, \frac{d\chi_2}{dt}, \frac{d\chi_3}{dt} \right]^T = \frac{d\vec{\chi}}{dt}. \quad (3.14)$$

It is desirable that the grid velocity is continuous, since the mesh should be conforming and watertight at each time level. The next statement provides conditions on the element boundaries to guarantee that the grid velocity becomes continuous.

**Lemma 3.1** Let  $e_1(t)$  and  $e_2(t)$  be two neighboring elements which share one of the faces

$$\vec{\Gamma}_1^1 = \vec{\Gamma}_2^2, \quad \vec{\Gamma}_3^1 = \vec{\Gamma}_5^2, \quad \vec{\Gamma}_4^1 = \vec{\Gamma}_6^2, \quad \vec{\Gamma}_1^2 = \vec{\Gamma}_2^1, \quad \vec{\Gamma}_3^2 = \vec{\Gamma}_5^1, \quad \vec{\Gamma}_4^2 = \vec{\Gamma}_6^1, \quad (3.15)$$

where  $\vec{\Gamma}_i^l$ ,  $l = 1, 2$ , and  $i = 1, 2, 3, 4, 5, 6$ , are the faces of the element  $e_l(t)$ . Furthermore, suppose that the faces  $\vec{\Gamma}_i^l(\cdot, \cdot, t)$  are continuously differentiable in the time interval  $[0, T]$ . Then the grid velocity field is continuous in the points which belong to the face that the elements share.

In Appendix A the Lemma 3.1 is proven in two dimensions. The three dimensional proof can be done by the same argumentation.

### 3.2 Transformation of the conservation law onto a reference element

In the following, we show that the system (3.10) can be transformed from a time-dependent element  $e_\kappa(t)$  onto the reference element  $E$ . The mapping  $\vec{\chi} = [\chi_1, \chi_2, \chi_3]^T$  provides the covariant basis vectors

$$\vec{a}_i := \frac{\partial \vec{\chi}}{\partial \xi^i}, \quad i = 1, 2, 3, \quad (3.16)$$

and the volume weighted contravariant vectors

$$J a^i = a_j \times a_k, \quad (i, j, k) \text{ cyclic}. \quad (3.17)$$

Thereby, is  $\vec{\xi} = (\xi^1, \xi^2, \xi^3)^T$  a vector in the reference element  $E = [-1, 1]^3$ . The covariant and the volume weighted contravariant vectors represent the Jacobian matrix  $\vec{\nabla}_{\vec{\xi}} \vec{\chi}$  and its adjoint matrix

$$\vec{\nabla}_{\vec{\xi}} \vec{\chi} = \begin{bmatrix} \vec{a}_1 & \vec{a}_2 & \vec{a}_3 \end{bmatrix}, \quad \text{adj} \left( \vec{\nabla}_{\vec{\xi}} \vec{\chi} \right) = \begin{bmatrix} (J \vec{a}^1)^T \\ (J \vec{a}^2)^T \\ (J \vec{a}^3)^T \end{bmatrix}. \quad (3.18)$$

Furthermore, the contravariant vectors satisfy the metric identities

$$\sum_{i=1}^3 \frac{\partial J \vec{a}^i}{\partial \xi^i} = 0. \quad (3.19)$$

In particular, the covariant and the contravariant vectors allow to transform differential operators on the time-independent reference element  $E$ . On the reference element the gradient of a function  $f$  is given by

$$\vec{\nabla}_x f = \frac{1}{J} \left( \sum_{i=1}^3 J \vec{a}^i \frac{\partial f}{\partial \xi^i} \right) = \frac{1}{J} \left[ \text{adj} \left( \vec{\nabla}_{\vec{\xi}} \vec{\chi} \right) \right]^T \vec{\nabla}_{\vec{\xi}} f \quad (3.20)$$

and the divergence of a vector valued function  $\vec{g}$  is given by

$$\vec{\nabla}_x \cdot \vec{g} = \frac{1}{J} \sum_{i=1}^3 \frac{\partial}{\partial \xi^i} (J \vec{a}^i \cdot \vec{g}) = \frac{1}{J} \vec{\nabla}_\xi \cdot \vec{g}, \quad (3.21)$$

where we used the contravariant flux

$$\vec{g} := \begin{bmatrix} J \vec{a}^1 \cdot \vec{g} \\ J \vec{a}^2 \cdot \vec{g} \\ J \vec{a}^3 \cdot \vec{g} \end{bmatrix} = \text{adj} \left( \vec{\nabla}_\xi \vec{\chi} \right) \vec{g}. \quad (3.22)$$

In [27], the following block matrix has been introduced to combine the transformations (3.20) and (3.21) with the block vector notation

$$\mathcal{M} = \left[ \text{adj} \left( \vec{\nabla}_\xi \vec{\chi} \right) \right]^T \otimes \mathcal{I}_p = \begin{bmatrix} J a_1^1 \mathcal{I}_p & J a_1^2 \mathcal{I}_p & J a_1^3 \mathcal{I}_p \\ J a_2^1 \mathcal{I}_p & J a_2^2 \mathcal{I}_p & J a_2^3 \mathcal{I}_p \\ J a_3^1 \mathcal{I}_p & J a_3^2 \mathcal{I}_p & J a_3^3 \mathcal{I}_p \end{bmatrix}, \quad (3.23)$$

where  $\otimes$  is the Kronecker product, the matrix  $\mathcal{I}_p$  is the  $p \times p$  identity matrix and  $J a_j^i$  is the component of  $J \vec{a}^i$  in the  $j$ -th Cartesian coordinate direction. The transformation of the gradient becomes

$$\vec{\nabla}_x \mathbf{u} = \frac{1}{J} \mathcal{M} \vec{\nabla}_\xi \mathbf{u}. \quad (3.24)$$

We note that for a vector valued function  $\vec{g}$  the following identity holds

$$\vec{g} \cdot \vec{\nabla}_x \mathbf{u} = \frac{1}{J} \vec{g} \cdot \mathcal{M} \vec{\nabla}_\xi \mathbf{u} = \frac{1}{J} \vec{g} \cdot \vec{\nabla}_\xi \mathbf{u}. \quad (3.25)$$

Moreover, by applying the metric identities (3.19), the transformation of the divergence can be written as

$$\vec{\nabla}_x \cdot \vec{\mathbf{f}} = \frac{1}{J} \vec{\nabla}_\xi \cdot \mathcal{M}^T \vec{\mathbf{f}}. \quad (3.26)$$

Hence, the contravariant block vector flux is given by

$$\vec{\mathbf{f}} := \begin{bmatrix} J \vec{a}^1 \cdot \vec{\mathbf{f}} \\ J \vec{a}^2 \cdot \vec{\mathbf{f}} \\ J \vec{a}^3 \cdot \vec{\mathbf{f}} \end{bmatrix} = \mathcal{M}^T \vec{\mathbf{f}}. \quad (3.27)$$

Since the elements  $\{e_k(t)\}_{k=1}^K$  are time-dependent, the time evolution of the quantity  $J$  needs to be analyzed. Thus, we apply Jacobi's formula (cf. e.g. Bellman [2]) and obtain by (3.14), (3.18)

$$\frac{dJ}{dt} = \text{tr} \left[ \text{adj} \left( \vec{\nabla}_\xi \vec{\chi} \right) \frac{d}{dt} \left( \vec{\nabla}_\xi \vec{\chi} \right) \right] = \sum_{i=1}^3 J \vec{a}^i \cdot \left( \frac{d \vec{a}^i}{dt} \right) = \sum_{i=1}^3 J \vec{a}^i \cdot \left( \frac{\partial \vec{v}}{\partial \xi^i} \right), \quad (3.28)$$

where  $\text{tr}[\cdot]$  denotes the trace of a matrix. The metric identities (3.19) allow to write the equation (3.28) in conservation form

$$\frac{dJ}{dt} = \sum_{i=1}^3 \frac{\partial}{\partial \xi^i} (J \vec{a}^i \cdot \vec{v}) = \vec{\nabla}_\xi \cdot \vec{v}. \quad (3.29)$$

The equation (3.29) is the GCL in three dimensions. The chain rule formula and the identity (3.25) provide

$$\frac{d\mathbf{u}}{dt} = \frac{\partial \mathbf{u}}{\partial t} + \frac{1}{J} \vec{v} \cdot \vec{\nabla}_\xi \mathbf{u}. \quad (3.30)$$

Next, we plug the GCL (3.29) into equation (3.30), apply the product rule (3.9) and rearrange. This provides the equation

$$J \frac{\partial \mathbf{u}}{\partial t} = \frac{d(J\mathbf{u})}{dt} - \vec{\nabla}_\xi \cdot (\vec{v} \mathbf{u}). \quad (3.31)$$

Finally, we combine the identities (3.26) and (3.31) to write the conservation law (3.10) in the following form

$$\frac{d(J\mathbf{u})}{dt} + \vec{\nabla}_\xi \cdot \overleftrightarrow{\mathbf{g}} = \mathbf{0}, \quad (3.32)$$

where

$$\overleftrightarrow{\mathbf{g}} = \begin{bmatrix} \mathbf{g}_1 \\ \mathbf{g}_2 \\ \mathbf{g}_3 \end{bmatrix} := \begin{bmatrix} \mathbf{f}_1 - \nu_1 \mathbf{u} \\ \mathbf{f}_2 - \nu_2 \mathbf{u} \\ \mathbf{f}_3 - \nu_3 \mathbf{u} \end{bmatrix} = \overleftrightarrow{\mathbf{f}} - \vec{\nu} \mathbf{u}. \quad (3.33)$$

The formulation (3.32) is the representation of the system (3.10) on the time-independent reference element  $E$  for a time-dependent element  $e_\kappa(t)$ .

### 3.3 Entropy analysis in three dimensions

The system (3.1) is assumed to be symmetrizable. Thus, in particular, it is equipped with entropy/entropy flux pairs  $(s, f_i^s)$ ,  $i = 1, 2, 3$ , (cf. Godunov [29] and Mock [48]). The strictly convex function  $s$  is the entropy function. The entropy function  $s$  provides the entropy variables

$$\mathbf{w} := \frac{\partial s}{\partial \mathbf{u}}, \quad (3.34)$$

and it follows by the chain rule

$$\frac{\partial s}{\partial t} = \mathbf{w}^T \frac{\partial \mathbf{u}}{\partial t}, \quad \frac{\partial s}{\partial x_i} = \mathbf{w}^T \frac{\partial \mathbf{u}}{\partial x_i}, \quad i = 1, 2, 3. \quad (3.35)$$

The entropy flux functions and the flux functions in the conservation law are related and satisfy

$$\mathbf{w}^T \frac{\partial \mathbf{f}_i}{\partial x_i} = \frac{\partial f_i^s}{\partial x_i} \quad i = 1, 2, 3. \quad (3.36)$$

The identities (3.25) and (3.35) give

$$\mathbf{w}^T \left( \vec{\nu} \cdot \vec{\nabla}_\xi \mathbf{u} \right) = J \mathbf{w}^T \left( \vec{\nu} \cdot \vec{\nabla}_x \mathbf{u} \right) = J \left( \vec{\nu} \cdot \vec{\nabla}_x s \right) = \vec{\nu} \cdot \vec{\nabla}_\xi s. \quad (3.37)$$

Hence, we obtain with the identity (3.30) and the chain rule

$$J \mathbf{w}^T \frac{d\mathbf{u}}{dt} = J \frac{\partial s}{\partial t} + \vec{\nu} \cdot \vec{\nabla}_\xi s = J \frac{ds}{dt}. \quad (3.38)$$

Therefore, the product rule provides the identity

$$\begin{aligned} \mathbf{w}^T \frac{d(J\mathbf{u})}{dt} &= J \frac{ds}{dt} + \left( \frac{dJ}{dt} \right) \mathbf{w}^T \mathbf{u} \\ &= \frac{d(Js)}{dt} + \left( \frac{dJ}{dt} \right) (\mathbf{w}^T \mathbf{u} - s) \\ &= \frac{d(Js)}{dt} + \left( \vec{\nabla}_\xi \cdot \vec{\nu} \right) (\mathbf{w}^T \mathbf{u} - s), \end{aligned} \quad (3.39)$$

where we used the GCL (3.29) in the last step. Next, we apply the relation (3.36) for the entropy flux functions and obtain

$$\mathbf{w}^T \frac{\partial \mathbf{g}_i}{\partial x_i} = \frac{\partial}{\partial x} (f_i^s - \nu_i s) - \left( \frac{\partial \nu_i}{\partial x_i} \right) (\mathbf{w}^T \mathbf{u} - s), \quad i = 1, 2, 3, \quad (3.40)$$

by the same arguments as in the one dimensional computation (2.11). Next, we apply the vector notation  $\vec{f}^s := [f_1^s, f_2^s, f_3^s]^T$ . Then (3.40) and the transformation formulas for the gradient and divergence in the Section 3.2 give

$$\mathbf{w}^T \left( \vec{\nabla}_\xi \cdot \overleftrightarrow{\mathbf{g}} \right) = \vec{\nabla}_\xi \cdot \left( \vec{f}^s - \vec{\nu} s \right) - \left( \vec{\nabla}_\xi \cdot \vec{\nu} \right) (\mathbf{w}^T \mathbf{u} - s). \quad (3.41)$$

Finally, the identities (3.39) and (3.41) provide the balance law

$$0 = \mathbf{w}^T \left( \frac{d(J\mathbf{u})}{dt} + \vec{\nabla}_\xi \cdot \vec{\mathbf{g}} \right) = \frac{d(Js)}{dt} + \vec{\nabla}_\xi \cdot (\vec{f}^s - \vec{v}s). \quad (3.42)$$

We integrate the equation (3.42) over the domain  $E \times [0, T]$  and obtain

$$\frac{d}{dt} \int_E Js d\xi = - \int_{\partial E} (\vec{f}^s - \vec{v}s)^T \hat{n} dS. \quad (3.43)$$

For discontinuous solutions the equation (3.43) is not satisfied, but it holds the inequality

$$\frac{d}{dt} \int_E Js d\xi \leq - \int_{\partial E} (\vec{f}^s - \vec{v}s)^T \hat{n} dS. \quad (3.44)$$

### 3.4 Buidling blocks for the spectral element approximation

The spectral element approximation based on a nodal approach with Lagrange basis functions given by

$$\ell_j(\xi) := \prod_{i=0, i \neq j}^N \frac{\xi - \xi_i}{\xi_j - \xi_i}, \quad j = 0, \dots, N, \quad (3.45)$$

where the nodal points  $\{\xi_i\}_{i=0}^N$  are the LGL points. We note that  $\xi_0 = -1$  and  $\xi_N = 1$ . The Lagrange basis functions satisfy the cardinal property

$$\ell_i(\xi_j) = \delta_{ji}, \quad (3.46)$$

where  $\delta_{ji}$  is the Kronecker delta. On the reference element  $E = [-1, 1]^3$  the solution and fluxes of the system (3.32) are approximated by tensor product Lagrange polynomials of degree  $N$ , e.g.,

$$\mathbf{u}(\xi^1, \xi^2, \xi^3, t) \approx \mathbf{U}(\xi^1, \xi^2, \xi^3, t) := \sum_{i,j,k=0}^N \mathbf{U}_{ijk}(t) \ell_i(\xi^1) \ell_j(\xi^2) \ell_k(\xi^3). \quad (3.47)$$

In the following, polynomial approximations are highlighted by capital letters, e.g.,  $\mathbf{U}$  is an approximation for the state vector  $\mathbf{u}$  and  $\mathbf{F}_l$ ,  $l = 1, 2, 3$ , are approximations for the fluxes  $\mathbf{f}_l$ ,  $l = 1, 2, 3$ . The determinant  $J$  of the Jacobian matrix  $\vec{\nabla}_\xi \vec{\chi}$  is also approximated by tensor product Lagrange polynomials

$$J(\xi^1, \xi^2, \xi^3, t) \approx J(\xi^1, \xi^2, \xi^3, t) := \sum_{i,j,k=0}^N J_{ijk}(t) \ell_i(\xi^1) \ell_j(\xi^2) \ell_k(\xi^3). \quad (3.48)$$

In particular, the interpolation operator for a function  $\mathbf{g}$  is given by

$$\mathbb{I}^N(\mathbf{g})(\xi^1, \xi^2, \xi^3) = \sum_{i,j,k=0}^N \mathbf{g}_{ijk} \ell_i(\xi^1) \ell_j(\xi^2) \ell_k(\xi^3), \quad (3.49)$$

where  $\mathbf{g}_{ijk} := \mathbf{g}(\xi_i^1, \xi_j^2, \xi_k^3)$  and  $\{\xi_i^1\}_{i=0}^N$ ,  $\{\xi_i^2\}_{i=0}^N$ ,  $\{\xi_i^3\}_{i=0}^N$  are sets of LGL points. Derivatives are approximated by exact differentiation of the polynomial interpolants. In general we have  $(\mathbb{I}^N(g))' \neq \mathbb{I}^{N-1}(g')$  (cf. e.g. [7, 37]), as differentiation and interpolation only commute if there are no discretization errors. However, the contravariant coordinate vectors need to be discretized in such a way that the metric identities (3.19) are satisfied on the discrete level, too. Kopriva [36] introduced the conservative curl form that computes

$$J\vec{a}_\beta^\alpha := -\hat{x}_\alpha \cdot \nabla_\xi \times (\mathbb{I}^N(\chi_\gamma \nabla_\delta \chi_m)), \quad \alpha = 1, 2, 3, \quad \beta = 1, 2, 3, \quad (\beta, \gamma, \delta) \text{ cyclic}, \quad (3.50)$$

to approximate the metric terms. Here  $\vec{\chi} = [\chi_1, \chi_2, \chi_3]^T$  represents the mapping from the element to the reference element and  $\hat{x}_i$  is the unit vector in the  $i$ -th Cartesian coordinate direction. The representation (3.50) ensures that

$$\sum_{\alpha=1}^3 \frac{\partial \mathbb{I}^N(Ja_\beta^\alpha)}{\partial \xi^\alpha} = 0, \quad \beta = 1, 2, 3. \quad (3.51)$$

We note that in the LGL points  $\xi_i^1, \xi_j^2, \xi_k^3, i, j, k = 0, \dots, N$ , the equation (3.51) gives

$$\sum_{m=0}^N \left( \mathcal{D}_{im} (Ja_\beta^1)_{mjk} + \mathcal{D}_{jm} (Ja_\beta^2)_{imk} + \mathcal{D}_{km} (Ja_\beta^3)_{ijm} \right) = 0, \quad \beta = 1, 2, 3. \quad (3.52)$$

Integrals are approximated by a tensor product extension of a  $2N - 1$  accurate LGL quadrature formula. Hence, interpolation and quadrature nodes are collocated. In one spatial dimension the LGL quadrature formula is given by

$$\int_{-1}^1 g(\xi) d\xi \approx \sum_{i=0}^N \omega_i g(\xi_i) = \sum_{i=0}^N \omega_i g_i, \quad (3.53)$$

where  $\omega_i, i = 0, \dots, N$ , are the quadrature weights and  $\xi_i, i = 0, \dots, N$ , are the LGL quadrature points. The formula (3.53) motivates the definition of the inner product notation

$$\langle \mathbf{f}, \mathbf{g} \rangle_N := \sum_{i=0}^N \sum_{j=0}^N \sum_{k=0}^N \omega_i \omega_j \omega_k \mathbf{f}_{ijk}^T \mathbf{g}_{ijk} = \sum_{i,j,k=0}^N \omega_{ijk} \mathbf{f}_{ijk}^T \mathbf{g}_{ijk} \quad (3.54)$$

for two functions  $\mathbf{f}$  and  $\mathbf{g}$ . We note that the inner product (3.54) satisfies

$$\langle \mathbb{I}^N(\mathbf{g}), \boldsymbol{\varphi} \rangle_N = \langle \mathbf{g}, \boldsymbol{\varphi} \rangle_N, \quad \forall \boldsymbol{\varphi} \in \mathbb{P}^N(E, \mathbb{R}^p). \quad (3.55)$$

Furthermore, for a block vector  $\overset{\leftrightarrow}{\mathbf{F}}$  and test functions  $\boldsymbol{\varphi} \in \mathbb{P}^N(E, \mathbb{R}^p)$ , we define the discrete surface integral

$$\begin{aligned} \int_{\partial E, N} \boldsymbol{\varphi}^T \left\{ \overset{\leftrightarrow}{\mathbf{F}} \cdot \hat{n} \right\} dS &:= \sum_{j,k=0}^N \omega_j \omega_k \left( \boldsymbol{\varphi}_{Njk}^T (\mathbf{F}_1)_{Njk} - \boldsymbol{\varphi}_{0jk}^T (\mathbf{F}_1)_{0jk} \right) \\ &+ \sum_{i,k=0}^N \omega_i \omega_k \left( \boldsymbol{\varphi}_{iNk}^T (\mathbf{F}_2)_{iNk} - \boldsymbol{\varphi}_{i0k}^T (\mathbf{F}_2)_{i0k} \right) \\ &+ \sum_{i,j=0}^N \omega_i \omega_j \left( \boldsymbol{\varphi}_{ijN}^T (\mathbf{F}_3)_{ijN} - \boldsymbol{\varphi}_{ij0}^T (\mathbf{F}_3)_{ij0} \right), \end{aligned} \quad (3.56)$$

where  $\hat{n}$  is the unit outward normal at the faces of the reference element  $E$ .

The spectral element approximation with LGL points for interpolation and quadrature provides a SBP operator  $\mathcal{Q} = \mathcal{M} \mathcal{D}$  with the mass matrix  $\mathcal{M}$  and the derivative matrix  $\mathcal{D}$ . The mass matrix and the derivative matrix are given by

$$\mathcal{M}_{ij} = \omega_i \delta_{ij}, \quad \mathcal{D}_{ij} = \ell'_j(\xi_i) \quad i, j = 0, \dots, N. \quad (3.57)$$

A SBP operator satisfies the property

$$\mathcal{Q} + \mathcal{Q}^T = \mathcal{B}, \quad (3.58)$$

where  $\mathcal{B} = \text{diag}(-1, 0, \dots, 0, 1)$ . A SBP operator provides a discrete analogue of the integration-by-parts formula [20, 25, 39].

### 3.5 The semi-discrete discontinuous Galerkin method

Now, we apply the notation introduced in Section 3.4 and construct a moving mesh DGSEM. Like for the moving mesh FV scheme (2.16), we discretize the equations (3.29) and (3.32) simultaneous. In this way, it is ensured that the GCL (3.29) is satisfied on the discrete level [38, 47, 63]. First, we replace the solution  $\mathbf{u}$  by (3.47), the Jacobian  $J$  by (3.48) and approximate the fluxes by the interpolation operator (3.49). Next, we multiply the GCL (3.29) with test functions  $\boldsymbol{\varphi} \in \mathbb{P}^N(E)$ , the equation (3.32) with  $\boldsymbol{\varphi} \in \mathbb{P}^N(E, \mathbb{R}^p)$ , integrate the resulting equations and use

integration-by-parts to separate boundary and volume contributions. The volume integrals in the variational form are approximated with the LGL quadrature. Then, we insert numerical surface fluxes  $\tilde{\nu}^*$  and  $\tilde{\mathbf{G}}^*$  at the spatial element interfaces. Afterwards, we use the SBP property (3.58) for the volume contribution to get the standard DGSEM in strong form:

$$\left\langle \frac{dJ}{dt}, \varphi \right\rangle_N = \left\langle \tilde{\nabla}_\xi \cdot \mathbb{I}^N(\tilde{\nu}), \varphi \right\rangle_N + \int_{\partial E, N} \varphi (\tilde{\nu}_n^* - \tilde{\nu}_n) dS, \quad \forall \varphi \in \mathbb{P}^N(E), \quad (3.59a)$$

$$\left\langle \frac{d(J\mathbf{U})}{dt}, \varphi \right\rangle_N = - \left\langle \tilde{\nabla}_\xi \cdot \mathbb{I}^N(\tilde{\mathbf{G}}), \varphi \right\rangle_N - \int_{\partial E, N} \varphi^T (\tilde{\mathbf{G}}_n^* - \tilde{\mathbf{G}}_n) dS, \quad \forall \varphi \in \mathbb{P}^N(E, \mathbb{R}^p). \quad (3.59b)$$

where we used the inner product notation (3.54) and the notation (3.56) for the discrete surface integral.

The approximation of  $\tilde{\nu}$  and the nonlinear flux  $\tilde{\mathbf{g}}$  by the interpolation operator (3.49) causes aliasing errors in the standard strong form. The aliasing errors cannot be bounded and the errors are independent of the choice of the numerical surface flux. In Gassner [25] a detailed explanation and analysis of the aliasing problem is given. Furthermore, a specific reformulation of the volume integrals by using the skew-symmetry strategy has been developed to fix the aliasing problem. This approach has been enhanced and generalized by Gassner et al. in [27, 26] with a technique developed for high-order finite difference (FD) schemes (cf. Fisher and Carpenter [21]). The generalized approach is called split form DG framework. Here, we proceed similar as in [27] and replace the interpolation operators in the discrete volume integrals by derivative projection operators. The interpolation operator in the discrete equation for the GCL (3.29) is replaced by

$$\begin{aligned} \tilde{\mathbb{D}}^N \cdot \tilde{\nu}_{ijk} &:= \sum_{m=0}^N 2 \mathcal{D}_{im} \{ \tilde{\nu} \}_{(i,m)jk} \cdot \{ \mathcal{J} \tilde{a}^1 \}_{(i,m)jk} \\ &\quad + 2 \mathcal{D}_{jm} \{ \tilde{\nu} \}_{i(j,m)k} \cdot \{ \mathcal{J} \tilde{a}^2 \}_{i(j,m)k} \\ &\quad + 2 \mathcal{D}_{km} \{ \tilde{\nu} \}_{ij(k,m)} \cdot \{ \mathcal{J} \tilde{a}^3 \}_{ij(k,m)} \end{aligned} \quad (3.60)$$

with the volume averages of the metric terms, e.g.

$$\{ \cdot \}_{(i,m)jk} := \frac{1}{2} \left[ (\cdot)_{ijk} + (\cdot)_{mjk} \right]. \quad (3.61)$$

The derivative projection operator in the discrete equation for (3.32) is computed as in [27]. Thus, the operator is given by

$$\begin{aligned} \tilde{\mathbb{D}}^N \cdot \tilde{\mathbf{G}}_{ijk}^{\leftrightarrow \text{EC}} &:= \sum_{m=0}^N 2 \mathcal{D}_{im} \left( \tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}(\tilde{\nu}_{ijk}, \tilde{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \cdot \{ \mathcal{J} \tilde{a}^1 \}_{(i,m)jk} \right) \\ &\quad + 2 \mathcal{D}_{jm} \left( \tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}(\tilde{\nu}_{ijk}, \tilde{\nu}_{imk}, \mathbf{U}_{ijk}, \mathbf{U}_{imk}) \cdot \{ \mathcal{J} \tilde{a}^2 \}_{i(j,m)k} \right) \\ &\quad + 2 \mathcal{D}_{km} \left( \tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}(\tilde{\nu}_{ijk}, \tilde{\nu}_{ijm}, \mathbf{U}_{ijk}, \mathbf{U}_{ijm}) \cdot \{ \mathcal{J} \tilde{a}^3 \}_{ij(k,m)} \right). \end{aligned} \quad (3.62)$$

The flux  $\tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}$  is consistent and symmetric such that, e.g.

$$\tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}(\tilde{\nu}_{ijk}, \tilde{\nu}_{mjk}, \mathbf{U}, \mathbf{U}) = \tilde{\mathbf{F}}(\mathbf{U}) - \{ \tilde{\nu} \}_{(i,m)jk} \mathbf{U}, \quad (3.63)$$

and

$$\tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}(\tilde{\nu}_{ijk}, \tilde{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) = \tilde{\mathbf{G}}^{\leftrightarrow \text{EC}}(\tilde{\nu}_{mjk}, \tilde{\nu}_{ijk}, \mathbf{U}_{mjk}, \mathbf{U}_{ijk}), \quad (3.64)$$

for  $i, j, k, m = 0, \dots, N$ . Furthermore, the flux functions  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , satisfy in the LGL points the moving mesh FV entropy condition (2.19). More precisely for  $i, j, k, m = 0, \dots, N$ , the following discrete entropy conditions are



satisfied

$$\begin{aligned}\llbracket \mathbf{W} \rrbracket_{(i,m)jk}^T \mathbf{G}_l^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) &= \llbracket \Psi_l \rrbracket_{(i,m)jk} - \{\!\!\{ \nu_l \}\!\!\}_{(i,m)jk} \llbracket \Phi \rrbracket_{(i,m)jk}, \\ \llbracket \mathbf{W} \rrbracket_{(j,m)k}^T \mathbf{G}_l^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{imk}, \mathbf{U}_{ijk}, \mathbf{U}_{imk}) &= \llbracket \Psi_l \rrbracket_{(j,m)k} - \{\!\!\{ \nu_l \}\!\!\}_{(j,m)k} \llbracket \Phi \rrbracket_{(j,m)k}, \\ \llbracket \mathbf{W} \rrbracket_{ij(k,m)}^T \mathbf{G}_l^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{ijm}, \mathbf{U}_{ijk}, \mathbf{U}_{ijm}) &= \llbracket \Psi_l \rrbracket_{ij(k,m)} - \{\!\!\{ \nu_l \}\!\!\}_{ij(k,m)} \llbracket \Phi \rrbracket_{ij(k,m)}.\end{aligned}\quad (3.65)$$

The quantities  $\Phi$  and  $\Psi_l$  are polynomial approximations which satisfy in the LGL points

$$\Phi_{ijk} = [\mathbf{w}(\mathbf{U})]_{ijk}^T \mathbf{U}_{ijk} - S_{ijk}, \quad (\Psi_l)_{ijk} := [\mathbf{w}(\mathbf{U})]_{ijk}^T (\mathbf{F}_l)_{ijk} - (F_l^s)_{ijk}, \quad l = 1, 2, 3, \quad (3.66)$$

where  $\mathbf{W}_{ijk}$ ,  $S_{ijk}$  and  $(F_l^s)_{ijk}$  are the nodal values of the polynomials

$$\mathbf{W} := \mathbb{I}^N(\mathbf{w}(\mathbf{U})), \quad S := \mathbb{I}^N(s(\mathbf{U})), \quad F_l^s := \mathbb{I}^N(f_l^s(\mathbf{U})), \quad l = 1, 2, 3. \quad (3.67)$$

Here,  $s$  represents an entropy for the system (3.1) with the corresponding entropy flux functions  $f_l^s$ ,  $l = 1, 2, 3$ , and entropy variables  $\mathbf{w}$ . Furthermore, the volume jumps in (3.65) are, e.g.

$$\llbracket \cdot \rrbracket_{(i,m)jk} := (\cdot)_{ijk} - (\cdot)_{mjk}. \quad (3.68)$$

In the Appendices B and C, flux functions with these properties are presented for the shallow water and Euler equations.

Finally, for each element  $e_\kappa(t)$  the semi-discrete moving mesh DGSEM can be written in the following form:

$$\left\langle \frac{d\mathbf{J}}{dt}, \boldsymbol{\varphi} \right\rangle_N = \left\langle \vec{\mathbb{D}}^N \cdot \vec{\nu}, \boldsymbol{\varphi} \right\rangle_N + \int_{\partial E, N} \varphi (\tilde{\nu}_n^* - \tilde{\nu}_n) dS, \quad \forall \boldsymbol{\varphi} \in \mathbb{P}^N(E), \quad (3.69a)$$

$$\left\langle \frac{d(\mathbf{J}\mathbf{U})}{dt}, \boldsymbol{\varphi} \right\rangle_N = - \left\langle \vec{\mathbb{D}}^N \cdot \overset{\leftrightarrow}{\mathbf{G}}^{\text{EC}}, \boldsymbol{\varphi} \right\rangle_N - \int_{\partial E, N} \boldsymbol{\varphi}^T (\tilde{\mathbf{G}}_n^* - \tilde{\mathbf{G}}_n) dS, \quad \forall \boldsymbol{\varphi} \in \mathbb{P}^N(E, \mathbb{R}^p). \quad (3.69b)$$

We note that the equation (3.69a) is the D-GCL for the moving mesh spectral element DG discretization.

The unit outward facing normal vector and surface element on the element side are constructed from the element metrics by

$$\vec{n} := \frac{1}{\hat{s}} \sum_{l=1}^3 (J\vec{a}^l) \hat{n}^l, \quad \hat{s} := \left| \sum_{l=1}^3 (J\vec{a}^l) \hat{n}^l \right|. \quad (3.70)$$

Thus, the quantity  $\tilde{\nu}_n$  in (3.69a) and the flux  $\tilde{\mathbf{G}}_n$  in (3.69b) are defined by

$$\tilde{\nu}_n = (\hat{s}\vec{n}) \cdot \vec{\nu} = \sum_{l=1}^3 \hat{n}^l (Ja_1^l \nu_1 + Ja_2^l \nu_2 + Ja_3^l \nu_3), \quad (3.71)$$

$$\tilde{\mathbf{G}}_n = (\hat{s}\vec{n}) \cdot \overset{\leftrightarrow}{\mathbf{G}} = \sum_{l=1}^3 \hat{n}^l (Ja_1^l \mathbf{G}_1 + Ja_2^l \mathbf{G}_2 + Ja_3^l \mathbf{G}_3) = \left\{ \mathcal{M} \overset{\leftrightarrow}{\mathbf{G}} \right\} \cdot \hat{n}. \quad (3.72)$$

To define the numerical surface fluxes in (3.69a) and (3.69b), we introduce notation for states at the LGL nodes along an interface between two spatial elements to be a primary “−” and complement the notation with a secondary “+” to denote the value at the LGL nodes on the opposite side. Then the orientated jump and the arithmetic mean at the interfaces are defined by

$$\llbracket \cdot \rrbracket := (\cdot)^+ - (\cdot)^-, \quad \text{and} \quad \{\!\!\{ \cdot \}\!\!\} := \frac{1}{2} \left[ (\cdot)^+ + (\cdot)^- \right]. \quad (3.73)$$

When applied to vectors, the average and jump operators are evaluated separately for each vector component. Then the normal vector  $\vec{n}$  is defined unique to point from the “−” to the “+” side. This notation allows to compute the contravariant surface numerical fluxes in (3.69a) as

$$\tilde{\nu}_n^* = \hat{s} (n_1 \{\!\!\{ v_1 \}\!\!\} + n_2 \{\!\!\{ v_2 \}\!\!\} + n_3 \{\!\!\{ v_3 \}\!\!\}). \quad (3.74)$$

The contravariant surface numerical fluxes in (3.69b) are given by

$$\tilde{\mathbf{G}}_n^* = \hat{s} \left( n_1 \mathbf{G}_1^{\text{EC}} + n_2 \mathbf{G}_2^{\text{EC}} + n_3 \mathbf{G}_3^{\text{EC}} \right), \quad (3.75)$$

where the Cartesian fluxes  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , satisfy (3.63), (3.64), (3.65). We note that these fluxes are the baseline choices without interface dissipation, to get a baseline scheme that is entropy conservative.

**Remark 3.2.** The discrete volume weighted contravariant vectors  $J\vec{a}^\alpha$ ,  $\alpha = 1, 2, 3$ , do not depend on the solution  $J$  of (3.69a), since these vectors are computed by the conservative curl form (3.50). Thus, the discrete metric identities (3.51) are satisfied and the normal computation (3.70) is watertight. This means the normal vector and the surface element are continuous across element interfaces.

### 3.6 Discrete entropy conservation for the semi-discrete method

The spatial integral of the entropy is bounded in time on the continuous level. Thus, it is desirable that a numerical method is stable in the sense that a discrete version of this integral is bounded in time, too. In the context of the moving mesh semi-discrete DGSEM (3.69), we are interested to find an upper bound for the quantity

$$\bar{S} := \sum_{k=1}^K \langle s(\mathbf{U}), J \rangle_N = \sum_{k=1}^K \langle \mathbb{I}^N(s(\mathbf{U})), J \rangle_N = \sum_{k=1}^K \langle S, J \rangle_N, \quad (3.76)$$

where we used the property (3.55) of the inner product (3.54) and the notation (3.67). Next, we prove the following statement for the semi-discrete moving mesh DGSEM.

**Theorem 3.3.** Suppose the flux functions  $\tilde{\mathbf{G}}^{\text{EC}}$  in the derivative projection operator (3.62) and the numerical surface fluxes  $\tilde{\mathbf{G}}_n^*$  are computed by Cartesian fluxes  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , with the properties (3.63), (3.64), (3.65). Then the semi-discrete moving mesh DGSEM (3.69) satisfies the discrete entropy equation

$$\frac{d\bar{S}}{dt} = \sum_{\substack{\text{Boundary} \\ \text{faces}}} \int_{\partial E, N} \left( \tilde{\Psi}_n - \tilde{\nu}_n^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_n^* \right) dS, \quad (3.77)$$

where  $\bar{S}$  is given by (3.76),  $\Phi$  as well as  $\Psi_l$ ,  $l = 1, 2, 3$ , are polynomials with nodal values (3.66) and  $\tilde{\Psi}_n := (\hat{s}\vec{n}) \cdot \vec{\Psi}$  with  $\vec{\Psi} = [\Psi_1, \Psi_2, \Psi_3]^T$ .

*Proof.* We proceed similar as in the continuous entropy analysis, use the polynomial approximation  $\varphi = \mathbb{I}^N(\mathbf{w}(\mathbf{U})) = \mathbf{W}$  as test function in the equation (3.69b) and obtain

$$\left\langle \frac{d(J\mathbf{U})}{dt}, \mathbf{W} \right\rangle_N = - \left\langle \mathbb{D}^N \cdot \tilde{\mathbf{G}}^{\text{EC}}, \mathbf{W} \right\rangle_N - \int_{\partial E, N} \mathbf{W}^T \left( \tilde{\mathbf{G}}_n^* - \tilde{\mathbf{G}}_n \right) dS. \quad (3.78)$$

First, we consider the left hand side in the equation (3.78). We obtain by the chain rule

$$\begin{aligned} \left\langle J \frac{d\mathbf{U}}{dt}, \mathbf{W} \right\rangle_N &= \left\langle \mathbf{w}(\mathbf{U})^T \frac{\partial \mathbf{U}}{\partial t}, J \right\rangle_N + \left\langle \vec{\nu} \cdot \left( \mathbf{w}(\mathbf{U})^T \vec{\nabla}_\xi \mathbf{U} \right), 1 \right\rangle_N \\ &= \left\langle \frac{\partial}{\partial t} s(\mathbf{U}), J \right\rangle_N + \left\langle \vec{\nu} \cdot \vec{\nabla}_\xi s(\mathbf{U}), 1 \right\rangle_N = \left\langle \frac{d}{dt} s(\mathbf{U}), J \right\rangle_N, \end{aligned} \quad (3.79)$$

where we used the identity  $\mathbb{I}^N(\mathbf{w}(\mathbf{U})) = \mathbf{W}$  and the property (3.55) of the inner product (3.54). Since we assume time continuity for our semi-discrete analysis, we apply the product rule in time and obtain by (3.79)

$$\begin{aligned} \left\langle \frac{d(J\mathbf{U})}{dt}, \mathbf{W} \right\rangle_N &= \left\langle \frac{ds(\mathbf{U})}{dt}, J \right\rangle_N + \left\langle \frac{dJ}{dt}, \mathbf{w}(\mathbf{U})^T \mathbf{U} \right\rangle_N \\ &= \frac{d}{dt} \langle s(\mathbf{U}), J \rangle_N + \left\langle \frac{dJ}{dt}, \mathbf{w}(\mathbf{U})^T \mathbf{U} - s(\mathbf{U}) \right\rangle_N. \end{aligned} \quad (3.80)$$

Next, the notations in (3.66), (3.67) and the property (3.55) of the inner product (3.54) give

$$\begin{aligned} \left\langle \frac{d(\mathbf{JU})}{dt}, \mathbf{W} \right\rangle_N &= \frac{d}{dt} \langle s(\mathbf{U}), J \rangle_N + \left\langle \frac{dJ}{dt}, \mathbf{W}(\mathbf{U})^T \mathbf{U} - s(\mathbf{U}) \right\rangle_N = \frac{d}{dt} \langle S, J \rangle_N + \left\langle \frac{dJ}{dt}, \Phi \right\rangle_N \\ &= \frac{d}{dt} \langle S, J \rangle_N + \left\langle \vec{\mathbb{D}}^N \cdot \vec{\nu}, \Phi \right\rangle_N + \int_{\partial E, N} (\tilde{\nu}_n^* - \tilde{\nu}_n) \Phi dS, \end{aligned} \quad (3.81)$$

where we used in the last step the D-GCL (3.69a) with the test function  $\varphi = \Phi$ . We note that the quantity  $\Phi$  is defined as a polynomial with the nodal values (3.66). In the Appendix D.2 the following equation is proven

$$\left\langle \vec{\mathbb{D}}^N \cdot \vec{\mathbf{G}}^{\text{EC}}, \mathbf{W} \right\rangle_N = \int_{\partial E, N} (\tilde{F}_n^s - \tilde{\nu}_n S) dS - \left\langle \vec{\mathbb{D}}^N \cdot \vec{\nu}, \Phi \right\rangle_N, \quad (3.82)$$

where  $\tilde{F}_n^s = (\hat{s}\vec{n}) \cdot \vec{F}^s$  with  $\vec{F}^s = [F_1^s, F_2^s, F_3^s]^T$ . Here the polynomials  $F_l^s, l = 1, 2, 3$ , are given by (3.67). Moreover, we obtain by (3.71) and (3.72)

$$\begin{aligned} & - \mathbf{W}^T (\tilde{\mathbf{G}}_n^* - \tilde{\mathbf{G}}_n) - (\tilde{F}_n^s - \tilde{\nu}_n S) \\ &= \sum_{l=1}^3 \{ \hat{s}n_l (\mathbf{W}^T \mathbf{F}_l - F_l^s) - \hat{s}n_l (\mathbf{W}^T \mathbf{U} - S) \} - \mathbf{W}^T \tilde{\mathbf{G}}_n^* \\ &= \tilde{\Psi}_n - \tilde{\nu}_n \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_n^*. \end{aligned} \quad (3.83)$$

Next, we plug the equations (3.81), (3.82), (3.83) in (3.78) and rearrange. This results in the equation

$$\begin{aligned} \frac{d}{dt} \langle S, J \rangle_N &= - \int_{\partial E, N} \left\{ \mathbf{W}^T (\tilde{\mathbf{G}}_n^* - \tilde{\mathbf{G}}_n) + (\tilde{F}_n^s - \tilde{\nu}_n S) - (\tilde{\nu}_n^* - \tilde{\nu}_n) \Phi \right\} dS \\ &= \int_{\partial E, N} (\tilde{\Psi}_n - \tilde{\nu}_n^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_n^*) dS. \end{aligned} \quad (3.84)$$

Then, we sum the equation (3.84) over all elements and use that the normal computation (3.70) is watertight. This provides the equation

$$\begin{aligned} \frac{d\bar{S}}{dt} &= \sum_{\substack{\text{Boundary} \\ \text{faces}}} \int_{\partial E, N} (\tilde{\Psi}_n - \nu_n^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_n^*) dS \\ &\quad - \sum_{\substack{\text{Interior} \\ \text{faces}}} \int_{\partial E, N} (\llbracket \tilde{\Psi}_n \rrbracket - \llbracket \tilde{\nu}_n \rrbracket \llbracket \Phi \rrbracket - \llbracket \mathbf{W} \rrbracket^T \tilde{\mathbf{G}}_n^*) dS. \end{aligned} \quad (3.85)$$

Since the numerical surface fluxes  $\tilde{\mathbf{G}}_n^*$  are computed by Cartesian fluxes  $\mathbf{G}_l^{\text{EC}}, l = 1, 2, 3$ , with the properties (3.65), it follows

$$\llbracket \tilde{\Psi}_n \rrbracket - \llbracket \tilde{\nu}_n \rrbracket \llbracket \Phi \rrbracket - \llbracket \mathbf{W} \rrbracket^T \tilde{\mathbf{G}}_n^* = \sum_{l=1}^3 \hat{s}n_l (\llbracket \Psi_l \rrbracket - \llbracket \nu_l \rrbracket \llbracket \Phi \rrbracket - \llbracket \mathbf{W} \rrbracket^T \mathbf{G}_l^*) = 0. \quad (3.86)$$

Hence, we obtain the equation

$$\frac{d\bar{S}}{dt} = \sum_{\substack{\text{Boundary} \\ \text{faces}}} \int_{\partial E, N} (\tilde{\Psi}_n - \tilde{\nu}_n^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_n^*) dS. \quad (3.87)$$

This completes the proof of Theorem 3.3.  $\square$

**Remark 3.4.** The boundary contribution in (3.77) becomes zero in the case of periodic boundary conditions. In this case the entropy is conserved.

### 3.7 Discrete entropy stability for the semi-discrete method

Entropy conservation can be merely expected when a reversible process is described by a system of PDEs. In general, conservation laws are describing irreversible processes with discontinuous solutions. Hence, it cannot be expected that the entropy conservative moving mesh DGSEM provides a physical meaningful discretization for the system (3.1). However, the entropy conservative flux can be augmented by an artificial dissipation term.

In the literature, there are different strategies to add dissipation to an entropy conservative flux. Here, dissipation is added via a matrix operator. This approach, for instance, has been used in the context of gas dynamics by Chandrashekar [8] or Winters et al. [62].

The conservative variables  $\mathbf{u}$  can be written in dependence of the entropy variables  $\mathbf{w}$ . Differentiation of the conservative variables  $\mathbf{u} = \mathbf{u}(\mathbf{w})$  provides the symmetric positive definite matrix  $\frac{\partial \mathbf{u}}{\partial \mathbf{w}}$ , since the system (3.1) is assumed to be symmetrizable (cf. e.g. [31]). Thus, it follows by a Taylor expansion up to first order

$$[\![\mathbf{u}]\!] = \left( \frac{\partial \mathbf{u}}{\partial \mathbf{w}} \right) [\![\mathbf{w}]\!] + \mathcal{O}([\![\mathbf{w}]\!]^2), \quad (3.88)$$

where the jump operator is defined by (3.73) at the interfaces. Furthermore, the system (3.1) is hyperbolic. Thus, the flux Jacobian matrices  $\frac{\partial \mathbf{f}_l}{\partial \mathbf{u}}$ ,  $l = 1, 2, 3$ , are diagonalizable and have real eigenvalues  $\{\lambda_i^l(\mathbf{u})\}_{i=1}^p \subseteq \mathbb{R}$ . The corresponding right eigenvector matrices are  $\tilde{\mathcal{R}}_l$ . According to the eigenvector scaling theorem (cf. Barth [1] Theorem 4)], there are symmetric block diagonal scaling matrices  $\mathcal{T}_l$  with

$$\frac{\partial \mathbf{f}_l}{\partial \mathbf{u}} = \tilde{\mathcal{R}}_l \Lambda_l(\mathbf{u}) \tilde{\mathcal{R}}_l^{-1}, \quad \frac{\partial \mathbf{u}}{\partial \mathbf{w}} = \tilde{\mathcal{R}}_l \tilde{\mathcal{R}}_l^T, \quad \tilde{\mathcal{R}}_l = \mathcal{R}_l \mathcal{T}_l, \quad l = 1, 2, 3, \quad (3.89)$$

where  $\Lambda_l(\mathbf{u}) := \text{diag}(\lambda_1^l(\mathbf{u}), \dots, \lambda_p^l(\mathbf{u}))$ . The flux Jacobian matrices  $\frac{\partial \mathbf{g}_l}{\partial \mathbf{u}} = \frac{\partial \mathbf{f}_l}{\partial \mathbf{u}} - \nu_l \mathcal{I}_p$  have the real eigenvalues  $\{\lambda_i^l(\mathbf{u}) - \nu_l\}_{i=1}^p$  and the same right eigenvectors as the flux Jacobian  $\frac{\partial \mathbf{f}_l}{\partial \mathbf{u}}$ . We note that  $\mathcal{I}_p$  is the  $p \times p$  identity matrix. Hence, it follows

$$\frac{\partial \mathbf{g}_l}{\partial \mathbf{u}} = \tilde{\mathcal{R}}_l \Lambda_l(\nu, \mathbf{u}) \tilde{\mathcal{R}}_l^{-1}, \quad \Lambda_l(\nu, \mathbf{u}) := \text{diag}(\lambda_1^l(\mathbf{u}) - \nu_l, \dots, \lambda_p^l(\mathbf{u}) - \nu_l), \quad l = 1, 2, 3. \quad (3.90)$$

Furthermore, we obtain by (3.89)

$$\left( \frac{\partial \mathbf{g}_l}{\partial \mathbf{w}} \right) = \left( \frac{\partial \mathbf{g}_l}{\partial \mathbf{u}} \right) \left( \frac{\partial \mathbf{u}}{\partial \mathbf{w}} \right) = \tilde{\mathcal{R}}_l \Lambda_l(\nu, \mathbf{u}) \tilde{\mathcal{R}}_l^T, \quad l = 1, 2, 3. \quad (3.91)$$

The equation (3.91) motivates the definition of the following matrix dissipation operators

$$\mathcal{H}_l = \hat{\mathcal{R}}_l |\Lambda_l| \hat{\mathcal{R}}_l^T, \quad \hat{\mathcal{R}}_l = \mathcal{R}_l^* \mathcal{T}_l^*, \quad l = 1, 2, 3. \quad (3.92)$$

where the matrices  $\mathcal{R}_l^*$ ,  $\mathcal{T}_l^*$ , depend on the averaged values of the states  $\mathbf{U}^-$ ,  $\mathbf{U}^+$  and they are consistent with the right eigenvector matrix  $\mathcal{R}_l$  and the scaling matrix  $\mathcal{T}_l$ . The matrix  $|\Lambda_l|$  depends on the values  $\{\lambda_i^l(\mathbf{U}^-) - \nu_l^-\}_{i=1}^p$  and  $\{\lambda_i^l(\mathbf{U}^+) - \nu_l^+\}_{i=1}^p$ . The matrix  $\mathcal{H}_l$  needs to be a symmetric positive definite matrix. Therefore, the matrix  $|\Lambda_l|$  has to be chosen carefully. Examples of matrices which lead to a symmetric positive definite dissipation matrix are Ismail-Roe type dissipation [34]

$$|\Lambda_l|^{\text{Roe}} := \text{diag}(|\{\lambda_1^l(\mathbf{U}) - \nu_l\}|, \dots, |\{\lambda_p^l(\mathbf{U}) - \nu_l\}|), \quad l = 1, 2, 3, \quad (3.93)$$

or Rusanov type dissipation

$$|\Lambda_l|^{\text{Rus}} := \max \left( \max_{1 \leq i \leq p} |\lambda_i^l(\mathbf{U}^-) - \nu_l^-|, \max_{1 \leq i \leq p} |\lambda_i^l(\mathbf{U}^+) - \nu_l^+| \right) \mathcal{I}_p, \quad l = 1, 2, 3. \quad (3.94)$$

The Rusanov type dissipation is known to be quite dissipative and leads to smearing of shocks and contact discontinuities. The Roe type dissipation is more accurate in the reproduction of the characteristic profile of shocks or

contact discontinuities, but in the regime of strong shock fronts the Roe type dissipation fails in the stabilization of the numerical method. Therefore, a convex combination of Roe and Rusanov type dissipation is a natural choice [8].

The dissipation operator (3.92) is used to modify the Cartesian numerical surface flux as follows

$$\mathbf{G}_l^{\text{ES}} := \mathbf{G}_l^{\text{EC}} - \frac{1}{2} \mathcal{H}_l [\mathbf{W}], \quad l = 1, 2, 3, \quad (3.95)$$

where the Cartesian fluxes  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , satisfy (3.63), (3.64), (3.65). The contravariant surface numerical fluxes  $\tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}}$  are computed by (3.75).

The numerical fluxes  $\tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}}$  do not generate an entropy conservative scheme, but the result in Theorem 3.3 can be used to prove that the moving mesh DGSEM becomes entropy stable, such that the discrete mathematical entropy is bounded at any time by its initial data, when the numerical fluxes  $\tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}}$  are used.

**Corollary 3.5.** Suppose the flux functions  $\mathbf{G}^{\text{EC}}$  in the derivative projection operator (3.62) are computed by Cartesian fluxes  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , with the properties (3.63), (3.64), (3.65) and the numerical surface fluxes  $\tilde{\mathbf{G}}_{\hat{n}}^* = \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}}$  are computed by the Cartesian fluxes  $\mathbf{G}_l^{\text{ES}}$ ,  $l = 1, 2, 3$ , given by (3.95). Then the semi-discrete moving mesh DGSEM (3.69) satisfies the discrete entropy inequality

$$\frac{d\bar{S}}{dt} \leq \sum_{\substack{\text{Boundary} \\ \text{faces}}} \int_{\partial E, N} \left( \tilde{\Psi}_{\hat{n}} - \tilde{\nu}_{\hat{n}}^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}} \right) dS. \quad (3.96)$$

*Proof.* We proceed as in the proof of Theorem 3.3 and obtain the equation

$$\begin{aligned} \frac{d\bar{S}}{dt} &= \sum_{\substack{\text{Boundary} \\ \text{faces}}} \int_{\partial E, N} \left( \tilde{\Psi}_{\hat{n}} - \tilde{\nu}_{\hat{n}}^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}} \right) dS \\ &\quad - \sum_{\substack{\text{Interior} \\ \text{faces}}} \int_{\partial E, N} \left( [\tilde{\Psi}_{\hat{n}}] - \{\tilde{\nu}_{\hat{n}}\} [\Phi] - [\mathbf{W}]^T \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}} \right) dS. \end{aligned} \quad (3.97)$$

Since the numerical surface fluxes  $\tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}}$  are computed by the Cartesian fluxes (3.95) and the fluxes  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , satisfy (3.65), it follows

$$\begin{aligned} &[\tilde{\Psi}_{\hat{n}}] - [\Phi] \{\tilde{\nu}_{\hat{n}}\} - [\mathbf{W}]^T \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}} \\ &= \sum_{l=1}^3 \hat{s} n_l \left( [\Psi_l] - [\Phi] \{\nu_l\} - [\mathbf{W}]^T \mathbf{G}_l^{\text{EC}} + \frac{1}{2} [\mathbf{W}]^T \mathcal{H}_l [\mathbf{W}] \right) \\ &= \frac{1}{2} \sum_{l=1}^3 \hat{s} n_l [\mathbf{W}]^T \mathcal{H}_l [\mathbf{W}]. \end{aligned} \quad (3.98)$$

Since the matrices  $\mathcal{H}_l$ ,  $l = 1, 2, 3$ , are symmetric positive definite and the outward normal vectors of the curved elements are positive oriented, the equation (3.98) provides

$$- \sum_{\substack{\text{Interior} \\ \text{faces}}} \int_{\partial E, N} \left( [\tilde{\Psi}_{\hat{n}}] - \{\tilde{\nu}_{\hat{n}}\} [\Phi] - [\mathbf{W}]^T \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}} \right) dS \leq 0. \quad (3.99)$$

Hence, we obtain the inequality

$$\frac{d\bar{S}}{dt} \leq \sum_{\substack{\text{Boundary} \\ \text{faces}}} \int_{\partial E, N} \left( \tilde{\Psi}_{\hat{n}} - \tilde{\nu}_{\hat{n}}^* \Phi - \mathbf{W}^T \tilde{\mathbf{G}}_{\hat{n}}^{\text{ES}} \right) dS. \quad (3.100)$$

□

### 3.8 Free stream preservation of the moving mesh DGSEM

In this section, we check the discretization of the geometric and metric terms in time. Since DG methods with the forward Euler discretization are unstable [10, 16], we investigate directly the discretization by an explicit RK method with  $s \geq 2$  stages and the characteristic coefficients  $\{a_{\tau\sigma}\}_{\tau,\sigma=1}^s$ ,  $\{b_\sigma\}_{\sigma=1}^s$ ,  $\{c_\sigma\}_{\sigma=1}^s$ . It is worth to mention that a Courant Lewy Friedrichs (CFL) restriction is necessary when an explicit  $s$ -stage RK method is used in the DG framework. In order to present the RK discretization of the semi-discrete DGSEM (3.69), it is beneficial to write the method in the equivalent nodal representation. This representation is for all  $i, j, k = 0, \dots, N$ , given by

$$\frac{dJ_{ijk}}{dt} = \mathbf{V} \left( (\vec{\nu})_{ijk} \right), \quad (3.101a)$$

$$\frac{d(J_{ijk} \mathbf{U}_{ijk})}{dt} = \mathbf{G} \left( (\vec{\nu})_{ijk}, \mathbf{U}_{ijk} \right), \quad (3.101b)$$

where the right hand sides are given by

$$\mathbf{V} \left( (\vec{\nu})_{ijk} \right) := \vec{\mathbb{D}}^N \cdot \vec{\nu}_{ijk} + \frac{1}{\omega_i \omega_j \omega_k} \int_{\partial E, N} \ell_i \ell_j \ell_k (\tilde{\nu}_n^* - \tilde{\nu}_n) dS, \quad (3.102)$$

$$\mathbf{G} \left( (\vec{\nu})_{ijk}, \mathbf{U}_{ijk} \right) := -\vec{\mathbb{D}}^N \cdot \vec{\mathbf{G}}_{ijk}^{\leftrightarrow \text{EC}} - \frac{1}{\omega_i \omega_j \omega_k} \int_{\partial E, N} \ell_i \ell_j \ell_k \left( \tilde{\mathbf{G}}_n^* - \tilde{\mathbf{G}}_n \right) dS \quad (3.103)$$

with the tensorial Lagrange polynomials  $\ell_i \ell_j \ell_k$  given by (3.45).

Next, as in the Section 2.3 the interval  $[0, T]$  is divided in time levels  $t^n$ . The step size of the time discretization is  $\Delta t$ . The DGSEM solutions, the fluxes and the grid velocity field are approximated in the time levels  $t^n$ , e.g.  $\mathbf{U}(t^n) \approx \mathbf{U}^n$ . Then, the RK discretization of the semi-discrete DGSEM is similar to the RK discretization of the semi-discrete moving mesh FV scheme (2.16). Thus, the moving mesh split form RK-DGSEM is given by

for  $\tau = 1, \dots, s$ :

$$\mathbf{J}_{ijk}^{(\tau)} = \mathbf{J}_{ijk}^n + \Delta t \sum_{\sigma=1}^{\tau-1} a_{\tau\sigma} \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right), \quad (3.104a)$$

$$\mathbf{U}_{ijk}^{(\tau)} = \mathbf{U}_{ijk}^n + \frac{\Delta t}{J_{ijk}^{(\tau)}} \sum_{\sigma=1}^{\tau-1} a_{\tau\sigma} \left( \mathbf{G} \left( (\vec{\nu})_{ijk}^{n+\sigma}, \mathbf{U}_{ijk}^{(\sigma)} \right) - \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right) \mathbf{U}_{ijk}^n \right), \quad (3.104b)$$

$$\mathbf{J}_{ijk}^{n+1} = \mathbf{J}_{ijk}^n + \Delta t \sum_{\sigma=1}^s b_\sigma \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right), \quad (3.104c)$$

$$\mathbf{U}_{ijk}^{n+1} = \mathbf{U}_{ijk}^n + \frac{\Delta t}{J_{ijk}^{(n+1)}} \sum_{\sigma=1}^s b_\sigma \left( \mathbf{G} \left( (\vec{\nu})_{ijk}^{n+\sigma}, \mathbf{U}_{ijk}^{(\sigma)} \right) - \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right) \mathbf{U}_{ijk}^n \right), \quad (3.104d)$$

where  $(\vec{\nu})_{ijk}^{n+\sigma} := \vec{\nu}(\xi_i^1, \xi_j^2, \xi_k^3, t^n + c_\sigma \Delta t)$  and  $\{\xi_i^1\}_{i=0}^N$ ,  $\{\xi_i^2\}_{i=0}^N$ ,  $\{\xi_i^3\}_{i=0}^N$  are sets of LGL points. Next, we prove that the fully-discrete split form RK-DGSEM (3.104) satisfies the free stream preservation property.

**Theorem 3.6.** Suppose the solution of the fully-discrete split form RK-DGSEM (3.104) is given by  $\mathbf{U}_{ijk}^n = \mathbf{C} := (c_1, \dots, c_p)^T \in \mathbb{R}^p$  for all elements  $e_\kappa(t^n)$ ,  $\kappa = 1, \dots, K$ , and the numerical fluxes satisfy (3.63). Then, the constant states  $c_l$ ,  $l = 1, \dots, p$ , are preserved in each Runge-Kutta stage (3.104b). In particular, the solution of the fully-discrete DGSEM method at time level  $t^{n+1}$  is  $\mathbf{U}_{ijk}^{n+1} = \mathbf{C}$ .

*Proof.* Let  $\tau \in \{1, \dots, s\}$  be an arbitrary fixed index. We are interested to investigate the  $\tau$ -th RK stage. Hence, without loss of generality, we can assume that  $\mathbf{U}^{(\sigma)} = \mathbf{C}$  for all  $\sigma = 0, \dots, \tau - 1$ . Then, since the flux  $\vec{\mathbf{G}}^{\leftrightarrow \text{EC}}$  satisfies

(3.63), it follows

$$\begin{aligned} \vec{\mathbb{D}}^N \cdot \vec{\mathbf{G}}_{ijk}^{\leftrightarrow \text{EC}} &= 2 \sum_{m=0}^N \mathcal{D}_{im} \{ \{ J \vec{a}^1 \} \}_{(i,m)jk} \cdot \vec{\mathbf{F}}(\mathbf{C}) \\ &\quad + 2 \sum_{m=0}^N \mathcal{D}_{jm} \{ \{ J \vec{a}^2 \} \}_{i(j,m)k} \cdot \vec{\mathbf{F}}(\mathbf{C}) \\ &\quad + 2 \sum_{m=0}^N \mathcal{D}_{km} \{ \{ J \vec{a}^3 \} \}_{ij(k,m)} \cdot \vec{\mathbf{F}}(\mathbf{C}) - \vec{\mathbb{D}}^N \cdot \left( \vec{\nu} \right)_{ijk}^{n+\sigma} \mathbf{C}. \end{aligned} \quad (3.105)$$

Furthermore, since the metric terms are computed by the conservative curl form (3.50), we obtain

$$\begin{aligned} &2 \sum_{m=0}^N \left( \mathcal{D}_{im} \{ \{ J \vec{a}^1 \} \}_{(i,m)jk} + \mathcal{D}_{jm} \{ \{ J \vec{a}^2 \} \}_{i(j,m)k} + \mathcal{D}_{km} \{ \{ J \vec{a}^3 \} \}_{ij(k,m)} \right) \\ &= \sum_{m=0}^N \left( \mathcal{D}_{im} (J \vec{a}^1)_{mjk} + \mathcal{D}_{jm} (J \vec{a}^2)_{imk} + \mathcal{D}_{km} (J \vec{a}^3)_{ijm} \right) = 0. \end{aligned} \quad (3.106)$$

Here we used the split form Lemma from Gassner et al. [26 Lemma 1] in the first step and in the second step we used the identity (3.52) for the discrete metric identities. Thus, it follows that

$$\vec{\mathbb{D}}^N \cdot \vec{\mathbf{G}}_{ijk}^{\leftrightarrow \text{EC}} = -\vec{\mathbb{D}}^N \cdot \left( \vec{\nu} \right)_{ijk}^{n+\sigma} \mathbf{C}. \quad (3.107)$$

Similar, since the flux  $\vec{\mathbf{G}}^*$  satisfies (3.63), it follows that

$$\begin{aligned} \vec{\mathbf{G}}_{\hat{n}}^* - \vec{\mathbf{G}}_{\hat{n}} &= \sum_{l=1}^3 \hat{s} n_l \left( \mathbf{F}_l(\mathbf{C}) - \{ \{ \nu_l^{n+\sigma} \} \} \mathbf{C} \right) - (\hat{s} \vec{n}) \cdot \left( \vec{\mathbf{F}}(\mathbf{C}) - (\vec{\nu})^{n+\sigma} \mathbf{C} \right) \\ &= -(\hat{s} \vec{n}) \cdot \left( \{ \{ (\vec{\nu})^{n+\sigma} \} \} - (\vec{\nu})^{n+\sigma} \right) \mathbf{C} = -(\tilde{\nu}_{\hat{n}}^{*,n+\sigma} - \tilde{\nu}_{\hat{n}}^{n+\sigma}) \mathbf{C}. \end{aligned} \quad (3.108)$$

Thus, the equations (3.107) and (3.108) give

$$\mathbf{G} \left( (\vec{\nu})_{ijk}^{n+\sigma}, \mathbf{C} \right) = \left( \vec{\mathbb{D}}^N \cdot \left( \vec{\nu} \right)_{ijk}^{n+\sigma} + \frac{1}{\omega_i \omega_j \omega_k} \int_{\partial E, N} \ell_i \ell_j \ell_k \left( \tilde{\nu}_{\hat{n}}^{*,n+\sigma} - \tilde{\nu}_{\hat{n}}^{n+\sigma} \right) dS \right) \mathbf{C} = \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right) \mathbf{C}. \quad (3.109)$$

Hence, the solution of the RK stage (3.104b) is given by

$$\mathbf{U}_{ijk}^{(\tau)} = \mathbf{C} + \frac{\Delta t}{J_{ijk}^{(\tau)}} \sum_{\sigma=1}^{\tau-1} a_{\tau\sigma} \left( \mathbf{G} \left( (\vec{\nu})_{ijk}^{n+\sigma}, \mathbf{C} \right) - \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right) \mathbf{C} \right) = \mathbf{C}. \quad (3.110)$$

Since the parameter  $\tau$  was arbitrary chosen, it follows  $\mathbf{U}_{ijk}^{(\tau)}$  for all  $\tau = 1, \dots, s$ . In particular, it follows

$$\mathbf{U}_{ijk}^{n+1} = \mathbf{C} + \frac{\Delta t}{J_{ijk}^{n+1}} \sum_{\sigma=1}^s b_{\sigma} \left( \mathbf{G} \left( (\vec{\nu})_{ijk}^{n+\sigma}, \mathbf{C} \right) - \mathbf{V} \left( (\vec{\nu})_{ijk}^{n+\sigma} \right) \mathbf{C} \right) = \mathbf{C}. \quad (3.111)$$

This completes the proof of Theorem 3.6.  $\square$

## 4 Numerical results

The numerical computations in this section are performed with the open source code FLEXI<sup>1</sup> and the three-dimensional high-order meshes for the simulations are generated with the open source tool HOPR<sup>2</sup>

<sup>1</sup>[www.flexi-project.org](http://www.flexi-project.org)

<sup>2</sup>[www.hopr-project.org](http://www.hopr-project.org)

We present tests on three dimensional moving hexahedral curved meshes for the compressible Euler equations. Based on these tests we will evaluate the theoretical findings of the previous sections. The three dimensional compressible Euler equations are given by

$$\frac{\partial \mathbf{u}}{\partial t} + \vec{\nabla} \cdot \overset{\leftrightarrow}{\mathbf{f}} = \mathbf{0}. \quad (4.1)$$

The state vector and the components of the block vector flux,  $\overset{\leftrightarrow}{\mathbf{f}}$ , are given by

$$\mathbf{u} = \begin{bmatrix} \rho \\ \rho u_1 \\ \rho u_2 \\ \rho u_3 \\ E \end{bmatrix}, \quad \mathbf{f}_1 = \begin{bmatrix} \rho u_1 \\ \rho u_1^2 + p \\ \rho u_1 u_2 \\ \rho u_1 u_3 \\ (E + p) u_1 \end{bmatrix}, \quad \mathbf{f}_2 = \begin{bmatrix} \rho u_2 \\ \rho u_1 u_2 \\ \rho u_2^2 + p \\ \rho u_2 u_3 \\ (E + p) u_2 \end{bmatrix}, \quad \mathbf{f}_3 = \begin{bmatrix} \rho u_3 \\ \rho u_1 u_3 \\ \rho u_2 u_3 \\ \rho u_3^2 + p \\ (E + p) u_3 \end{bmatrix}, \quad (4.2)$$

where the conserved variables are the density  $\rho$ , the momentum  $\rho \vec{u} = [\rho u_1, \rho u_2, \rho u_3]^T$  and the total energy  $E$ . In order to close the system, we assume an ideal gas such that the pressure is defined as

$$p = (\gamma - 1) \left( E - \frac{\rho}{2} |\vec{u}|^2 \right), \quad (4.3)$$

where  $\gamma$  is the adiabatic exponent. We choose  $\gamma = 1.4$  in the following experiments. The system (4.1) is investigated in the domain  $\Omega = [x_{\min}, x_{\max}]^3$ . At initial time  $t = 0$  the domain is divided in  $K$  non-overlapping and conforming cartesian hexahedral elements  $e_\kappa(0)$ ,  $\kappa = 1, \dots, K$ . For each element  $e_\kappa(0)$ ,  $\kappa = 1, \dots, K$ , the temporal distribution of a grid point

$$\vec{x}_\kappa(0) = (x_1^\kappa(0), x_2^\kappa(0), x_3^\kappa(0))^T \in e_\kappa(0) \quad (4.4)$$

is given by

$$\vec{x}_\kappa(t) = \vec{x}_\kappa(0) + 0.05L \sin(2\pi t) \sin\left(\frac{2\pi}{L} x_1^\kappa(0)\right) \sin\left(\frac{2\pi}{L} x_2^\kappa(0)\right) \sin\left(\frac{2\pi}{L} x_3^\kappa(0)\right), \quad (4.5)$$

where  $L := x_{\max} - x_{\min}$ . In Figure 4.2, we show a slice through a three dimensional mesh with  $K = 16^3$  elements at initial time and at its maximal distortion.

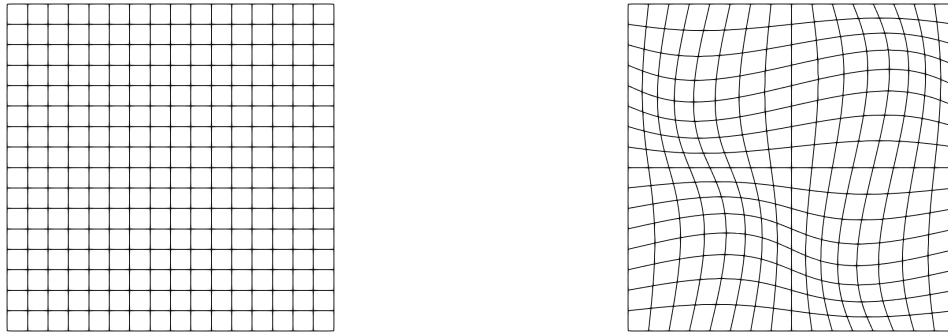


Figure 4.2: A slice through a three dimensional mesh with  $K = 16^3$  elements at initial time (left) and at its maximal distortion (right).

Furthermore, the five stage fourth order low-storage explicit RK method from Kennedy, Carpenter and Lewis [35] is used for the time-integration in the numerical experiments. The CFL restriction is computed as in [13]

$$\frac{\Delta t}{\min_{1 \leq \kappa \leq K} |h_\kappa(t^n)|} \leq \frac{C_{\text{CFL}}}{(2N + 1) \lambda_{\max}}, \quad (4.6)$$



where  $h_\kappa(t^n)$  is the minimum element size of  $e_\kappa(t^n)$ ,  $C_{\text{CFL}} \in (0, 1]$  and  $\lambda_{\text{max}}$  is the largest advective wave speed at the current time level traveling in either the  $x_1, x_2, x_3$ -direction.

#### 4.1 Experimental convergence rates

In this section, we verify the high-order approximation of the moving DGSEM (3.69). For this purpose, we investigate the domain  $\Omega = [-1, 1]^3$  and apply the method of manufactured solutions. Thus, we assume a solution of the form

$$\mathbf{U}(\vec{x}, t) = \begin{bmatrix} \rho(\vec{x}, t) \\ \rho u_1(\vec{x}, t) \\ \rho u_2(\vec{x}, t) \\ \rho u_3(\vec{x}, t) \\ E(\vec{x}, t) \end{bmatrix} = \begin{bmatrix} 2 + 0.1 \sin(\pi(x_1 + x_2 + x_3 - 2 \cdot 0.3t)) \\ 2 + 0.1 \sin(\pi(x_1 + x_2 + x_3 - 2 \cdot 0.3t)) \\ 2 + 0.1 \sin(\pi(x_1 + x_2 + x_3 - 2 \cdot 0.3t)) \\ 2 + 0.1 \sin(\pi(x_1 + x_2 + x_3 - 2 \cdot 0.3t)) \\ [2 + 0.1 \sin(\pi(x_1 + x_2 + x_3 - 2 \cdot 0.3t))]^2 \end{bmatrix}. \quad (4.7)$$

We plug solution (4.7) into the Euler system and compute the residual using a computer algebra system. This term is used as a source term in our convergence tests. We note that this term is handled and discretized as a solution independent part in the numerical computation.

We run the convergence test with periodic boundary conditions. Furthermore, the moving mesh DGSEM (3.69) is applied with the flux function in Appendix C.1 as volume and surface flux. In addition, the surface flux is stabilized by the dissipation operator in Appendix C.3. Besides using the grid point distribution given in (4.5), we also compute static reference solutions, by setting the grid velocity to zero. In this case, the moving mesh DGSEM (3.69) degenerates to the split form DGSEM for static meshes [26, 27].

In Table 4.1, we listed the experimental order of convergence (EOC) and  $L^2$  errors for the conserved variables that we obtain for polynomials with odd degree  $N = 3$  on a static mesh (top) and on a moving mesh (bottom). The convergence rates on the moving mesh are not as good as on a static mesh, which can be justified by the high distortion in the mesh from the grid point distribution formula (4.5). However, with an increasing number of elements the same convergence rates as on a static mesh are almost reached. Moreover, the experimental order of convergence (EOC) and  $L^2$  errors for the conserved variables that we obtain for polynomials with even degree  $N = 4$  are listed in the Table 4.2. We observe a similar behavior as for the odd degree  $N = 3$ . This indicates the high-order approximation properties of the moving mesh DGSEM.

$K$	$L^2(\rho)$	$EOC(\rho)$	$L^2(\rho u_1)$	$EOC(\rho u_1)$	$L^2(\rho u_2)$	$EOC(\rho u_2)$	$L^2(\rho u_3)$	$EOC(\rho u_3)$	$L^2(E)$	$EOC(E)$
$2^3$	2.84E-02	-	2.74E-02	-	2.74E-02	-	2.74E-02	-	5.47E-02	-
$4^3$	5.54E-03	2.36	5.43E-03	2.34	5.43E-03	2.34	5.43E-03	2.34	1.03E-03	2.40
$8^3$	4.35E-05	6.99	4.28E-05	6.99	4.28E-05	6.99	4.28E-05	6.99	1.06E-04	6.61
$16^3$	2.10E-06	4.37	2.07E-06	4.37	2.08E-06	4.37	2.07E-06	4.37	5.33E-06	4.31
$32^3$	1.26E-07	4.06	1.24E-07	4.06	1.24E-07	4.06	1.24E-07	4.06	3.19E-07	4.06
$64^3$	7.82E-09	4.01	7.67E-09	4.01	7.67E-09	4.01	7.67E-09	4.01	1.97E-08	4.01
$2^3$	4.16E-02	-	3.73E-02	-	3.73E-02	-	3.73E-02	-	5.61E-02	-
$4^3$	3.77E-03	3.46	3.52E-03	3.41	3.52E-03	3.41	3.52E-03	3.41	6.06E-03	3.21
$8^3$	1.99E-04	4.25	1.75E-04	4.33	1.75E-04	4.33	1.75E-04	4.33	3.24E-04	4.23
$16^3$	5.37E-06	5.21	4.91E-06	5.16	4.91E-06	5.16	4.91E-06	5.16	1.20E-05	4.75
$32^3$	2.18E-07	4.62	2.07E-07	4.57	2.07E-07	4.57	2.07E-07	4.57	5.83E-07	4.36
$64^3$	1.45E-08	3.92	1.34E-08	3.95	1.34E-08	3.95	1.34E-08	3.95	3.95E-08	3.88

Table 4.1: Experimental order of convergence (EOC) and  $L^2$  errors at time  $T = 5$  for the Euler manufactured solution test (4.7). The moving mesh DGSEM is used with  $N = 3$  on a static mesh (top) and on a moving mesh (bottom) with the grid point distribution (4.5).

$K$	$L^2(\rho)$	$EOC(\rho)$	$L^2(\rho u_1)$	$EOC(\rho u_1)$	$L^2(\rho u_2)$	$EOC(\rho u_2)$	$L^2(\rho u_3)$	$EOC(\rho u_3)$	$L^2(E)$	$EOC(E)$
$2^3$	6.99E-03	-	6.64E-03	-	6.64E-03	-	6.64E-03	-	1.16E-02	-
$4^3$	4.02E-04	4.12	3.97E-04	4.06	3.97E-04	4.06	3.97E-04	4.06	7.96E-04	3.87
$8^3$	4.50E-06	6.48	4.50E-06	6.47	4.50E-06	6.47	4.50E-06	6.47	1.16E-05	6.10
$16^3$	1.37E-07	5.04	1.38E-07	5.02	1.38E-07	5.02	1.38E-07	5.02	3.66E-07	4.98
$32^3$	4.33E-09	4.98	4.40E-09	4.97	4.40E-09	4.97	4.40E-09	4.97	1.16E-08	4.97
$64^3$	1.36E-10	4.99	1.38E-10	4.99	1.38E-10	4.99	1.38E-10	4.99	3.66E-10	4.99
$2^3$	1.02E-02	-	9.06E-03	-	9.06E-03	-	9.06E-03	-	1.45E-02	-
$4^3$	4.53E-04	4.50	4.13E-04	4.46	4.13E-04	4.46	4.13E-04	4.46	7.18E-04	4.33
$8^3$	1.10E-05	5.37	1.02E-05	5.35	1.02E-05	5.35	1.02E-05	5.35	1.86E-05	5.27
$16^3$	1.91E-07	5.85	1.72E-07	5.88	1.72E-07	5.88	1.72E-07	5.88	3.81E-07	5.61
$32^3$	7.28E-09	4.71	6.33E-09	4.77	6.33E-09	4.77	6.33E-09	4.77	1.38E-08	4.78
$64^3$	2.79E-10	4.71	2.38E-10	4.74	2.38E-10	4.74	2.38E-10	4.74	5.40E-10	4.68

Table 4.2: Experimental order of convergence (EOC) and  $L^2$  errors at time  $T = 5$  for the Euler manufactured solution test (4.7). The moving mesh DGSEM is used with  $N = 4$  on a static mesh (top) and on a moving mesh (bottom) with the grid point distribution (4.5).

## 4.2 Entropy analysis validation

The three dimensional Euler equations (4.1) are equipped with the entropy/entropy flux pairs

$$s = -\frac{\rho\varsigma}{\gamma-1}, \quad f_l^s = -\frac{\rho\varsigma u_l}{\gamma-1}, \quad l = 1, 2, 3, \quad (4.8)$$

where  $\varsigma = \log(p\rho^{-\gamma})$ . We are interested in the behavior of the discrete entropy conservation error

$$\Delta_S(T) = \bar{S}(T) - \bar{S}(0), \quad (4.9)$$

where  $\bar{S}(\cdot)$  is computed by (3.76). We investigate the inviscid Taylor-Green vortex (TGV) test case [53] in the domain  $\Omega = [0, 2\pi]^3$ . The inviscid TGV can be a challenging test case regarding the robustness of a numerical scheme, partly because the dynamics produce arbitrarily small scales. The flow field is thus by design under-resolved, which makes it a suitable test case to investigate the entropy conservation properties of the scheme. The TGV evolves from the initial data

$$\begin{aligned} \rho &= 1, \\ \vec{u} &= [\sin(x_1) \cos(x_2) \cos(x_3), -\cos(x_1) \sin(x_2) \cos(x_3), 0]^T, \\ p &= p_0 + \frac{1}{16} (\cos(2x_1) + \cos(2x_2)) (\cos(2x_3) + 2). \end{aligned} \quad (4.10)$$

To render the simulation close to incompressible, the Mach number  $M_0 = \frac{1}{\sqrt{\gamma p_0}}$  is set to 0.1 by adjusting the pressure correspondingly. We run the simulation with  $K = 16^3$  elements and periodic boundary conditions. The final time is chosen to be  $T = 13$ . Furthermore, we apply the flux function in Appendix C.1 to compute the derivative projection operator (3.62). In Figure 4.3 we present a log-log plot of the entropy conservation error for  $N = 3, 4$ . We note that the flux in Appendix C.1 was used as surface flux without a dissipation term in these computations, rendering the semi-discrete discretization fully entropy conserving. As expected, we observe the reduction of the remaining entropy conservation error according to the order of the RK method for decreasing CFL numbers. In Figure 4.4 the temporal evolution of the entropy conservation errors  $\Delta_S(T)$  is given. The CFL number is set to  $C_{\text{CFL}} = 0.125$  and polynomial degrees  $N = 3$  and  $N = 4$  are used. We observe that the entropy conservation error  $\Delta_S(T)$  is constant in time (dashed line) when the flux in Appendix C.1 is used without a dissipation term. This indicates the entropy conservation in the TVG test case. On the other hand the entropy conservation error  $\Delta_S(T)$  is decreasing in time (solid line) when the surface flux is stabilized by the dissipation term in Appendix C.3. Thus, the moving

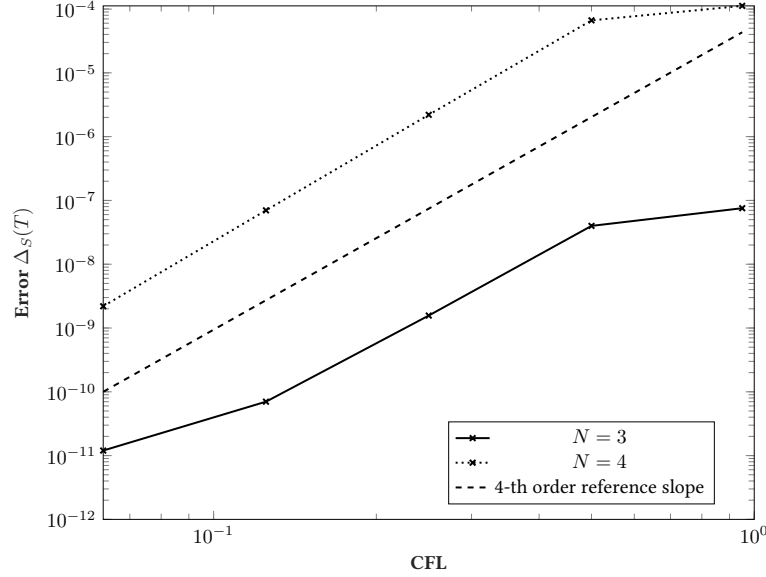


Figure 4.3: Log-log plot of the entropy conservation errors  $\Delta_S(T)$  for the Euler equations with initial data (4.10). The errors are given at time  $T = 13$  for polynomials with degree  $N = 3$  (solid line) and  $N = 4$  (dotted line) on a curved moving mesh with  $K = 16^3$  elements.

mesh DGSEM is an entropy stable scheme in this test case. These observations agree with the results in Theorem 3.3 and Corollary 3.5.

### 4.3 Free stream preservation validation

We consider the Euler equations (4.1) on the domain  $\Omega = [0, 2\pi]^3$  with the initial data

$$\mathbf{U}(x, t) = \begin{bmatrix} \rho(x, t) \\ \rho u_1(x, t) \\ \rho u_2(x, t) \\ \rho u_3(x, t) \\ E(x, t) \end{bmatrix} = \begin{bmatrix} 1 \\ 0.3 \\ 0 \\ 0 \\ 17 \end{bmatrix}. \quad (4.11)$$

The entropy stable DGSEM is applied with the flux function in Appendix C.1 as volume and surface flux as well as the dissipation operator in Appendix C.3 to stabilize the surface flux. We apply  $K = 16^3$  elements, the formula (4.5) to describe the displacement of the mesh points and periodic boundary conditions are used in the simulation. Furthermore, the final time is set to  $T = 20$ . In Table 4.11, we present the  $L^\infty$  errors between the initial data (4.11) and the numerical solution at time  $T = 20$  for polynomials of degree  $N = 3$  (top),  $N = 4$  (bottom) and different CFL numbers  $C_{\text{CFL}}$ . We observe that the errors are close to zero and vary slightly for the different CFL numbers. These results indicate the compliance of the free stream preservation property.

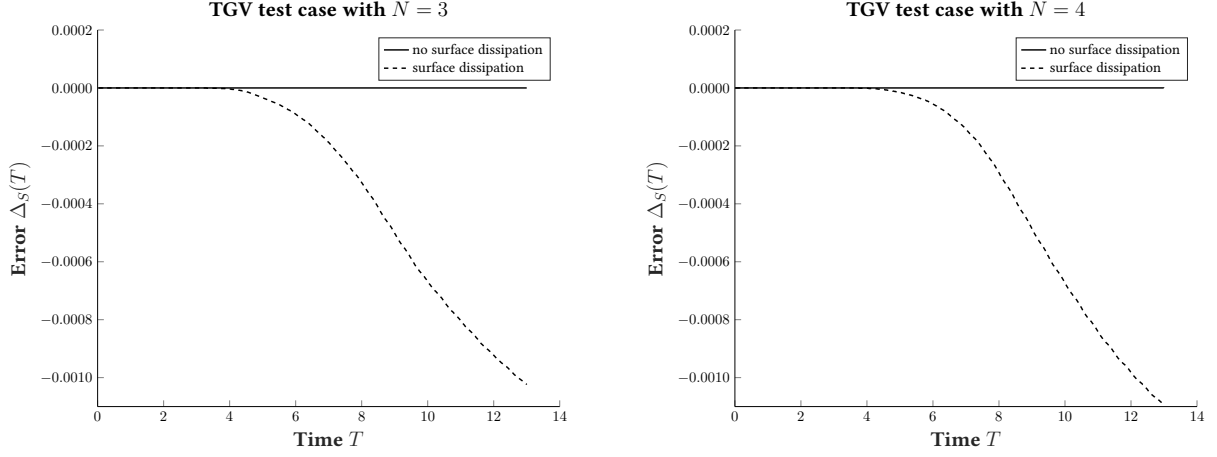


Figure 4.4: Temporal evolution of the entropy conservation errors  $\Delta_S(T)$  for the Euler equations with initial data (4.10). The flux in Appendix C.1 is used as surface flux without dissipation (solid line) and with the dissipation term in Appendix C.3 (dashed line).

$C_{\text{CFL}}$	$L^\infty(\rho)$	$L^\infty(\rho u_1)$	$L^\infty(\rho u_2)$	$L^\infty(\rho u_3)$	$L^\infty(E)$
0.95	2.47E-14	1.40E-12	4.46E-12	4.48E-12	1.33E-12
0.5	2.47E-14	1.40E-12	4.46E-12	4.48E-12	1.33E-12
0.25	2.70E-14	1.40E-12	4.46E-12	4.48E-12	1.36E-12
0.125	3.10E-14	1.40E-12	4.46E-12	4.48E-12	1.43E-12
0.0625	3.78E-14	1.40E-12	4.46E-12	4.48E-12	1.56E-12
0.95	2.07E-14	1.24E-12	5.28E-12	5.21E-12	1.12E-12
0.5	2.49E-14	1.24E-12	5.28E-12	5.21E-12	1.30E-12
0.25	2.81E-14	1.24E-12	5.28E-12	5.21E-12	1.34E-12
0.125	3.32E-14	1.24E-12	5.28E-12	5.21E-12	1.40E-12
0.0625	4.24E-14	1.24E-12	5.28E-12	5.21E-12	1.59E-12

Table 4.3: Free stream preservation test for  $N = 3$  (top) and  $N = 4$  (bottom). The  $L^\infty$  errors measure the difference between the initial data (4.11) and the numerical solution at time  $T = 20$  for different constants  $C_{\text{CFL}}$ .

#### 4.4 Robustness Test

As has been stated in Section 4.2 and noted in literature [49, 61], the inviscid TGV is a notoriously challenging test case for the stability of a high order discretization. While for lower polynomial degrees calculations may be possible, high-order simulations are known to crash even if aliasing-reducing methods like polynomial dealiasing are used [49]. Thus, we use the TGV test case (4.10) to demonstrate the increased robustness of the entropy stable moving mesh DGSEM. To do so, we run the simulation up to  $T = 13$  using a polynomial degree of  $N = 7$  on three different meshes employing  $K_1 = 14^3$ ,  $K_2 = 19^3$  and  $K_3 = 26^3$  elements. These cases correspond to the most restrictive simulations from [49]. Again, the point distribution given in (4.5) is used. We use the flux function in Appendix C.1 as volume and surface flux and stabilize the surface flux by the dissipation operator in Appendix C.3

Using the entropy stable moving DGSEM, we were able to run all simulations until final time. This shows that the consistent dissipation operators in combination with the entropy conservative volume fluxes can lead to superior

stability properties.

## 5 Conclusions

In this work a moving mesh DGSEM to solve non-linear conservation laws has been constructed and analyzed. The semi-discrete method is provably entropy stable and the free stream preservation property is satisfied for each explicit  $s$ -stage Runge-Kutta method.

We started the paper with a view on the ALE transformation for one dimensional conservation laws and constructed a moving mesh FV method. The condition (2.19) to construct entropy conservative two-point fluxes in the ALE framework has been presented. This condition can be seen as an extension of Tadmor's discrete entropy condition [56] in the ALE framework. It is remarkable that an entropy conservative moving mesh flux can be constructed directly from an entropy conservative static mesh flux, when a numerical state function with the property (2.23) is available. This approach has been used in the appendix to construct entropy conservative fluxes for the shallow water and Euler equations.

Then, the moving mesh DGSEM has been presented for three dimensional conservation laws. The derivatives in space are approximated with high-order derivative matrices which are SBP operators. Furthermore, the split form DG framework [26, 27] has been used to avoid aliasing in the discretization of the volume integrals. In addition, two-point flux functions with the generalized entropy condition (2.19) are used in the split form DG framework. These modules in the spatial discretization are the basis to prove that the moving mesh DGSEM is an entropy stable scheme. It is worth to mention that the discrete entropy analysis requires merely the assumption that the time derivatives can be evaluated exactly. Operations like the integration-by-parts formula are mimicked by the SPB operators on the discrete level.

The three dimensional Euler equations have been considered to verify the proven properties of the moving mesh DGSEM in our numerical experiments. We presented convergence tests for smooth test problems to verify that the split form DG framework provides also on a moving mesh a high-order accurate approximation. Furthermore, the numerical robustness tests in the Section 4.4 emphasize the relevance of the entropy stable DGSEM, since the method was able to run the challenging inviscid TGV test case until final time.

## Acknowledgement

Gero Schnücke and Gregor Gassner are supported by the European Research Council (ERC) under the European Union's Eight Framework Program Horizon 2020 with the research project Extreme, ERC grant agreement no. 714487. The authors gratefully acknowledge the support and the computing time on "Hazel Hen" provided by the HLRS through the project "hpcdg".

## References

- [1] T. J. Barth. Numerical methods for gasdynamic systems on unstructured meshes. In: An introduction to recent developments in theory and numerics for conservation laws. Springer, Berlin, Heidelberg, 1999. S. 195-285.
- [2] R. Bellman. Introduction to matrix analysis, volume 19 of Classics in Applied Mathematics. SIAM, Philadelphia, PA, 2nd Edition, 1987.

- [3] M. Böhm, A. R. Winters, G. J. Gassner, D. Derigs, F. Hindenlang and J. Saur. An entropy stable nodal discontinuous Galerkin method for the resistive MHD equations. Part I: Theory and Numerical Verification. *J. Comput. Phys.* (2018), [doi.org/10.1016/j.jcp.2018.06.027](https://doi.org/10.1016/j.jcp.2018.06.027).
- [4] W. Boscheri and M. Dumbser. Arbitrary-Lagrangian-Eulerian discontinuous Galerkin schemes with a posteriori subcell finite volume limiting on moving unstructured meshes. *J. Comput. Phys.* 346 (2017), 449-479.
- [5] J. C. Butcher. *The Numerical Analysis of Ordinary Differential Equations. Runge-Kutta and General Linear Methods*. Wiley, Chichester, 1987.
- [6] J. Chan. On discretely entropy conservative and entropy stable discontinuous Galerkin methods. *J. Comput. Phys.* 362 (2018), 346-374.
- [7] C. Canuto, M. Y. Hussaini, A. Quarteroni and T. A. Zang. *Spectral methods: Fundamentals in single domains*. Springer Verlag, 2006.
- [8] P. Chandrashekar. Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations. *Commun. Comput. Phys.* 14 (2013), 1252-1286.
- [9] P. Chandrashekar and M. Zenk. Well-balanced nodal discontinuous Galerkin method for Euler equations with gravity. *J. Sci. Comput.* 71 (2017), 1062-1093.
- [10] G. Chavent and B. Cockburn. The Local Projection  $P^0 P^1$ -Discontinuous Galerkin Finite Element Method for Scalar Conservation Laws. *ESAIM-Math. Model. Num. ( $M^2AN$ )* 23 (1989), 565-592.
- [11] T. Chen and C.-W. Shu. Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws. *J. Comput. Phys.* 345 (2017), 427-461.
- [12] E. Chiodaroli, C. De Lellis and O. Kreml. Global Ill-Posedness of the Isentropic System of Gas Dynamics. *Commun. Pur. Appl. Math.* 68 (2015), 1157-1190.
- [13] B. Cockburn and C.-W. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *J. Sci. Comput.* 16 (2001), 173-261.
- [14] J. Crean, J. E. Hicken, D. C. Del Rey Fernández, D. W. Zingg and M. H. Carpenter. Entropy-stable summation-by-parts discretization of the Euler equations on general curved elements. *J. Comput. Phys.* 356 (2018), 410-438.
- [15] R. J. Di Perna. Convergence of approximate solutions to conservation laws. *Arch. Ration. Mech. An.* 82 (1983), 27-70.
- [16] D. A. Di Pietro and A. Ern. *Mathematical aspects of discontinuous Galerkin methods*. Vol. 69. Springer Science & Business Media, 2011.
- [17] J. Donea, A. Huerta, J. P. Ponthot and A. Rodríguez-Ferran. In *Encyclopedia of Computational Mechanics*, E. Stein, R. De Borst and T. J.R. Hughes (Eds.), Volume. 1: Fundamentals., Chapter 14: Arbitrary Lagrangian-Eulerian Methods, Wiley, 2004.
- [18] C. Farhat, P. Geuzaine and C. Grandmont. The Discrete Geometric Conservation Law and the Nonlinear Stability of ALE Schemes for the Solution of Flow Problems on Moving Grids. *J. Comput. Phys.* 174 (2001), 669-694.
- [19] R. Fazio and R. J. LeVeque. Moving-mesh methods for one-dimensional hyperbolic problems using CLAWPACK. *Comput. Math. Appl.* 45 (2003), 273-298.

- [20] D. C. Del Rey Fernández, J. E. Hicken, and D. W. Zingg. Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations. *Comput. Fluids* (2014), 171-196.
- [21] T. C. Fisher and M. H. Carpenter. high order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains. *J. Comput. Phys.* 252 (2013), 518-557.
- [22] U. S. Fjordholm, S. Mishra and E. Tadmor. Well-balanced and energy stable schemes for the shallow water equations with discontinuous topography. *J. Comput. Phys.* 230 (2011), 5587-5609.
- [23] L. Friedrich, G. Schnücke, A. R. Winters, D. C. Del Rey Fernández, G. J. Gassner and M. H. Carpenter. Entropy Stable Space-Time Discontinuous Galerkin Schemes with Summation-by-Parts Property for Hyperbolic Conservation Laws. (2018) arXiv preprint [arXiv:1808.08218](https://arxiv.org/abs/1808.08218).
- [24] P. Fu, G. Schnücke, and Y. Xia. Arbitrary Lagrangian-Eulerian discontinuous Galerkin method for conservation laws on moving simplex meshes. *Math. Comput.* (2018). Available via <https://doi.org/10.1090/mcom/3417> (Cited December 24, 2018)
- [25] G. J. Gassner. A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods. *SIAM J. Sci. Comput.* 35 (2013), A1233-A1253.
- [26] G. J. Gassner, A. R. Winters and D. A. Kopriva. Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations. *J. Comput. Phys.* 327 (2016), 39-66.
- [27] G. J. Gassner, A. R. Winters, F. J. Hindenlang and D. A. Kopriva. The BR1 scheme is stable for the compressible Navier-Stokes equations. *J. Sci. Comput.* 77 (2018), 1-47.
- [28] G. J. Gassner, A. R. Winters and D. A. Kopriva. A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations. *Appl. Math. Comput.* 272 (2016), 291-308.
- [29] S. K. Godunov. An interesting class of quasilinear systems. *Dokl. Acad. Nauk SSSR*. Vol. 139 (1961), 521-523.
- [30] H. Guillard and C. Farhat. On the significance of the geometric conservation law for flow computations on moving meshes, *Comput. Method. Appl. M.* 190 (2000), 1467-1482.
- [31] A. Harten. On the symmetric form of systems of conservation laws with entropy. *J. Comput. Phys.* 49 (1983), 151-164.
- [32] F. J. Hindenlang, T. Bolemann and C.-D. Munz. Mesh curving techniques for high order discontinuous Galerkin simulations. In: *IDIHOM: Industrialization of high-order methods-a top-down approach*. Springer, Cham, 2015, 133-152.
- [33] W. Huang and R. D. Russell. Adaptive moving mesh methods. Vol. 174. Springer Science & Business Media, 2010.
- [34] F. Ismail and P. L. Roe. Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks. *J. Comput. Phys.* 228 (2009), 5410-5436.
- [35] C. A. Kennedy, M. H. Carpenter and R. M. Lewis. Low-storage, explicit Runge-Kutta schemes for the compressible Navier-Stokes equations. *Appl. Numer. Math.* 35 (2000), 177-219.
- [36] D. A. Kopriva. Metric identities and the discontinuous spectral element method on curvilinear meshes. *J. Sci. Comput.* 26 (2006), 301-327.

- [37] D. A. Kopriva. Implementing spectral methods for partial differential equations: Algorithms for scientists and engineers. Springer Science & Business Media, 2009.
- [38] D. A. Kopriva, A. R. Winters, M. Böhm and G. J. Gassner. A provably stable discontinuous Galerkin spectral element approximation for moving hexahedral meshes. *Comput. Fluids* 139 (2016), 148-160.
- [39] H.-O. Kreiss and J. Oliger. Comparison of accurate methods for the integration of hyperbolic equations. *Tellus* 24 (1972), 199-215.
- [40] S. N. Kružkov. First order quasilinear equations in several independent variables. *Math. USSR SB+* 10 (1970), 217-243.
- [41] M. Lesoinne and C. Farhat. Geometric conservation laws for flow problems with moving boundaries and deformable meshes, and their impact on aeroelastic computations. *Comput. Method. Appl. M.* 134 (1996), 71-90.
- [42] C. K. Lombard and P. D. Thomas. Geometric conservation law and its application to flow computations on moving grids. *AIAA J.*, 17 (1979), 1030-1037.
- [43] I. Lomtev, R. M. Kirby and G. E. Karniadakis. A Discontinuous Galerkin ALE Method for Compressible Viscous Flows in Moving Domains. *J. Comput. Phys.* 155 (1999), 128-159.
- [44] R. Loubère, P. H. Maire, M. Shashkov, J. Breil and S. Galera. ReALE: A reconnection-based arbitrary-Lagrangian-Eulerian method. *J. Comput. Phys.* 229 (2010), 4724-4761.
- [45] F. Marinacci, R. Pakmor and V. Springel. The formation of disc galaxies in high-resolution moving-mesh cosmological simulations. *Mon. on Not. R. Astron. Soc.* 437 (2013), 1750-1775.
- [46] D. J. Mavriplis and Z. Yang. Construction of the discrete geometric conservation law for high-order time-accurate simulations on dynamic meshes, *J. Comput. Phys.* 213 (2006), 557-573.
- [47] C. A. A. Minoli and D. A. Kopriva. Discontinuous Galerkin spectral element approximations on moving meshes. *J. Comput. Phys.* 230 (2011), 1876-1902.
- [48] M. S. Mock. Systems of conservation laws of mixed type. *J. Differ. Equations* 37 (1980), 70-88.
- [49] R. C. Moura, G. Mengaldo, J. Peiró and S. J. Sherwin. On the eddy-resolving capability of high-order discontinuous Galerkin approaches to implicit LES / under-resolved DNS of Euler turbulence. *J. Comput. Phys.* 330 (2017) 615-623.
- [50] V. T. Nguyen. An arbitrary Lagrangian–Eulerian discontinuous Galerkin method for simulations of flows over variable geometries. *J. Fluid. Struct.* 26 (2010), 312-329.
- [51] P. O. Persson, J. Bonet, J. Peraire. Discontinuous Galerkin Solution of the Navier-Stokes Equations on Deformable Domains. *Comput. Method. Appl. M.* 198 (2009), 1585-1595.
- [52] H. Ranocha. Generalised Summation-by-Parts Operators and Entropy Stability of Numerical Methods for Hyperbolic Balance Laws. Cuvillier Verlag, 2018.
- [53] C. W. Shu, W. S. Don, D. Gottlieb, O. Schilling and L. Jameson. Numerical convergence study of nearly incompressible, inviscid Taylor–Green vortex flow. *J. Sci. Comput.* 24 (2005), 1-27.
- [54] L. Shunn and F. Ham. Method of manufactured solutions applied to variable-density flow solvers. *Annual Research Briefs-2007*, Center for Turbulence Research (2007), 155-168.



- [55] V. Springel. E pur si muove: Galilean-invariant cosmological hydrodynamical simulations on a moving mesh. *Mon. on Not. R. Astron. Soc.* 401 (2010), 791-851.
- [56] Tadmor, Eitan. The numerical viscosity of entropy stable schemes for systems of conservation laws. I. *Math. Comput.* 49 (1987), 91-103.
- [57] E. Tadmor. Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems, *Acta Numer.* 12 (2003). 451-512.
- [58] E. F. Toro. *Riemann solvers and numerical methods for fluid dynamics: a practical introduction*. Springer Science & Business Media, 2013.
- [59] L. Wang and P. O. Persson. High-order Discontinuous Galerkin Simulations on Moving Domains using ALE Formulations and Local Remeshing and Projections, 53rd AIAA Aerospace Sciences Meeting. AIAA SciTech Forum (AIAA 2015-0820). Available via <https://dx.doi.org/10.2514/6.2015-0820> (Cited December 24, 2018)
- [60] N. Wintermeyer, A. R. Winters, G. J. Gassner and D. A. Kopriva. An entropy stable nodal discontinuous Galerkin method for the two dimensional shallow water equations on unstructured curvilinear meshes with discontinuous bathymetry. *J. Comput. Phys.* 340 (2017), 200-242.
- [61] A. R. Winters, R. C. Moura, G. Mengaldo, G. J. Gassner, S. Walch, J. Peiro, S. J. Sherwin. A comparative study on polynomial dealiasing and split form discontinuous Galerkin schemes for under-resolved turbulence computations. *J. Comput. Phys.* 372 (2018), 1-21.
- [62] A. R. Winters, D. Derigs, G. J. Gassner and S. Walch. A uniquely defined entropy stable matrix dissipation operator for high Mach number ideal MHD and compressible Euler simulations. *J. Comput. Phys.* 332 (2017), 274-289.
- [63] A. R. Winters and D. A. Kopriva. ALE-DGSEM approximation of wave reflection and transmission from a moving medium. *J. Comput. Phys.* 263 (2014), 233-267.
- [64] A. R. Winters. *Discontinuous Galerkin spectral element approximations for the reflection and transmission of waves from moving material interfaces*. Diss. The Florida State University, 2014.

## A Proof for Lemma 3.1

In this section, we prove Lemma 3.1. For the sake of simplicity we present merely the proof in two dimensions. The three dimensional proof can be done by the same argumentation.

The two dimensional bijective isoparametric transfinite mapping can be computed as in Kopriva [37] Chapter 6, equation (6.18)). The mapping is for all  $[\xi^1, \xi^2]^T \in E$  given by

$$\begin{aligned}
 \bar{\chi}(\xi^1, \xi^2, t) = & \frac{1}{2} \left[ (1 - \xi^1) \mathbb{I}^N(\bar{\Gamma}_4)(\xi^2, t) + (1 + \xi^1) \mathbb{I}^N(\bar{\Gamma}_2)(\xi^2, t) \right] \\
 & + \frac{1}{2} \left[ (1 - \xi^2) \mathbb{I}^N(\bar{\Gamma}_1)(\xi^1, t) + (1 + \xi^2) \mathbb{I}^N(\bar{\Gamma}_3)(\xi^1, t) \right] \\
 & - \frac{1}{4} \left[ (1 + \xi^1) \left\{ (1 - \xi^2) \mathbb{I}^N(\bar{\Gamma}_1)(1, t) + (1 + \xi^2) \mathbb{I}^N(\bar{\Gamma}_3)(1, t) \right\} \right] \\
 & - \frac{1}{4} \left[ (1 - \xi^1) \left\{ (1 - \xi^2) \mathbb{I}^N(\bar{\Gamma}_1)(-1, t) + (1 + \xi^2) \mathbb{I}^N(\bar{\Gamma}_3)(-1, t) \right\} \right].
 \end{aligned} \tag{A.1}$$

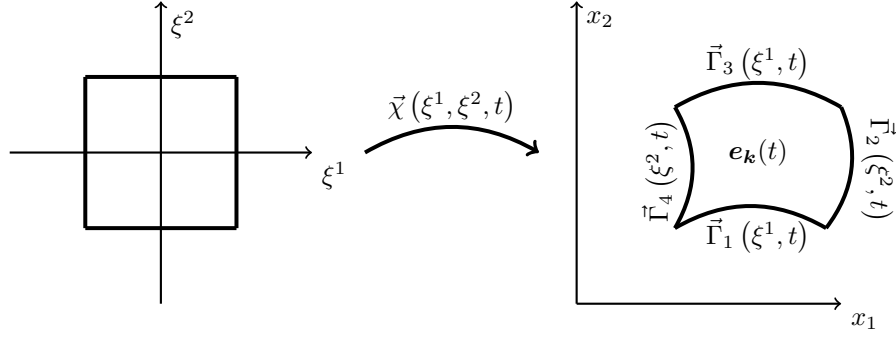


Figure A.1: Left the reference element  $E = [-1, 1]^2$  and on the right a general quadrilateral element  $e_k(t)$  with the curved faces  $\vec{\Gamma}_1(\xi^1, t)$ ,  $\vec{\Gamma}_2(\xi^2, t)$ ,  $\vec{\Gamma}_3(\xi^1, t)$  and  $\vec{\Gamma}_4(\xi^2, t)$ . The mapping  $\vec{\chi}(\xi^1, \xi^2, t)$  connects  $E$  and  $e_k(t)$ .

It is worth to mention that the mapping  $\vec{\chi}(\xi^1, \xi^2, t)$  matches with the boundary faces in the interpolation points. The location of the curved faces  $\vec{\Gamma}_1(\xi^1, t)$ ,  $\vec{\Gamma}_2(\xi^2, t)$ ,  $\vec{\Gamma}_3(\xi^1, t)$  and  $\vec{\Gamma}_4(\xi^2, t)$  is sketched in Figure A.1

In the following,  $e_1(t)$  and  $e_2(t)$  are two neighboring elements which share the same boundary face. Without loss of generality the elements share the face  $\vec{\Gamma}_3^1 = \vec{\Gamma}_1^2$  as it is illustrated in Figure A.2. Then, for the elements  $e_l(t)$ ,

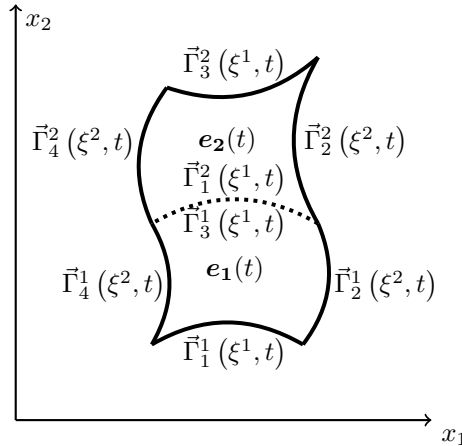


Figure A.2: Two elements  $e_1(t)$  and  $e_2(t)$  of a conforming mesh sharing the same curved boundary (dotted curve).

$l = 1, 2$ , the grid velocity field is given by

$$\begin{aligned}
 \vec{v}^l(\xi^1, \xi^2, t) = & \frac{1}{2} \left[ (1 - \xi^1) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_4^l \right) (\xi^2, t) + (1 + \xi^1) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_2^l \right) (\xi^2, t) \right] \\
 & + \frac{1}{2} \left[ (1 - \xi^2) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_1^l \right) (\xi^1, t) + (1 + \xi^2) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_3^l \right) (\xi^1, t) \right] \\
 & - \frac{1}{4} \left[ (1 + \xi^1) \left\{ (1 - \xi^2) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_1^l \right) (1, t) + (1 + \xi^2) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_3^l \right) (1, t) \right\} \right] \\
 & - \frac{1}{4} \left[ (1 - \xi^1) \left\{ (1 - \xi^2) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_1^l \right) (-1, t) + (1 + \xi^2) \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_3^l \right) (-1, t) \right\} \right],
 \end{aligned} \tag{A.2}$$

since it holds the identity

$$\frac{d}{dt} \mathbb{I}^N \left( \vec{\Gamma}_i^l \right) = \mathbb{I}^N \left( \frac{d}{dt} \vec{\Gamma}_i^l \right), \quad l = 1, 2, \quad \text{and} \quad i = 1, 2, 3, 4. \tag{A.3}$$

Furthermore, since for  $l = 1, 2$ , and  $i = 1, 2, 3, 4$ , the faces  $\vec{\Gamma}_i^l(\cdot, \cdot, t)$  are continuously differentiable in the time

interval  $[0, T]$ , it holds

$$\mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_4^1 \right) (1, t) = \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_3^1 \right) (-1, t), \quad \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_2^1 \right) (1, t) = \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_3^1 \right) (1, t), \quad (\text{A.4})$$

$$\mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_4^2 \right) (-1, t) = \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_1^2 \right) (-1, t), \quad \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_2^2 \right) (-1, t) = \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_1^2 \right) (1, t), \quad (\text{A.5})$$

$$\frac{d}{dt} \tilde{\Gamma}_3^1 (\zeta_j, t) = \frac{d}{dt} \tilde{\Gamma}_1^2 (\zeta_j, t), \quad j = 0, \dots, N, \quad (\text{A.6})$$

where  $\{\zeta_j\}_{j=0}^N$  are interpolation points.

For the element  $e_1(t)$  the points along the interface with  $e_2(t)$  are mapped in the set  $\{(\xi, 1) : \xi \in [-1, 1]\}$ . Hence, the grid velocity becomes

$$\vec{v}^1(\xi, 1, t) = \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_3^1 \right) (\xi, t), \quad \forall \xi \in [-1, 1] \quad (\text{A.7})$$

by (A.4). On the opposite, for the element  $e_2(t)$  the points along the interface with  $e_1(t)$  are mapped in the set  $\{(\xi, -1) : \xi \in [-1, 1]\}$  and we obtain

$$\vec{v}^2(\xi, -1, t) = \mathbb{I}^N \left( \frac{d}{dt} \tilde{\Gamma}_1^2 \right) (\xi, t), \quad \forall \xi \in [-1, 1], \quad (\text{A.8})$$

by (A.5). Thus, we obtain  $\vec{v}^1(\cdot, 1, t) = \vec{v}^2(\cdot, -1, t)$  by (A.6). This proves that the grid velocity is continuous in the interface points of the two neighboring elements.

## B Entropy stable moving mesh shallow water fluxes

In this section, we apply the methodology in Remark 2.2 to construct two entropy conservative moving mesh fluxes for the shallow water equations. The homogeneous shallow water equations (without bottom topography) are given by

$$\frac{\partial \mathbf{u}}{\partial t} + \vec{\nabla} \cdot \overset{\leftrightarrow}{\mathbf{f}} = \mathbf{0}. \quad (\text{B.1})$$

The state vector and the components of the block vector flux,  $\overset{\leftrightarrow}{\mathbf{f}}$ , are defined as

$$\mathbf{u} = \begin{bmatrix} h \\ hu_1 \\ hu_2 \end{bmatrix}, \quad \mathbf{f}_1 = \begin{bmatrix} hu_1 \\ hu_1^2 + \frac{g}{2}h^2 \\ hu_1u_2 \end{bmatrix}, \quad \mathbf{f}_2 = \begin{bmatrix} hu_2 \\ hu_1u_2 \\ hu_2^2 + \frac{g}{2}h^2 \end{bmatrix}, \quad (\text{B.2})$$

where the conserved states are the water height and the discharge  $h\vec{u} = [hu_1, hu_2]^T$ . The quantity  $g$  is the gravitational constant. The shallow water equations are equipped with the entropy/entropy flux pairs

$$s = \frac{1}{2}h|\vec{u}|^2 + \frac{1}{2}gh^2, \quad f_l^s = \frac{1}{2}hu_l|\vec{u}|^2 + gh^2u_l, \quad l = 1, 2. \quad (\text{B.3})$$

The entropy variables are

$$\mathbf{w} = \left[ gh - \frac{1}{2}|\vec{u}|^2, u_1, u_2 \right]^T. \quad (\text{B.4})$$

The entropy functionals are given by

$$\phi = \mathbf{w}^T \mathbf{u} - s = \frac{1}{2}gh^2, \quad \psi_l = \mathbf{w}^T \mathbf{f}_l - f_l^s = \frac{1}{2}gh^2u_l, \quad l = 1, 2. \quad (\text{B.5})$$

## B.1 Entropy conservative shallow water flux based on the flux in [60]

Friedrich et al. [23] Appendix A, Equation (A.4)] constructed the following state function

$$\mathbf{U}^\# = \begin{bmatrix} \{h\} \\ \{h\}\{u_1\} \\ \{h\}\{u_2\} \end{bmatrix}. \quad (\text{B.6})$$

The state function (B.6) is consistent with the shallow water state, symmetric and satisfies

$$[\mathbf{w}]^T \mathbf{U}^\# = \frac{1}{2} g [h^2] = [\phi]. \quad (\text{B.7})$$

In addition, an entropy conservative numerical flux for the shallow water equations has been constructed by Wintemeyer, Winters, Gassner, Kopriva (WGWK) [60]. The flux is given by

$$\mathbf{F}_1^{\text{EC-WGWK}} = \begin{bmatrix} \{hu_1\} \\ \{hu_1\}\{u_1\} + g\{h\}^2 - \frac{1}{2}\{h^2\} \\ \{hu_1\}\{u_2\} \end{bmatrix}, \quad \mathbf{F}_2^{\text{EC-WGWK}} = \begin{bmatrix} \{hu_2\} \\ \{hu_2\}\{u_1\} \\ \{hu_2\}\{u_2\} + g\{h\}^2 - \frac{1}{2}\{h^2\} \end{bmatrix}. \quad (\text{B.8})$$

The flux (B.8) is consistent, symmetric and satisfies

$$[\mathbf{w}]^T \mathbf{F}_l^{\text{EC-WGWK}} = \frac{1}{2} g [h^2 u_l] = [\psi_l], \quad l = 1, 2. \quad (\text{B.9})$$

The state function (B.6) and the flux (B.8) are used to construct the flux

$$\mathbf{G}_1^{\text{EC-WGWK}} = \mathbf{F}_1^{\text{EC-WGWK}} - \{v_1\} \mathbf{U}^\# = \begin{bmatrix} \{hu_1\} - \{h\}\{v_1\} \\ \{hu_1\}\{u_1\} - \{h\}\{u_1\}\{v_1\} + g\{h\}^2 - \frac{1}{2}\{h^2\} \\ \{hu_1\}\{u_2\} - \{h\}\{u_2\}\{v_1\} \end{bmatrix}, \quad (\text{B.10a})$$

and

$$\mathbf{G}_2^{\text{EC-WGWK}} = \mathbf{F}_2^{\text{EC-WGWK}} - \{v_2\} \mathbf{U}^\# = \begin{bmatrix} \{hu_2\} - \{h\}\{v_2\} \\ \{hu_2\}\{u_1\} - \{h\}\{u_1\}\{v_2\} \\ \{hu_2\}\{u_2\} - \{h\}\{u_2\}\{v_2\} + g\{h\}^2 - \frac{1}{2}\{h^2\} \end{bmatrix}. \quad (\text{B.10b})$$

The flux functions (B.10a), (B.10b) are consistent with  $\mathbf{g}_1$ ,  $\mathbf{g}_2$ , symmetric and it follows by (B.7) and (B.9)

$$[\mathbf{w}]^T \mathbf{G}_l^{\text{EC-WGWK}} = [\mathbf{w}]^T \mathbf{F}_l^{\text{EC-WGWK}} - \{v_l\} [\mathbf{w}]^T \mathbf{U}^\# = [\psi_l] - \{v_l\} [\phi], \quad l = 1, 2. \quad (\text{B.11})$$

## B.2 Entropy conservative shallow water flux based on the flux in [22]

Fjordholm, Mishra, Tadmor (FMT) constructed a further entropy conservative numerical flux for the shallow water equations in [22]. This flux is given by

$$\mathbf{F}_1^{\text{EC-FMT}} = \begin{bmatrix} \{h\}\{u_1\} \\ \{h\}\{u_1\}^2 + \frac{g}{2}\{h^2\} \\ \{h\}\{u_1\}\{u_2\} \end{bmatrix}, \quad \mathbf{F}_2^{\text{EC-FMT}} = \begin{bmatrix} \{h\}\{u_2\} \\ \{h\}\{u_1\}\{u_2\} \\ \{h\}\{u_2\}^2 + \frac{g}{2}\{h^2\} \end{bmatrix}. \quad (\text{B.12})$$

The state function (B.6) and the flux (B.12) are used to construct the flux

$$\mathbf{G}_1^{\text{EC-FMT}} = \begin{bmatrix} \{h\}\{u_1 - v_1\} \\ \{h\}\{u_1 - v_1\}\{u_1\} + \frac{1}{2}\{h^2\} \\ \{h\}\{u_1 - v_1\}\{u_2\} \end{bmatrix}, \quad \mathbf{G}_2^{\text{EC-FMT}} = \begin{bmatrix} \{h\}\{u_2 - v_2\} \\ \{h\}\{u_2 - v_2\}\{u_1\} \\ \{h\}\{u_2 - v_2\}\{u_2\} + \frac{1}{2}\{h^2\} \end{bmatrix}. \quad (\text{B.13})$$

The flux functions (B.13) are consistent with  $\mathbf{g}_1$ ,  $\mathbf{g}_2$ , symmetric and satisfy (B.11).

### B.3 Matrix dissipation term for the shallow water flux

In the Section 3.7 it has been shown that the matrix dissipation operators to stabilize the entropy conservative moving mesh flux are basically the same as on a static mesh. Thus, we use the shallow water fluxes from the previous Appendices B.1 and B.2 with the dissipation operators from Wintermeyer et al. [60]. In the following, the matrices to construct the entropy stable dissipation operators (3.92) are listed. The scaling matrix for the right eigenvalues is given by

$$\mathcal{T} = \text{diag} \left( \frac{1}{\sqrt{2g}}, \sqrt{h}, \frac{1}{\sqrt{2g}} \right) \quad (\text{B.14})$$

in both directions. Therefore, we apply the following discrete scaling matrix

$$\mathcal{T}^* = \text{diag} \left( \frac{1}{\sqrt{2g}}, \sqrt{\langle\langle h \rangle\rangle}, \frac{1}{\sqrt{2g}} \right). \quad (\text{B.15})$$

The average components of the other matrices to compute the dissipation term in the  $x_1$ -direction are

$$\mathcal{R}_1^* = \begin{bmatrix} 1 & 0 & 1 \\ \langle\langle u_1 \rangle\rangle - \sqrt{g\langle\langle h \rangle\rangle} & 0 & \langle\langle u_1 \rangle\rangle + \sqrt{g\langle\langle h \rangle\rangle} \\ \langle\langle u_2 \rangle\rangle & 1 & \langle\langle u_2 \rangle\rangle \end{bmatrix}, \quad (\text{B.16})$$

$$\Lambda_1 = \text{diag} \left( \left| \langle\langle u_1 - \nu_1 \rangle\rangle - \sqrt{g\langle\langle h \rangle\rangle} \right|, \left| \langle\langle u_1 - \nu_1 \rangle\rangle \right|, \left| \langle\langle u_1 - \nu_1 \rangle\rangle + \sqrt{g\langle\langle h \rangle\rangle} \right| \right). \quad (\text{B.17})$$

In the  $x_2$ -direction the components are given by

$$\mathcal{R}_2^* = \begin{bmatrix} 1 & 0 & 1 \\ \langle\langle u_1 \rangle\rangle & 1 & \langle\langle u_1 \rangle\rangle \\ \langle\langle u_2 \rangle\rangle - \sqrt{g\langle\langle h \rangle\rangle} & 0 & \langle\langle u_2 \rangle\rangle + \sqrt{g\langle\langle h \rangle\rangle} \end{bmatrix}, \quad (\text{B.18})$$

$$\Lambda_2 = \text{diag} \left( \left| \langle\langle u_2 - \nu_2 \rangle\rangle - \sqrt{g\langle\langle h \rangle\rangle} \right|, \left| \langle\langle u_2 - \nu_2 \rangle\rangle \right|, \left| \langle\langle u_2 - \nu_2 \rangle\rangle + \sqrt{g\langle\langle h \rangle\rangle} \right| \right). \quad (\text{B.19})$$

## C Entropy stable moving mesh Euler fluxes

We present entropy stable Cartesian fluxes  $\mathbf{G}_l^{\text{EC}}, l = 1, 2, 3$ , for the compressible Euler equations (4.1) equipped with the entropy/entropy flux pairs (4.8). Then the entropy variables are given by

$$\mathbf{w} = \left[ \frac{\gamma - \varsigma}{\gamma - 1} - \beta |\vec{u}|^2, 2\beta u_1, 2\beta u_2, 2\beta u_3, -2\beta \right]^T, \quad \text{with} \quad \beta := \frac{\rho}{2p} \quad (\text{C.1})$$

and the entropy functionals are given by

$$\phi = \mathbf{w}^T \mathbf{u} - s = \rho, \quad \psi_l = \mathbf{w}^T \mathbf{f}_l - f_l^s = \rho u_l, \quad l = 1, 2, 3. \quad (\text{C.2})$$

### C.1 Entropy conservative Euler flux based on the flux in [8]

In the following the logarithmic mean  $\langle\langle \cdot \rangle\rangle^{\log}$  will be used. For two positive states  $a^-$  and  $a^+$ , the logarithmic mean is defined by

$$\langle\langle a \rangle\rangle^{\log} := \begin{cases} \frac{\langle\langle a \rangle\rangle}{\langle\langle \log(a) \rangle\rangle}, & \text{if } a^- \neq a^+, \\ a^-, & \text{if } a^- = a^+. \end{cases} \quad (\text{C.3})$$

A numerically stable procedure to compute the logarithmic mean (C.3) is provided by Ismail and Roe [34, Appendix B]. Friedrich et al. [23, Theorem 3] constructed the following state function

$$\mathbf{U}^\# = \begin{bmatrix} \{\!\!\{\rho}\!\!\}^{\log} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_2}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_3}\!\!\} \\ \frac{\{\!\!\{\rho}\!\!\}^{\log}}{2(\gamma-1)\{\!\!\{\beta}\!\!\}^{\log}} + \frac{1}{2} \{\!\!\{\rho}\!\!\}^{\log} \overline{|\vec{u}|^2} \end{bmatrix}, \quad (\text{C.4})$$

where

$$\overline{|\vec{u}|^2} = 2 (\{\!\!\{u_1}\!\!\}^2 + \{\!\!\{u_2}\!\!\}^2 + \{\!\!\{u_3}\!\!\}^2) - (\{\!\!\{u_1^2}\!\!\} + \{\!\!\{u_2^2}\!\!\} + \{\!\!\{u_3^2}\!\!\}). \quad (\text{C.5})$$

The state function (C.4) is consistent, symmetric and it holds

$$[\mathbf{w}]^T \mathbf{U}^\# = [\rho] = [\phi]. \quad (\text{C.6})$$

Furthermore, Chandrashekar constructed in [8] a kinetic energy preserving and entropy conservative (KEPEC) numerical flux function for the compressible Euler equations. In the  $x_1$ -direction Chandrashekar's KEPEC flux is given by

$$\mathbf{F}_1^{\text{EC.CH}} = \begin{bmatrix} \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\}^2 + \frac{\{\!\!\{\rho}\!\!\}}{2\{\!\!\{\beta}\!\!\}} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\} \{\!\!\{u_2}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\} \{\!\!\{u_3}\!\!\} \\ \frac{\{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\}}{2(\gamma-1)\{\!\!\{\beta}\!\!\}^{\log}} + \frac{1}{2} \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\} \overline{|\vec{u}|^2} + \frac{\{\!\!\{\rho}\!\!\} \{\!\!\{u_1}\!\!\}}{2\{\!\!\{\beta}\!\!\}} \end{bmatrix}. \quad (\text{C.7})$$

The flux (C.7) is consistent with  $\mathbf{f}_1$  and symmetric. In particular, Chandrashekar proved that

$$[\mathbf{w}]^T \mathbf{F}_1^{\text{EC.CH}} = [\rho u_1] = [\psi_1]. \quad (\text{C.8})$$

The state function (C.4) and the flux function (C.7) are used to construct the flux

$$\mathbf{G}_1^{\text{EC.CH}} = \mathbf{F}_1^{\text{EC.CH}} - \{\!\!\{\nu_1}\!\!\} \mathbf{U}^\# = \begin{bmatrix} \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1 - \nu_1}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1 - \nu_1}\!\!\} \{\!\!\{u_1}\!\!\} + \frac{\{\!\!\{\rho}\!\!\}}{2\{\!\!\{\beta}\!\!\}} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1 - \nu_1}\!\!\} \{\!\!\{u_2}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1 - \nu_1}\!\!\} \{\!\!\{u_3}\!\!\} \\ \frac{\{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1 - \nu_1}\!\!\}}{2(\gamma-1)\{\!\!\{\beta}\!\!\}^{\log}} + \frac{1}{2} \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1 - \nu_1}\!\!\} \overline{|\vec{u}|^2} + \frac{\{\!\!\{\rho}\!\!\} \{\!\!\{u_1}\!\!\}}{2\{\!\!\{\beta}\!\!\}} \end{bmatrix}. \quad (\text{C.9})$$

The flux (C.9) is consistent with  $\mathbf{g}_1 = \mathbf{f}_1 - \nu_1 \mathbf{u}$ , symmetric and it follows

$$[\mathbf{w}]^T \mathbf{G}_1^{\text{EC.CH}} = [\mathbf{w}]^T \mathbf{F}_1^{\text{EC.CH}} - \{\!\!\{\nu_1}\!\!\} [\mathbf{w}]^T \mathbf{U}^\# = [\psi_1] - \{\!\!\{\nu_1}\!\!\} [\phi] \quad (\text{C.10})$$

by (C.6) and (C.8). In the same way, the  $x_2$ -direction and the  $x_3$ -direction of Chandrashekar's KEPEC flux can be used to construct

$$\mathbf{G}_2^{\text{EC.CH}} = \begin{bmatrix} \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_2 - \nu_2}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_1}\!\!\} \{\!\!\{u_2 - \nu_2}\!\!\} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_2}\!\!\} \{\!\!\{u_2 - \nu_2}\!\!\} + \frac{\{\!\!\{\rho}\!\!\}}{2\{\!\!\{\beta}\!\!\}} \\ \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_2 - \nu_2}\!\!\} \{\!\!\{u_3}\!\!\} \\ \frac{\{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_2 - \nu_2}\!\!\}}{2(\gamma-1)\{\!\!\{\beta}\!\!\}^{\log}} + \frac{1}{2} \{\!\!\{\rho}\!\!\}^{\log} \{\!\!\{u_2 - \nu_2}\!\!\} \overline{|\vec{u}|^2} + \frac{\{\!\!\{\rho}\!\!\} \{\!\!\{u_2}\!\!\}}{2\{\!\!\{\beta}\!\!\}} \end{bmatrix} \quad (\text{C.11})$$

and

$$\mathbf{G}_3^{\text{EC.CH}} = \begin{bmatrix} \{\rho\}^{\log} \{u_3 - \nu_3\} \\ \{\rho\}^{\log} \{u_1\} \{u_3 - \nu_3\} \\ \{\rho\}^{\log} \{u_2\} \{u_3 - \nu_3\} \\ \{\rho\}^{\log} \{u_3\} \{u_3 - \nu_3\} + \frac{\{\rho\}}{2\{\beta\}} \\ \frac{\{\rho\}^{\log} \{u_3 - \nu_3\}}{2(\gamma-1)\{\beta\}^{\log}} + \frac{1}{2} \{\rho\}^{\log} \{u_3 - \nu_3\} |\vec{u}|^2 + \frac{\{\rho\} \{u_3\}}{2\{\beta\}} \end{bmatrix}. \quad (\text{C.12})$$

The fluxes (C.11), (C.12) are consistent with  $\mathbf{g}_2 = \mathbf{f}_2 - \nu_2 \mathbf{u}$ ,  $\mathbf{g}_3 = \mathbf{f}_3 - \nu_3 \mathbf{u}$ , symmetric and satisfy

$$[\mathbf{w}]^T \mathbf{G}_2^{\text{EC.CH}} = [\mathbf{w}]^T \mathbf{F}_2^{\text{EC.CH}} - \{\nu_2\} [\mathbf{w}]^T \mathbf{U}^\# = [\psi_2] - \{\nu_2\} [\phi], \quad (\text{C.13})$$

$$[\mathbf{w}]^T \mathbf{G}_3^{\text{EC.CH}} = [\mathbf{w}]^T \mathbf{F}_3^{\text{EC.CH}} - \{\nu_3\} [\mathbf{w}]^T \mathbf{U}^\# = [\psi_3] - \{\nu_3\} [\phi]. \quad (\text{C.14})$$

## C.2 Entropy conservative Euler flux based on the flux in [52]

Ranocha constructed in his PHD thesis [52] another KEPEC numerical flux for the compressible Euler equations. We proceed as in the Appendix C.1 and use the state (C.4) and Ranocha's flux to construct the following two-point flux functions

$$\mathbf{G}_1^{\text{EC.R}} = \begin{bmatrix} \{\rho\}^{\log} \{u_1 - \nu_1\} \\ \{\rho\}^{\log} \{u_1 - \nu_1\} \{u_1\} + \{p\} \\ \{\rho\}^{\log} \{u_1 - \nu_1\} \{u_2\} \\ \{\rho\}^{\log} \{u_1 - \nu_1\} \{u_3\} \\ \left( \frac{\{\rho\}^{\log}}{2(\gamma-1)\{\beta\}^{\log}} + \frac{1}{2} \{\rho\}^{\log} \right) \{u_1 - \nu_1\} |\vec{u}|^2 + 2\{p\} \{u_1\} - \{pu_1\} \end{bmatrix}, \quad (\text{C.15a})$$

$$\mathbf{G}_2^{\text{EC.R}} = \begin{bmatrix} \{\rho\}^{\log} \{u_2 - \nu_2\} \\ \{\rho\}^{\log} \{u_1\} \{u_2 - \nu_2\} \\ \{\rho\}^{\log} \{u_2\} \{u_2 - \nu_2\} + \{p\} \\ \{\rho\}^{\log} \{u_2 - \nu_2\} \{u_3\} \\ \left( \frac{\{\rho\}^{\log}}{2(\gamma-1)\{\beta\}^{\log}} + \frac{1}{2} \{\rho\}^{\log} \right) \{u_2 - \nu_2\} |\vec{u}|^2 + 2\{p\} \{u_2\} - \{pu_2\} \end{bmatrix}, \quad (\text{C.15b})$$

and

$$\mathbf{G}_3^{\text{EC.R}} = \begin{bmatrix} \{\rho\}^{\log} \{u_3 - \nu_3\} \\ \{\rho\}^{\log} \{u_1\} \{u_3 - \nu_3\} \\ \{\rho\}^{\log} \{u_2\} \{u_3 - \nu_3\} \\ \{\rho\}^{\log} \{u_3\} \{u_3 - \nu_3\} + \{p\} \\ \left( \frac{\{\rho\}^{\log}}{2(\gamma-1)\{\beta\}^{\log}} + \frac{1}{2} \{\rho\}^{\log} \right) \{u_3 - \nu_3\} |\vec{u}|^2 + 2\{p\} \{u_3\} - \{pu_3\}. \end{bmatrix}. \quad (\text{C.15c})$$

The flux functions (C.15) are consistent with  $\mathbf{g}_1$ ,  $\mathbf{g}_2$ ,  $\mathbf{g}_3$ , symmetric and satisfy

$$[\mathbf{w}]^T \mathbf{G}_l^{\text{EC.R}} = [\psi_l] - \{\nu_l\} [\phi], \quad l = 1, 2, 3. \quad (\text{C.16})$$

## C.3 Matrix dissipation term for the Euler flux

We will use the Euler fluxes from the previous Appendices C.1 and C.2 with the dissipation operators from Winters et al. [62]. In the following, the matrices to construct the entropy stable dissipation operators (3.92) are listed. The average components of the dissipation term in the  $x_1$ -direction are given by

$$\mathcal{R}_1^\star = \begin{bmatrix} 1 & 1 & 0 & 0 & 1 \\ \{u_1\} - \bar{c} & \{u_1\} & 0 & 0 & \{u_1\} + \bar{c} \\ \{u_2\} & \{u_2\} & 1 & 0 & \{u_2\} \\ \{u_3\} & \{u_3\} & 0 & 1 & \{u_3\} \\ \bar{h} - \{u_1\} \bar{c} & \frac{1}{2} |\vec{u}|^2 & \{u_2\} & \{u_3\} & \bar{h} + \{u_1\} \bar{c} \end{bmatrix}, \quad (\text{C.17})$$

$$\mathcal{T}_1^\star = \text{diag} \left( \sqrt{\frac{\{\rho\}^{\log}}{2\gamma}}, \sqrt{\frac{(\gamma-1)}{\gamma} \{\rho\}^{\log}}, \sqrt{\frac{\{\rho\}}{2\{\beta\}}}, \sqrt{\frac{\{\rho\}}{2\{\beta\}}}, \sqrt{\frac{\{\rho\}^{\log}}{2\gamma}} \right), \quad (\text{C.18})$$

$$\Lambda_1 = \text{diag} (|\{u_1 - \nu_1\} - \bar{c}|, |\{u_1 - \nu_1\}|, |\{u_1 - \nu_1\}|, |\{u_1 - \nu_1\}|, |\{u_1 - \nu_1\} + \bar{c}|), \quad (\text{C.19})$$

where

$$\bar{c} := \sqrt{\frac{\gamma \{\rho\}}{2 \{\rho\}^{\log} \{\beta\}}}, \quad \bar{h} := \frac{\gamma}{2(\gamma-1) \{\beta\}^{\log}} + \frac{1}{2} |\vec{u}|^2. \quad (\text{C.20})$$

In the  $x_2$ -direction the components are given by

$$\mathcal{R}_2^\star = \begin{bmatrix} 1 & 0 & 1 & 0 & 1 \\ \{u_1\} & 1 & \{u_1\} & 0 & \{u_1\} \\ \{u_2\} - \bar{c} & 0 & \{u_2\} & 0 & \{u_2\} + \bar{c} \\ \{u_3\} & 0 & \{u_3\} & 1 & \{u_3\} \\ \bar{h} - \{u_2\} \bar{c} & \{u_1\} & \frac{1}{2} |\vec{u}|^2 & \{u_3\} & \bar{h} + \{u_2\} \bar{c} \end{bmatrix}, \quad (\text{C.21})$$

$$\mathcal{T}_2^\star = \text{diag} \left( \sqrt{\frac{\{\rho\}^{\log}}{2\gamma}}, \sqrt{\frac{\{\rho\}}{2\{\beta\}}}, \sqrt{\frac{(\gamma-1)}{\gamma} \{\rho\}^{\log}}, \sqrt{\frac{\{\rho\}}{2\{\beta\}}}, \sqrt{\frac{\{\rho\}^{\log}}{2\gamma}} \right), \quad (\text{C.22})$$

$$\Lambda_2 = \text{diag} (|\{u_2 - \nu_2\} - \bar{c}|, |\{u_2 - \nu_2\}|, |\{u_2 - \nu_2\}|, |\{u_2 - \nu_2\}|, |\{u_2 - \nu_2\} + \bar{c}|), \quad (\text{C.23})$$

and in the  $x_3$ -direction the components are given by

$$\mathcal{R}_3^\star = \begin{bmatrix} 1 & 0 & 0 & 1 & 1 \\ \{u_1\} & 1 & 0 & \{u_1\} & \{u_1\} \\ \{u_2\} & 0 & 1 & \{u_2\} & \{u_2\} \\ \{u_3\} - \bar{c} & 0 & 0 & \{u_3\} & \{u_3\} + \bar{c} \\ \bar{h} - \{u_3\} \bar{c} & \{u_1\} & \{u_2\} & \frac{1}{2} |\vec{u}|^2 & \bar{h} + \{u_3\} \bar{c} \end{bmatrix}, \quad (\text{C.24})$$

$$\mathcal{T}_3^\star = \text{diag} \left( \sqrt{\frac{\{\rho\}^{\log}}{2\gamma}}, \sqrt{\frac{\{\rho\}}{2\{\beta\}}}, \sqrt{\frac{\{\rho\}}{2\{\beta\}}}, \sqrt{\frac{(\gamma-1)}{\gamma} \{\rho\}^{\log}}, \sqrt{\frac{\{\rho\}^{\log}}{2\gamma}} \right), \quad (\text{C.25})$$

$$\Lambda_3 = \text{diag} (|\{u_3 - \nu_3\} - \bar{c}|, |\{u_3 - \nu_3\}|, |\{u_3 - \nu_3\}|, |\{u_3 - \nu_3\}|, |\{u_3 - \nu_3\} + \bar{c}|). \quad (\text{C.26})$$

## D Proofs of entropy conservation for advection terms

In this section, we apply the SBP formula or Abel transformation

$$\sum_{k=1}^K a_k (b_{k+1} - b_k) = a_K b_{K+1} - a_1 b_1 - \sum_{k=1}^{K-1} a_{k+1} (b_{k+1} - b_k) \quad (\text{D.1})$$

for two sequences  $\{a_k\}_{k=1}^K$  and  $\{b_k\}_{k=1}^{K+1}$ . Furthermore, we apply the following identities which result from the properties of the SBP operator  $\mathcal{Q}$

$$\sum_{i,j=0}^N \mathcal{Q}_{ij} [a]_{(i,j)} \{b\}_{(i,j)} = \sum_{i,j=0}^N \mathcal{Q}_{ij} a_i b_j - (a_N b_N - a_0 b_0), \quad (\text{D.2})$$

$$\sum_{i,j=0}^N \mathcal{Q}_{ij} [a]_{(i,j)} \{b\}_{(i,j)} \{c\}_{(i,j)} = \sum_{i,j=0}^N 2 \mathcal{Q}_{ij} a_i \{b\}_{(i,j)} \{c\}_{(i,j)} - (a_N b_N c_N - a_0 b_0 c_0), \quad (\text{D.3})$$

where  $\{a\}_{i=0}^N$ ,  $\{b\}_{i=0}^N$  and  $\{c\}_{i=0}^N$  are generic nodal values. These identities can be proven in a similar way as the discrete split forms in Lemma 1 in [26]. Thus, we skip a proof in this paper.



## D.1 Proof for Theorem 2.1

We multiply the equation (2.16b) by the discrete entropy variables  $\mathbf{w}_k$  and sum over all elements  $I_k(t)$

$$\sum_{k=1}^K \mathbf{w}_k^T \left( \frac{dJ_k \mathbf{u}_k}{dt} \right) = - \sum_{k=1}^K \mathbf{w}_k^T \left( \mathbf{g}_{k+\frac{1}{2}}^* - \mathbf{g}_{k-\frac{1}{2}}^* \right). \quad (\text{D.4})$$

The same arguments as in the computation of (2.10) provide the identity

$$\begin{aligned} \mathbf{w}_k^T \left( \frac{dJ_k \mathbf{u}_k}{dt} \right) &= J_k \left( \frac{d}{dt} s_k(\mathbf{u}_k) \right) + \left( \frac{dJ_k}{dt} \right) \mathbf{w}_k^T \mathbf{u}_k \\ &= \left( \frac{d}{dt} J_k s_k(\mathbf{u}_k) \right) + \left( \frac{dJ_k}{dt} \right) \phi_k \\ &= \frac{d}{dt} (J_k s_k(\mathbf{u}_k)) + \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right) \phi_k, \end{aligned} \quad (\text{D.5})$$

where we used the D-GCL (2.16a) in the last step. Next, we sum the equation (D.5) over all elements  $I_k(t)$  and obtain

$$\sum_{k=1}^K \mathbf{w}_k^T \left( \frac{dJ_k \mathbf{u}_k}{dt} \right) = \frac{d}{dt} \sum_{k=1}^K J_k s_k(\mathbf{u}_k) + \sum_{k=1}^K \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right) \phi_k. \quad (\text{D.6})$$

By the entropy condition (2.19) and the SBP formula (D.1) follows

$$\begin{aligned} \sum_{k=1}^K \mathbf{w}_k^T \left( \mathbf{g}_{k+\frac{1}{2}}^* - \mathbf{g}_{k-\frac{1}{2}}^* \right) &= \mathbf{w}_K^T \mathbf{g}_{K+\frac{1}{2}}^* - \mathbf{w}_1^T \mathbf{g}_{\frac{1}{2}}^* - \sum_{k=1}^{K-1} (\mathbf{w}_{k+1} - \mathbf{w}_k)^T \mathbf{g}_{k+\frac{1}{2}}^* \\ &= \mathbf{w}_K^T \mathbf{g}_{K+\frac{1}{2}}^* - \mathbf{w}_1^T \mathbf{g}_{\frac{1}{2}}^* - \sum_{k=1}^{K-1} (\psi_{k+1} - \psi_k) + \sum_{k=1}^{K-1} \nu_{k+\frac{1}{2}}^* (\phi_{k+1} - \phi_k). \end{aligned} \quad (\text{D.7})$$

The first sum in the last line is a telescope sum and it follows

$$\sum_{k=1}^{K-1} (\psi_{k+1} - \psi_k) = \psi_K - \psi_1. \quad (\text{D.8})$$

To evaluate the second sum we use again the SBP formula (D.1) and obtain

$$\begin{aligned} \sum_{k=1}^{K-1} \nu_{k+\frac{1}{2}}^* (\phi_{k+1} - \phi_k) &= \sum_{k=1}^{K-1} \nu_{k+\frac{1}{2}}^* (\phi_{k+1} - \phi_k) + \sum_{k=1}^K \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right) \phi_k - \sum_{k=1}^K \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right) \phi_k \\ &= \nu_{K+\frac{1}{2}}^* \psi_K - \nu_{\frac{1}{2}}^* \psi_1 - \sum_{k=1}^K \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right) \phi_k. \end{aligned} \quad (\text{D.9})$$

Next, we plug the equations (D.8), (D.9) in (D.7) and obtain

$$\begin{aligned} \sum_{k=1}^K \mathbf{w}_k^T \left( \mathbf{g}_{k+\frac{1}{2}}^* - \mathbf{g}_{k-\frac{1}{2}}^* \right) &= - \sum_{k=1}^K \left( \nu_{k+\frac{1}{2}}^* - \nu_{k-\frac{1}{2}}^* \right) \phi_k \\ &\quad + \mathbf{w}_K^T \mathbf{g}_{K+\frac{1}{2}}^* - \psi_K + \nu_{K+\frac{1}{2}}^* \psi_K \\ &\quad - \left( \mathbf{w}_1^T \mathbf{g}_{\frac{1}{2}}^* - \psi_1 + \nu_{\frac{1}{2}}^* \psi_1 \right). \end{aligned} \quad (\text{D.10})$$

Finally, we plug the equations (D.6) and (D.10) in (D.4) and obtain

$$\frac{d}{dt} \sum_{k=1}^K J_k s(\mathbf{u}_k) = - \left( \mathbf{w}_K^T \mathbf{g}_{K+\frac{1}{2}}^* - \psi_K + \nu_{K+\frac{1}{2}}^* \psi_K \right) + \left( \mathbf{w}_1^T \mathbf{g}_{\frac{1}{2}}^* - \psi_1 + \nu_{\frac{1}{2}}^* \psi_1 \right). \quad (\text{D.11})$$

## D.2 Proof for equation (3.82)

The flux  $\vec{\mathbf{G}}^{\text{EC}}$  satisfies the symmetry property (3.64) and the SBP property (3.58) provides

$$2\omega_i \mathcal{D}_{im} = 2\mathcal{Q}_{im} = \mathcal{Q}_{im} - \mathcal{Q}_{mi} + \mathcal{B}_{im}, \quad i, m = 0, \dots, N. \quad (\text{D.12})$$

Thus, we obtain

$$\begin{aligned} \left\langle \vec{\mathbb{D}}^N \cdot \vec{\mathbf{G}}^{\text{EC}}, \mathbf{W} \right\rangle_N &= \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m,k=0}^N \mathcal{Q}_{im} [\mathbf{W}]_{(i,m)j}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \cdot \{\{J\vec{a}^1\}\}_{(i,m)jk} \right) \\ &+ \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{B}_{im} \mathbf{W}_{ijk}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \cdot \{\{J\vec{a}^1\}\}_{(i,m)jk} \right) \\ &+ \sum_{i,k=0}^N \omega_i \omega_k \sum_{j,m=0}^N \mathcal{Q}_{jm} [\mathbf{W}]_{i(j,m)k}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{imk}, \mathbf{U}_{ijk}, \mathbf{U}_{imk}) \cdot \{\{J\vec{a}^2\}\}_{i(j,m)k} \right) \\ &+ \sum_{i,k=0}^N \omega_i \omega_k \sum_{j,m=0}^N \mathcal{B}_{jm} \mathbf{W}_{ijk}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{imk}, \mathbf{U}_{ijk}, \mathbf{U}_{imk}) \cdot \{\{J\vec{a}^2\}\}_{i(j,m)k} \right) \\ &+ \sum_{i,j=0}^N \omega_i \omega_j \sum_{k,m=0}^N \mathcal{Q}_{km} [\mathbf{W}]_{ij(k,m)}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{ijm}, \mathbf{U}_{ijk}, \mathbf{U}_{ijm}) \cdot \{\{J\vec{a}^3\}\}_{ij(k,m)} \right) \\ &+ \sum_{i,j=0}^N \omega_i \omega_j \sum_{k,m=0}^N \mathcal{B}_{km} \mathbf{W}_{ijk}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{ijm}, \mathbf{U}_{ijk}, \mathbf{U}_{ijm}) \cdot \{\{J\vec{a}^3\}\}_{ij(k,m)} \right) \end{aligned} \quad (\text{D.13})$$

by the same calculation as in [23] Appendix C.1., Equations (C.4) and (C.5)] or [27] Appendix B.3., Equation (B.31)].

To evaluate the fluxes  $\vec{\mathbf{G}}^{\text{EC}}$  at the element interfaces, we apply the consistence condition (3.63), such that e.g.

$$\vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{Njk}, \vec{\nu}_{Njk}, \mathbf{U}_{Njk}, \mathbf{U}_{Njk}) = \vec{\mathbf{F}}(\mathbf{U}_{Njk}) - \{\{\vec{\nu}\}\}_{(N,N)jk} \mathbf{U}_{Njk} = \vec{\mathbf{G}}_{Njk}, \quad j, k = 0, \dots, N. \quad (\text{D.14})$$

This provides the identity

$$\begin{aligned} &\sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{B}_{im} \mathbf{W}_{ijk}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \cdot \{\{J\vec{a}^1\}\}_{(i,m)jk} \right) \\ &+ \sum_{i,k=0}^N \omega_i \omega_k \sum_{j,m=0}^N \mathcal{B}_{jm} \mathbf{W}_{ijk}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{imk}, \mathbf{U}_{ijk}, \mathbf{U}_{imk}) \cdot \{\{J\vec{a}^2\}\}_{i(j,m)k} \right) \\ &+ \sum_{i,j=0}^N \omega_i \omega_j \sum_{k,m=0}^N \mathcal{B}_{km} \mathbf{W}_{ijk}^T \left( \vec{\mathbf{G}}^{\text{EC}}(\vec{\nu}_{ijk}, \vec{\nu}_{ijm}, \mathbf{U}_{ijk}, \mathbf{U}_{ijm}) \cdot \{\{J\vec{a}^3\}\}_{ij(k,m)} \right) \\ &= \sum_{j,k=0}^N \omega_j \omega_k \left[ \mathbf{W}_{Njk}^T \left( \vec{\mathbf{G}}_{Njk} \cdot (J\vec{a}^1)_{Njk} \right) - \mathbf{W}_{0jk}^T \left( \vec{\mathbf{G}}_{0jk} \cdot (J\vec{a}^1)_{0jk} \right) \right] \\ &+ \sum_{i,k=0}^N \omega_i \omega_k \left[ \mathbf{W}_{iNk}^T \left( \vec{\mathbf{G}}_{iNk} \cdot (J\vec{a}^2)_{iNk} \right) - \mathbf{W}_{i0k}^T \left( \vec{\mathbf{G}}_{i0k} \cdot (J\vec{a}^2)_{i0k} \right) \right] \\ &+ \sum_{i,j=0}^N \omega_i \omega_j \left[ \mathbf{W}_{ijN}^T \left( \vec{\mathbf{G}}_{ijN} \cdot (J\vec{a}^3)_{ijN} \right) - \mathbf{W}_{ij0}^T \left( \vec{\mathbf{G}}_{ij0} \cdot (J\vec{a}^3)_{ij0} \right) \right] \\ &= \int_{\partial E, N} \mathbf{W}^T \left\{ \vec{\mathbf{G}} \cdot \hat{n} \right\} dS. \end{aligned} \quad (\text{D.15})$$

Next, we investigate the first sum on the right hand side in the equation (D.13). Since the fluxes  $\mathbf{G}_l^{\text{EC}}$ ,  $l = 1, 2, 3$ , satisfy the entropy condition (3.65), follows

$$\begin{aligned}
& \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} [\mathbf{W}]_{(i,m)jk}^T \left( \overset{\leftrightarrow}{\mathbf{G}}^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \cdot \{\mathbf{J}\vec{a}^1\}_{(i,m)jk} \right) \\
&= \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} [\mathbf{W}]_{(i,m)jk}^T \mathbf{G}_1^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \{Ja_1^1\}_{(i,m)jk} \\
&+ \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} [\mathbf{W}]_{(i,m)jk}^T \mathbf{G}_2^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \{Ja_2^1\}_{(i,m)jk} \\
&+ \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} [\mathbf{W}]_{(i,m)jk}^T \mathbf{G}_3^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \{Ja_3^1\}_{(i,m)jk} \tag{D.16} \\
&= \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} (\llbracket \Psi_1 \rrbracket_{(i,m)jk} - \{\nu_1\}_{(i,m)jk} \llbracket \Phi \rrbracket_{(i,m)jk}) \{Ja_1^1\}_{(i,m)jk} \\
&+ \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} (\llbracket \Psi_2 \rrbracket_{(i,m)jk} - \{\nu_2\}_{(i,m)jk} \llbracket \Phi \rrbracket_{(i,m)jk}) \{Ja_2^1\}_{(i,m)jk} \\
&+ \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} (\llbracket \Psi_3 \rrbracket_{(i,m)jk} - \{\nu_3\}_{(i,m)jk} \llbracket \Phi \rrbracket_{(i,m)jk}) \{Ja_3^1\}_{(i,m)jk}.
\end{aligned}$$

For  $l = 1, 2, 3$ , the SPB properties (D.2) and (D.3) provide

$$\begin{aligned}
& \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} (\llbracket \Psi_l \rrbracket_{(i,m)jk} - \{\nu_l\}_{(i,m)jk} \llbracket \Phi \rrbracket_{(i,m)jk}) \{Ja_l^1\}_{(i,m)jk} \\
&= - \sum_{j,k=0}^N \omega_j \omega_k \left[ \left( (\Psi_l)_{Njk} - (\nu_l)_{Njk} \Phi_{Njk} \right) (Ja_l^1)_{Njk} - \left( (\Psi_l)_{0jk} - (\nu_l)_{0jk} \Phi_{0jk} \right) (Ja_l^1)_{0jk} \right] \tag{D.17} \\
&+ \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k (\Psi_l)_{ijk} \sum_{m=0}^N \mathcal{D}_{im} (Ja_l^1)_{mjk} - \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{im} \{\nu_l\}_{(i,m)jk} \{Ja_l^1\}_{(i,m)jk}.
\end{aligned}$$

Hence, we obtain the identity

$$\begin{aligned}
& \sum_{j,k=0}^N \omega_j \omega_k \sum_{i,m=0}^N \mathcal{Q}_{im} [\mathbf{W}]_{(i,m)jk}^T \left( \overset{\leftrightarrow}{\mathbf{G}}^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{mjk}, \mathbf{U}_{ijk}, \mathbf{U}_{mjk}) \cdot \{\mathbf{J}\vec{a}^1\}_{(i,m)jk} \right) \\
&= - \sum_{j,k=0}^N \omega_j \omega_k \left[ \left( \left( \vec{\Psi} \right)_{Njk} - (\vec{\nu})_{Njk} \Phi_{Njk} \right) \cdot (\mathbf{J}\vec{a}^1)_{Njk} - \left( \left( \vec{\Psi} \right)_{0jk} - (\vec{\nu})_{0jk} \Phi_{0jk} \right) \cdot (\mathbf{J}\vec{a}^1)_{0jk} \right] \tag{D.18} \\
&+ \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \left( \vec{\Psi} \right)_{ijk} \cdot \left( \sum_{m=0}^N \mathcal{D}_{im} (\mathbf{J}\vec{a}^1)_{mjk} \right) \\
&- \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{im} \{\vec{\nu}\}_{(i,m)jk} \cdot \{\mathbf{J}\vec{a}^1\}_{(i,m)jk}.
\end{aligned}$$

By the same computation, the third sum on the right hand side in the equation (D.13) becomes

$$\begin{aligned}
& \sum_{i,k=0}^N \omega_i \omega_k \sum_{j,m=0}^N \mathcal{Q}_{jm} [\mathbf{W}]_{i(j,m)k}^T \left( \overset{\leftrightarrow}{\mathbf{G}}^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{imk}, \mathbf{U}_{ijk}, \mathbf{U}_{imk}) \cdot \{\{J\tilde{a}^2\}\}_{i(j,m)k} \right) \\
&= - \sum_{i,k=0}^N \omega_i \omega_k \left[ \left( \left( \vec{\Psi} \right)_{iNk} - (\vec{\nu})_{iNk} \Phi_{iNk} \right) \cdot (J\tilde{a}^2)_{iNk} - \left( \left( \vec{\Psi} \right)_{i0k} - (\vec{\nu})_{i0k} \Phi_{i0k} \right) (J\tilde{a}^2)_{i0k} \right] \\
&+ \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \left( \vec{\Psi} \right)_{ijk} \cdot \left( \sum_{m=0}^N \mathcal{D}_{jm} (J\tilde{a}^2)_{imk} \right) \\
&- \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{jm} \{\{ \vec{\nu} \}\}_{i(j,m)k} \cdot \{\{J\tilde{a}^2\}\}_{i(j,m)k}
\end{aligned} \tag{D.19}$$

and the sum next-to-last on the right hand side in the equation (D.13) becomes

$$\begin{aligned}
& \sum_{i,j=0}^N \omega_i \omega_j \sum_{k,m=0}^N \mathcal{Q}_{km} [\mathbf{W}]_{ij(k,m)}^T \left( \overset{\leftrightarrow}{\mathbf{G}}^{\text{EC}} (\vec{\nu}_{ijk}, \vec{\nu}_{ijm}, \mathbf{U}_{ijk}, \mathbf{U}_{ijm}) \cdot \{\{J\tilde{a}^3\}\}_{ij(k,m)} \right) \\
&= - \sum_{i,j=0}^N \omega_i \omega_j \left[ \left( \left( \vec{\Psi} \right)_{ijN} - (\vec{\nu})_{ijN} \Phi_{ijN} \right) \cdot (J\tilde{a}^3)_{ijN} - \left( \left( \vec{\Psi} \right)_{ij0} - (\vec{\nu})_{ij0} \Phi_{ij0} \right) (J\tilde{a}^3)_{ij0} \right] \\
&+ \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \left( \vec{\Psi} \right)_{ijk} \cdot \left( \sum_{m=0}^N \mathcal{D}_{km} (J\tilde{a}^3)_{ijm} \right) \\
&- \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{km} \{\{ \vec{\nu} \}\}_{ij(k,m)} \cdot \{\{J\tilde{a}^3\}\}_{ij(k,m)}.
\end{aligned} \tag{D.20}$$

The definition of the derivative projection operator (3.60) in the D-GCL (3.69a) provides

$$\begin{aligned}
& - \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{im} \{\{ \vec{\nu} \}\}_{(i,m)jk} \cdot \{\{J\tilde{a}^1\}\}_{(i,m)jk} \\
& - \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{jm} \{\{ \vec{\nu} \}\}_{i(j,m)k} \cdot \{\{J\tilde{a}^2\}\}_{i(j,m)k} \\
& - \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \sum_{m=0}^N 2 \mathcal{D}_{km} \{\{ \vec{\nu} \}\}_{ij(k,m)} \cdot \{\{J\tilde{a}^3\}\}_{ij(k,m)} \\
&= - \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \Phi_{ijk} \left( \vec{\mathbb{D}}^N \cdot \vec{\nu}_{ijk} \right) = - \left\langle \vec{\mathbb{D}}^N \cdot \vec{\nu}, \Phi \right\rangle_N.
\end{aligned} \tag{D.21}$$

Next, we plug the equations (D.15), (D.18), (D.19), (D.20) in the equation (D.13) and apply the identity (D.21). This results in the identity

$$\begin{aligned}
\left\langle \vec{\mathbb{D}}^N \cdot \overset{\leftrightarrow}{\mathbf{G}}^{\text{EC}}, \mathbf{W} \right\rangle_N &= \int_{\partial E, N} \left[ \mathbf{W}^T \left\{ \overset{\leftrightarrow}{\mathbf{G}} \cdot \hat{n} \right\} - \left( \vec{\Psi} - \vec{\nu} \Phi \right) \cdot \hat{n} \right] dS - \left\langle \vec{\mathbb{D}}^N \cdot \vec{\nu}, \Phi \right\rangle_N \\
&+ \sum_{i,j,k=0}^N \omega_i \omega_j \omega_k \left( \vec{\Psi} \right)_{ijk} \cdot \left( \sum_{m=0}^N \left( \mathcal{D}_{im} (J\tilde{a}^1)_{mjk} + \mathcal{D}_{jm} (J\tilde{a}^2)_{imk} + \mathcal{D}_{km} (J\tilde{a}^3)_{ijm} \right) \right).
\end{aligned} \tag{D.22}$$

The definition of the contravariant vector flux functions (3.22) and contravariant block vector flux functions (3.27) provide the equality

$$\begin{aligned}
 \mathbf{W}^T \left\{ \tilde{\mathbf{G}}^{\leftrightarrow} \cdot \hat{n} \right\} - \left( \vec{\Psi} - \vec{\nu} \Phi \right) \cdot \hat{n} &= \sum_{l,r=1}^3 \text{Ja}_l^r \left( \mathbf{W}^T \mathbf{G}_l - \Psi_l + \nu_l \Phi \right) \hat{n}^r \\
 &= \sum_{l,r=1}^3 \text{Ja}_l^r \left( F_l^s - \nu_l S \right) \hat{n}^r \\
 &= \left( \hat{s} \vec{n} \right) \cdot \left( \vec{F}^s - \vec{\nu} S \right) = \tilde{F}_{\hat{n}}^s - \tilde{\nu}_{\hat{n}} S.
 \end{aligned} \tag{D.23}$$

Therefore, the equation (D.22) simplifies to

$$\left\langle \vec{\mathbb{D}}^N \cdot \tilde{\mathbf{G}}^{\text{EC}}, \mathbf{W} \right\rangle_N = \int_{\partial E, N} \left( \tilde{F}_{\hat{n}}^s - \tilde{\nu}_{\hat{n}} S \right) dS - \left\langle \vec{\mathbb{D}}^N \cdot \vec{\nu}, \Phi \right\rangle_N, \tag{D.24}$$

since the discrete volume weighted contravariant vectors  $J\vec{a}^\alpha$ ,  $\alpha = 1, 2, 3$ , are computed by the conservative curl form (3.50) and the discrete metric identities (3.52) are satisfied in the LGL points.