Split Form Nodal Discontinuous Galerkin Schemes with Summation-By-Parts Property for the Compressible Euler Equations

Gregor J. Gassner[a,*], Andrew R. Winters[a], David Kopriva[b]

[a]*Mathematisches Institut, Universität zu Köln, Weyertal 86-90, 50931 Köln, Germany*
[b]*Department of Mathematics, The Florida State University, Tallahassee FL 32312, USA*

Abstract

Fisher and Carpenter (*High-order entropy stable finite difference schemes for non-linear conservation laws: Finite domains, Journal of Computational Physics, 252:518–557, 2013*) found a remarkable equivalence of general diagonal norm high-order summation-by-parts operators to a subcell based high-order finite volume formulation. This equivalence enables the construction of provably entropy stable schemes by a specific choice of the subcell finite volume flux. We show that besides the construction of entropy stable high order schemes, a careful choice of subcell finite volume fluxes generates split formulations of quadratic or cubic terms. Thus, by changing the subcell finite volume flux to a specific choice, we are able to generate, in a systematic way, all common split forms of the compressible Euler advection terms, such as the Ducros splitting and the Kennedy and Gruber splitting. Although these split forms are not entropy stable, we present a systematic way to prove which of those split forms are at least kinetic energy preserving. With this, we show we construct a unified high-order split form DG framework. We investigate with three dimensional numerical simulations of the inviscid Taylor-Green vortex and show that the new split forms enhance the robustness of high order simulations in comparison to the standard scheme when solving turbulent vortex dominated flows. In fact, we show that for certain test cases, the novel split form discontinuous Galerkin schemes are more robust than the discontinuous Galerkin scheme with over-integration.

Keywords: Ducros splitting, Kennedy and Gruber splitting, kinetic energy, discontinuous Galerkin spectral element method, 3D compressible Euler equations, Taylor-Green vortex, split form

1. Introduction

This paper presents robust nodal discontinuous Galerkin (DG) approximations for the advective terms of the three dimensional compressible Navier-Stokes equations, namely the compressible Euler equations. In the discontinuous Galerkin community, stabilising an approximation is frequently done by polynomial de-aliasing through over-integration [15, 23, 30]. The typical DG implementation is under-integrated. It is well-known that when the numerical quadrature in the DG scheme is constructed so that flux functions that depend linearly on the solution (e.g. linear advection equation) are integrated exactly, the approximation retains the formal order of accuracy [5, 6, 7, 8, 9]. The exact number of quadrature points depends on the polynomial ansatz space, on the element type and, of course, on the specific quadrature rule used. A similar concept is used for nodal DG schemes where the ansatz uses interpolation and (multi-)variate Lagrange-type basis functions. In many cases, DG discretisations for linear fluxes are directly applied to problems with non-linear flux functions by simply exchanging the linear flux function with the corresponding non-linear flux. The reasoning is clear, as the minimum number of quadrature (or interpolation) nodes necessary to obtain the expected order of convergence gives an implementation with the lowest number of arithmetic operations, and thus increased efficiency at first sight.

---

*Corresponding author.
Email address: ggassner@math.uni-koeln.de (Gregor J. Gassner)

1.1. Stabilisation strategies for discontinuous Galerkin based discretisations

It turns out that the strategy of minimal effort has a drastic impact when the numerical solution is under-resolved. In such cases, e.g. under-resolved turbulence or shocks, aliasing errors due to variational errors corrupt the approximate solution and may even drive a non-linear instability. Whereas such a non-linear instability is often masked by excessive artificial viscosity when using low order approximations, high-order discretisations with their lower inherent numerical dissipation are prone to such instabilities. In fact, without additional counter-measures such high-order discretisations are unstable and crash [23]. It is of course arguable why one should even consider under-resolution at all, as the accuracy of results with low resolution is at least questionable. However, recent investigations show that it is possible to achieve quite accurate results with (very) high-order DG schemes, even with under-resolution as long as proper de-aliasing mechanisms are augmented [15].

A very popular strategy is the use of polynomial de-aliasing, as mentioned above. Motivated by spectral methods, the authors of [23] proposed to de-alias by increasing the number of quadrature nodes according to the non-linearity of the flux function so that the variational terms are evaluated exactly. As a prime example, the non-linear Burger's equation has a quadratic flux function and hence roughly a factor of 1.5 times the number of interpolation nodes is needed in each spatial direction to integrate the non-linear flux exactly. It is also remarkable that for DG discretisations with exact evaluations of the integral, Shu and Jiang [21] proved a cell entropy inequality and, with this, non-linear L_2 stability. It is very important to note however, that Shu and Jiang consider *scalar* non-linear conservation laws and consequently non-linear stability of DG discretisations with exact integration is only valid for *scalar* conservation laws. Their stability estimate does not carry over to systems of non-linear conservation laws such as the compressible Euler equations, independent of the number of quadrature nodes used to evaluate the inner products.

In fact, recent results by [32] show that even with up to four times the number of quadrature nodes in each spatial direction, DG discretisations with either local Lax-Friedrichs or Roe's flux function crash for under-resolved turbulence simulations. The "up to four times the quadrature nodes" statements necessitates an additional remark: For the compressible Euler equations, the polynomial ansatz is typically done for the conserved quantities such as mass, momentum and energy. But, the flux functions are rational polynomials of the conserved quantities. This is problematic because the available quadrature rules are only exact for polynomial integrands. Thus, it is formally impossible to implement an exact integration based on standard quadrature rules for the compressible Euler equations.

To formally prove non-linear stability of DG discretisations for systems of conservation laws, it is necessary to reformulate the equations in terms of entropy variables. Here, the polynomial ansatz (and the test-function) space approximates the entropy variables and not the conserved variables. This is important, as it is now formally possible to show that the DG discretisation satisfies a cell entropy inequality for systems [17]. However, it is important to note, that again, the stability proof relies on exact evaluation of the variational terms, similar to the proof of Shu and Jiang [21]. As discussed above, in the case of the compressible Euler equations it is impossible (or impractical) to find a fixed number of quadrature nodes so that the variational errors due to the rational non-linearity of the flux functions with respect to the entropy variables are zero. At least an adaptive numerical quadrature approach would be necessary, which of course makes this strategy quite cumbersome with respect to implementation and efficiency.

Up to now, the only known stability proof without relying on exact evaluation of inner products for the compressible Euler equations is presented in Carpenter et al. [2]. The proof is possible because it uses a very specific form of the DG discretisation. We note that in addition to the correct treatment of the volume integral terms, the numerical fluxes at the interfaces as well as the boundary conditions need to be treated carefully [33, 34, 38] to obtain entropy stability.

By choosing a nodal DG ansatz with Gauss-Lobatto (GL) nodes used for both interpolation and numerical integration, the so-called discontinuous Galerkin spectral element method with collocation (DGSEM) results, e.g. [25]. In the present work, we always use interpolation and integration based on the GL nodes. This variant of the DG methodology is special because it possesses the summation-by-parts (SBP) property. The discrete mass matrix $\mathbf{M}$ and the discrete derivative matrix $\mathbf{D}$ of the DGSEM satisfy all formal definitions of a SBP operator [37]

$$\mathbf{Q} := \mathbf{M}\,\mathbf{D} \quad \text{with} \quad \mathbf{Q} + \mathbf{Q}^T = \mathbf{B}, \tag{1.1}$$

where $\mathbf{B} = \mathrm{diag}([-1, 0, \ldots, 0, 1])$ is the boundary evaluation operator and the rows of $\mathbf{Q}$ are undivided differences, e.g. [2, 13]. Furthermore, the mass matrix is diagonal and is used to define a discrete L_2-norm, e.g. to compute errors. To be more specific, the DGSEM operators form a so-called diagonal norm SBP operator. It was possible in [2, 12] to construct a specific form of the DGSEM that satisfies a cell entropy type inequality for all conservation laws while retaining the nodal nature of the discretisation, i.e. without relying on exact evaluation of the inner products. The only necessary ingredient to achieve a cell entropy type inequality is a two-point numerical flux function that gives exact entropy conservation in a standard finite volume type discretisation [2]. A remarkable advantage of these derivations are that they do not rely on any DG specifics, but only on the SBP property. Thus, the non-linear stability results directly carry over to all discretisations constructed with SBP operators, where the mass matrix $\mathbf{M}$ (often called norm matrix) is diagonal as in the case of DGSEM.

1.2. Alternative stabilisation strategies for the compressible Euler equations

We leave the DG community momentarily and investigate de-aliasing strategies used in other high-order communities such as finite differences, e.g. [10, 22, 28, 29, 31] and spectral methods, e.g. [1, 40]. A very prominent example for de-aliasing is the use of alternative formulations of the non-linear advection terms, e.g. so-called split formulations. Due to the non-linear character of the advective terms of the Euler equations, there are many ways to re-write the equations. A good overview of the different split forms can be found in [36].

We consider here split forms of the three dimensional compressible Euler equations. In a general operator notation, we can write the Euler equations as

$$U_t + \mathcal{L}_x(U) + \mathcal{L}_y(U) + \mathcal{L}_z(U) = 0, \tag{1.2}$$

with the conservative variables

$$U = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \end{pmatrix} := \begin{pmatrix} \rho \\ \rho\, u \\ \rho\, v \\ \rho\, w \\ \rho\, e \end{pmatrix}, \tag{1.3}$$

where $\rho\, e = \rho\, \theta + \frac{1}{2}\rho\, (u^2 + v^2 + w^2)$ and ρ, u, v, w, p, θ, e are density, velocity, pressure, specific inner energy and specific total energy respectively, and $\mathcal{L}_{x,y,z}(U)$ are the non-linear spatial differential operators in the respective Cartesian direction x, y, z acting on U. We close the system by considering a perfect gas equation which relates the internal energy and pressure as $p = (\gamma - 1)\,\rho\theta$, where γ denotes the adiabatic coefficient.

If we consider, for instance, the standard divergence form of the Euler equations, the non-linear operators are

$$\mathcal{L}_x^{div}(U) := F(U)_x = \begin{bmatrix} \rho\, u \\ \rho\, u^2 + p \\ \rho\, u\, v \\ \rho\, u\, w \\ (\rho\, e + p)\, u \end{bmatrix}_x, \quad \mathcal{L}_y^{div}(U) := G(U)_y = \begin{bmatrix} \rho\, v \\ \rho\, u\, v \\ \rho\, v^2 + p \\ \rho\, v\, w \\ (\rho\, e + p)\, v \end{bmatrix}_y, \quad \mathcal{L}_z^{div}(U) := H(U)_z = \begin{bmatrix} \rho\, w \\ \rho\, u\, w \\ \rho\, v\, w \\ \rho\, w^2 + p \\ (\rho\, e + p)\, w \end{bmatrix}_z. \tag{1.4}$$

The divergence form (1.4) is typically used to construct a DG discretisation, as it directly gives discrete conservation of the resulting approximation. However, depending on how one interprets the non-linearities of the Euler flux (quadratic, cubic or rational) there are several ways to re-write the equations in an equivalent split form.

We use different split form approximations of derivatives of products to determine a particular splitting. The derivative of a product of two quantities is approximated by

$$(a\, b)_x = \frac{1}{2}\, (a\, b)_x + \frac{1}{2}\, (a_x\, b + a\, b_x), \tag{1.5}$$

3

which originates from the general split form

$$(a\,b)_x = \alpha\,(a\,b)_x + (1-\alpha)\,(a_x\,b + a\,b_x)\,, \quad \alpha \in \mathbb{R}, \tag{1.6}$$

when choosing $\alpha = 1/2$. We note that with the quadratic splitting it is possible to prove stability of linear variable coefficient problems [27].

For the derivative of cubic terms Kennedy and Gruber [22] proposed the following split form

$$\frac{\partial}{\partial x}(abc) = \alpha\frac{\partial}{\partial x}(abc) + \beta\left[a\frac{\partial}{\partial x}(bc) + bc\frac{\partial}{\partial x}(a)\right] + \kappa\left[b\frac{\partial}{\partial x}(ac) + ac\frac{\partial}{\partial x}(b)\right] + \delta\left[c\frac{\partial}{\partial x}(ab) + ab\frac{\partial}{\partial x}(c)\right]$$
$$+ \epsilon\left[bc\frac{\partial}{\partial x}(a) + ac\frac{\partial}{\partial x}(b) + ab\frac{\partial}{\partial x}(c)\right], \tag{1.7}$$

where $\epsilon = 1 - \alpha - \beta - \kappa - \delta$ and $\alpha, \beta, \kappa, \delta \in \mathbb{R}$. From all the possible combinations, we choose the case $\alpha = \beta = \kappa = \delta = \frac{1}{4}$ and $\epsilon = 0$, which gives

$$(a\,b\,c)_x = \frac{1}{4}\,(a\,b\,c)_x + \frac{1}{4}\,(a_x\,(b\,c) + b_x\,(a\,c) + c_x\,(a\,b)) + \frac{1}{4}\,(a\,(b\,c)_x + b\,(a\,c)_x + c\,(a\,b)_x)\,. \tag{1.8}$$

With the two split forms (1.5) and (1.8) it is now possible to compose new equivalent forms of the non-linear operator $\mathcal{L}(U)$ to enhance the stability of discretisations. For instance, Morinishi [31] introduced a skew-symmetric form for the momentum equations, which was rewritten in Gassner [14] assuming time continuity as

$$\mathcal{L}_x^{MO}(U) := \begin{bmatrix} (\rho\,u)_x \\ \frac{1}{2}\left((\rho u^2)_x + \rho u\,(u)_x + u\,(\rho u)_x\right) + p_x \\ \frac{1}{2}\left((\rho uv)_x + \rho u\,(v)_x + v\,(\rho u)_x\right) \\ \frac{1}{2}\left((\rho uw)_x + \rho u\,(w)_x + w\,(\rho u)_x\right) \\ ((\rho\theta + p)\,u)_x + \frac{1}{2}\left(\rho\,u^2\,(u)_x + u\,(\rho\,u^2)_x + \rho\,u\,v\,(v)_x + v\,(\rho\,u\,v)_x + \rho\,u\,w\,(w)_x + w\,(\rho\,u\,w)_x\right) \end{bmatrix}, \tag{1.9}$$

with the operators in the y and z direction defined similarly. It is important to note that the form of Morinishi does not use the split form of the quadratic terms ($\alpha = 1/2$), but the pure advective form ($\alpha = 0$ in (1.6)) in the energy equation. The form (1.9) allows the construction of a formally kinetic energy preserving discontinuous Galerkin method [14], but we will show in the numerical results section that the positive stabilisation effect of this alternative form of Morinishi is the lowest in comparison to the other forms presented below. A possible reason could be that only the advective form is used in the total energy equation.

In contrast to Morinishi's flux, Ducros et al. [10] proposed the following form, where only the split form ($\alpha = 1/2$) of the quadratic product (1.5) is used

$$\mathcal{L}_x^{DU}(U) := \begin{bmatrix} \frac{1}{2}\left((\rho u)_x + \rho(u)_x + u(\rho)_x\right) \\ \frac{1}{2}\left((\rho u^2)_x + \rho u(u)_x + u(\rho u)_x\right) + p_x \\ \frac{1}{2}\left((\rho uv)_x + \rho v(u)_x + u(\rho v)_x\right) \\ \frac{1}{2}\left((\rho uw)_x + \rho w(u)_x + u(\rho w)_x\right) \\ \frac{1}{2}\left(((\rho e + p)u)_x + (\rho e + p)(u)_x + u(\rho e + p)_x\right) \end{bmatrix}. \tag{1.10}$$

However, this alternative formulation does not lead to a discretisation that is formally kinetic energy preserving, e.g. [36].

The cubic form (1.8) was first introduced and applied in [22]. The operator proposed by Kennedy and

Gruber in the $x-$direction reads

$$
\mathcal{L}_x^{KG}(U) := \begin{bmatrix} \frac{1}{2}\left((\rho u)_x + \rho(u)_x + u(\rho)_x\right) \\ \frac{1}{4}\left[(\rho u^2)_x + \rho(u^2)_x + 2u(\rho u)_x + u^2(\rho)_x + 2\rho u(u)_x\right] + p_x \\ \frac{1}{4}\left[(\rho uv)_x + \rho(uv)_x + u(\rho v)_x + v(\rho u)_x + uv(\rho)_x + \rho v(u)_x + \rho u(v_x)\right] \\ \frac{1}{4}\left[(\rho uw)_x + \rho(uw)_x + u(\rho w)_x + w(\rho u)_x + uw(\rho)_x + \rho w(u)_x + \rho u(w_x)\right] \\ \frac{1}{2}\left((pu)_x + p(u)_x + u(p_x)\right) + \frac{1}{4}\left[(\rho eu)_x + \rho(eu)_x + e(\rho u)_x + u(\rho e)_x \right. \\ \left. + eu(\rho)_x + \rho u(e)_x + \rho e(u)_x\right] \end{bmatrix} . \tag{1.11}
$$

The standard quadratic form is used for the continuity equation, whereas the cubic form is used for the advection terms in the momentum equation. In the energy equation, both the quadratic form for $p\,u$ and the cubic form for $\rho\,e\,u$ are applied. Again, this form allows one to construct formally kinetic energy preserving discretisations [20].

In his overview, Pirozzoli [36] re-intrepreted the work of Kennedy and Gruber and used a similar but slightly different form of the flux splitting. Whereas the continuity and momentum equations don't change, Pirozzoli re-wrote the energy term $(\rho\,e + p)u$ as $\rho\,h\,p$, where the specific enthalpy is given by $h = e + p/\rho$. With this, only the cubic form (1.8) can be used for the energy equation, which results in

$$
\mathcal{L}_x^{PI}(U) := \begin{bmatrix} \frac{1}{2}\left((\rho u)_x + \rho(u)_x + u(\rho)_x\right) \\ \frac{1}{4}\left[(\rho u^2)_x + \rho(u^2)_x + 2u(\rho u)_x + u^2(\rho)_x + 2\rho u(u)_x\right] + p_x \\ \frac{1}{4}\left[(\rho uv)_x + \rho(uv)_x + u(\rho v)_x + v(\rho u)_x + uv(\rho)_x + \rho v(u)_x + \rho u(v_x)\right] \\ \frac{1}{4}\left[(\rho uw)_x + \rho(uw)_x + u(\rho w)_x + w(\rho u)_x + uw(\rho)_x + \rho w(u)_x + \rho u(w_x)\right] \\ \frac{1}{4}\left[(\rho uh)_x + \rho(uh)_x + h(\rho u)_x + u(\rho h)_x + uh(\rho)_x + \rho u(h)_x + \rho h(u)_x\right] \end{bmatrix} . \tag{1.12}
$$

There are other split forms available in literature, e.g. given by Kravchenko and Moin [28]. However, we will restrict this discussion to the forms presented above. The goal of the paper is to show how to discretise such forms for discontinuous Galerkin methods. More specifically, we focus on DGSEM (with GL nodes), as this specific variant satisfies the diagonal norm SBP (1.1) which is key to achieve a conservative approximation for the split forms, e.g. [27].

The remainder of the paper is organised as follows: In the next section the nodal collocation spectral element framework is introduced. The main Sec. 3 introduces the volume flux difference form and introduces specific numerical volume fluxes that generate known split forms of the compressible Euler equations. In Sec. 4 numerical experiments that support the theoretical findings are presented with our conclusions drawn in the last section.

2. The Nodal Discontinuous Galerkin Spectral Element Method

In this section, we provide the basic construction of a nodal discontinuous Galerkin spectral element method (DGSEM) on tensor product elements. The implementation of the DGSEM for the compressible Euler equations is based on [18], which includes a detailed description of the standard weak form discretisation of the compressible Euler equations in divergence form. It is important to note that we consider the strong form of the DGSEM in this work due to its close relation to the so-called simultaneous-approximation term (SAT) SBP finite difference methods [13]. In [26] it was shown that the weak and strong forms are algebraically equivalent. This equivalence of strong form and weak form is itself equivalent to the SBP property of an operator [13]. We specifically focus on the volume discretisation of the Euler terms. All other parts of the implementation remain unchanged. Thus, it is straightforward to extend an existing DGSEM code to the split form approximations presented in this paper. We restrict the following discussion and the numerical results section to Cartesian meshes as our focus is on the effect of the non-linear terms of the compressible Euler equations. For completeness, we provide the algebraic extensions to curvilinear meshes in Appendix B.

The first step in the discretisation is to subdivide the computational domain into non-overlapping hexahedral elements C. Each element is transformed to the reference element $C_0 = [-1, 1]^3$ with a polynomial mapping. The degree of the mapping is chosen to be at most the degree of the element-wise polynomial approximation N to ensure free-stream preservation [24]. For non-curved hexahedral elements, the mapping we use is trilinear and reads as

$$
\begin{aligned}
\underline{X}(\underline{\xi}) = (X(\underline{\xi}), Y(\underline{\xi}), Z(\underline{\xi}))^T = \frac{1}{8} \{ & \underline{x}_1(1 - \xi)(1 - \eta)(1 - \zeta) + \underline{x}_2(1 + \xi)(1 - \eta)(1 - \zeta) \\
& + \underline{x}_3(1 + \xi)(1 + \eta)(1 - \zeta) + \underline{x}_4(1 - \xi)(1 + \eta)(1 - \zeta) \\
& + \underline{x}_5(1 - \xi)(1 - \eta)(1 + \zeta) + \underline{x}_6(1 + \xi)(1 - \eta)(1 + \zeta) \\
& + \underline{x}_7(1 + \xi)(1 + \eta)(1 + \zeta) + \underline{x}_8(1 - \xi)(1 + \eta)(1 + \zeta) \},
\end{aligned}
\tag{2.1}
$$

where $\underline{x}_i$, $i = 1, \ldots, 8$ are the physical coordinates of the corners of the hexahedral element and $\underline{\xi} = (\xi, \eta, \zeta)^T$ are the reference coordinates. As we restrict ourselves here to Cartesian meshes, the Jacobian and metric terms simplify to

$$
J = \frac{1}{8}\Delta x \Delta y \Delta z, \quad X_\xi = \frac{1}{2}\Delta x, \quad Y_\eta = \frac{1}{2}\Delta y, \quad Z_\zeta = \frac{1}{2}\Delta z,
\tag{2.2}
$$

with element side lengths Δx, Δy, and Δz.

For each element, each component of the conservative variable vector is approximated by a polynomial of degree N in reference space, e.g. for the density

$$
u_1(x, y, z, t)\big|_C = \rho(x, y, z, t)\big|_C \approx \rho(\xi, \eta, \zeta, t) := \sum_{i,j,k=0}^{N} \rho^{i,j,k}(t)\, \ell_i(\xi)\, \ell_j(\eta) \ell_k(\zeta),
\tag{2.3}
$$

where $\{u_1^{i,j,k}(t)\}_{i,j,k=0}^{N}$ are the time dependent nodal degrees of freedom of the element C at the nodes (i, j, k). Each nodal interpolant is defined with $(N + 1)^3$ GL nodes $\{\xi_i\}_{i=0}^N$, $\{\eta_j\}_{j=0}^N$, and $\{\zeta_j\}_{j=0}^N$ in the reference cube C_0. The associated Lagrange basis functions are given by

$$
\ell_j(\xi) = \prod_{i=0, i \neq j}^{N} \frac{\xi - \xi_i}{\xi_j - \xi_i}, \qquad j = 0, ..., N,
\tag{2.4}
$$

and satisfy the cardinal property

$$
\ell_j(\xi_i) = \delta_{ij},
\tag{2.5}
$$

where δ_{ij} denotes Kronecker's symbol with $\delta_{ij} = 1$ for $i = j$ and $\delta_{ij} = 0$ for $i \neq j$. Integrating the polynomial Lagrange basis functions over the unit interval $[-1, 1]$ gives the GL quadrature weights $\{\omega_j\}_{j=0}^N$. The GL nodes and weights form a quadrature rule with integration precision $2N - 1$ that is used in the nodal DGSEM for the approximation of the weak form integrals. The Lagrange basis functions, $\{\ell_j\}_{j=0}^N$, are discretely orthogonal to each other, and the mass matrix is diagonal

$$
\mathbf{M} := \mathrm{diag}([\omega_0, ..., \omega_N]).
\tag{2.6}
$$

In addition to the discrete integration, the polynomial basis functions and the interpolation nodes form a discrete differentiation. We introduce the polynomial derivative matrix

$$
D_{ij} = \frac{\partial \ell_j}{\partial \xi}\bigg|_{\xi = \xi_i}, \quad i, j = 0, \ldots, N.
\tag{2.7}
$$

As mentioned in the introduction, the derivative operator (2.7) is special, as it satisfies the SBP-property (1.1) for all polynomial degrees N, e.g. [2, 13, 14]. We define the matrix $\mathbf{Q} := \mathbf{MD}$, which has the SBP-property $\mathbf{Q} + \mathbf{Q}^T = \mathbf{B} := \mathrm{diag}(-1, 0, \ldots, 0, 1)$. The SBP-property is used to mimic integration-by-parts, by manipulating the derivative matrix to become

$$
\mathbf{D} = \mathbf{M}^{-1}\mathbf{Q} = \mathbf{M}^{-1}\mathbf{B} - \mathbf{M}^{-1}\mathbf{Q}^T.
\tag{2.8}
$$

With all these ingredients, we are able to formulate the standard DGSEM approximation of the divergence form of the flux equations (1.2). We use the elemental mapping and the metric terms (2.2) to transform the system (1.2) into the reference element

$$J U_t + \widetilde{\mathcal{L}}_\xi(U) + \widetilde{\mathcal{L}}_\eta(U) + \widetilde{\mathcal{L}}_\zeta(U) = 0, \tag{2.9}$$

with

$$\widetilde{\mathcal{L}}_\xi(U) = Y_\eta Z_\zeta \mathcal{L}_\xi(U), \ \widetilde{\mathcal{L}}_\eta(U) = X_\xi Z_\zeta \mathcal{L}_\eta(U), \ \widetilde{\mathcal{L}}_\zeta(U) = X_\xi Y_\eta \mathcal{L}_\zeta(U), \tag{2.10}$$

where we used the restriction to Cartesian meshes to simplify the expressions.

We collect the three dimensional DGSEM approximation of the divergence form of the equations written in indicial notation. Consider a component l of the system (2.9) and a GL node (i, j, k). The DGSEM approximation in strong form is

$$(\mathcal{L}_\xi(U))_{ijk}^l \approx \left[F^{*,l}(1, \eta_j, \zeta_k; \underline{n}) - F_{Njk}^l \right] - \left[F^{*,l}(-1, \eta_j, \zeta_k; \underline{n}) - F_{0jk}^l \right] + \sum_{m=0}^N D_{im} F_{mjk}^l,$$

$$(\mathcal{L}_\eta(U))_{ijk}^l \approx \left[G^{*,l}(\xi_i, 1, \zeta_k; \underline{n}) - G_{iNk}^l \right] - \left[G^{*,l}(\xi_i, -1, \zeta_k; \underline{n}) - G_{i0k}^l \right] + \sum_{m=0}^N D_{jm} G_{imk}^l, \tag{2.11}$$

$$(\mathcal{L}_\zeta(U))_{ijk}^l \approx \left[H^{*,l}(\xi_i, \eta_j, 1; \underline{n}) - H_{ijN}^l \right] - \left[H^{*,l}(\xi_i, \eta_j, -1; \underline{n}) - H_{ij0}^l \right] + \sum_{m=0}^N D_{km} H_{ijm}^l,$$

where $l = 1, \ldots, 5$. We use collocation for the non-linear flux functions, e.g. $F_{ijk}^l := f^l(U_{ijk})$, and denote the outward pointing normal vector by $\underline{n}$. Due to the discontinuous nature of the approximation space, it is necessary to introduce numerical surface flux functions, which resolve the jumps at the interface. These specific surface flux functions depend on the left and right state values at an interface and are marked by a $*$. We will specify the numerical surface fluxes in Sec. 3.4.

Remark 1. We note, that in the case of over-integration the fluxes F_{mjk}^l, G_{imk}^l, H_{ijm}^l in sum in (2.11) need to be changed. Instead of using a collocation approach where the nodal coefficients of the fluxes are just evaluated with the nodal values of the discrete solution U, it is necessary to compute a proper L_2-projection of the non-linear fluxes onto the polynomial space of degree N. This can be, for instance, implemented with an intermediate quadrature grid with a high number of nodes (such that the integrals in the L_2-projection are evaluated exactly), see e.g. [15, 23, 30].

The resulting semi-discrete form is a coupled system of ordinary differential equations in time, which we integrate with a low storage 5-stage 4th order accurate explicit Runge-Kutta method, e.g. [3].

3. Split form stabilisation for DGSEM

In this section we demonstrate how to implement the different split forms. To do so, we begin with a standard strong form DGSEM implementation (2.11), and modify the discrete volume integrals that lead to $\sum_{m=0}^N D_{im} F_{mjk}^l$, $\sum_{m=0}^N D_{jm} G_{imk}^l$ and $\sum_{m=0}^N D_{km} H_{ijm}^l$ to include the different split forms presented in the introduction, Sec. 1.2.

3.1. DGSEM with numerical volume flux function

An important result that we use in this work is presented in Fisher et al. [12] and Carpenter et al. [2]. These authors showed that it is possible to rewrite the application of a differencing operator **D** with the

diagonal SBP property into an equivalent subcell based finite volume type differencing formulation

$$\sum_{m=0}^{N} D_{im} F_{mjk}^l = \frac{\bar{F}_{(i+1)jk}^l - \bar{F}_{(i)jk}^l}{\omega_i}, \quad i,j,k = 0, ..., N,$$

$$\sum_{m=0}^{N} D_{jm} G_{imk}^l = \frac{\bar{G}_{i(j+1)k}^l - \bar{G}_{i(j)k}^l}{\omega_j}, \quad i,j,k = 0, ..., N, \qquad (3.1)$$

$$\sum_{m=0}^{N} D_{km} H_{ijm}^l = \frac{\bar{H}_{ij(k+1)}^l - \bar{H}_{ij(k)}^l}{\omega_k}, \quad i,j,k = 0, ..., N,$$

with consistent auxiliary fluxes $\{\bar{F}_{(ii)jk}^l\}_{ii,j,k=0}^{(N+1),N,N}$, $\{\bar{G}_{i(jj)k}^l\}_{i,jj,k=0}^{N,(N+1),N}$ and $\{\bar{H}_{ij(kk)}^l\}_{i,j,kk=0}^{N,N,(N+1)}$. This result is important, as it directly gives conservation of diagonal norm SBP discretisations in the sense of Lax-Wendroff due to the telescoping flux differencing [12]. In addition to the desired conservation property, the flux differencing interpretation (3.1) enables the construction of entropy conserving discretisations without relying on exact integration.

Fisher et al. [12] and Carpenter et al. [2] used a diagonal norm SBP operator, the volume flux differencing relation (3.1), and the existence of a two-point entropy conserving flux function $F_{EC}^{\#} = F_{EC}^{\#}(U_{ijk}, U_{mjk})$ to obtain a high-order accurate discretisation

$$\frac{\bar{F}_{(i+1)jk}^l - \bar{F}_{(i)jk}^l}{\omega_i} \approx 2 \sum_{m=0}^{N} D_{im} F_{EC}^{\#,l}(U_{ijk}, U_{mjk}). \qquad (3.2)$$

The important theorems and proofs can be found e.g. in Fisher et al. [12] (Thms. 3.1 and 3.2 on page 15 and 18). By choosing the two-point entropy flux $F_{EC}^{\#}$ also as the numerical surface flux in a DGSEM discretisation, with the volume terms computed with (3.2), it was shown that the resulting DG discretisation conserves the (discrete) integral of the entropy [2]. Note that the goal of the work of Fisher and Carpenter et al. [2, 12] is not to construct an entropy conserving discretisation, but a discretisation that is entropy stable. However, from a scheme that exactly preserves entropy, it is possible to introduce dissipation terms, e.g. at the element interfaces so that the (discrete) integral of entropy is guaranteed to decrease (typically named *entropy stability*), e.g. [13, 33]. Thus, entropy conservation can be seen as an intermediate step to entropy stability.

In the present work, we rewrite the volume terms using the flux differencing form (3.2). However, we keep the expression general and use the yet to be specified numerical volume fluxes $F^{\#}$, $G^{\#}$, and $H^{\#}$

$$(\mathcal{L}_\xi(U))_{ijk}^l \approx \left[F^{*,l}(1, \eta_j, \zeta_k; \underline{n}) - F_{Njk}^l \right] - \left[F^{*,l}(-1, \eta_j, \zeta_k; \underline{n}) - F_{0jk}^l \right] + 2 \sum_{m=0}^{N} D_{im} F^{\#,l}(U_{ijk}, U_{mjk}),$$

$$(\mathcal{L}_\eta(U))_{ijk}^l \approx \left[G^{*,l}(\xi_i, 1, \zeta_k; \underline{n}) - G_{iNk}^l \right] - \left[G^{*,l}(\xi_i, -1, \zeta_k; \underline{n}) - G_{i0k}^l \right] + 2 \sum_{m=0}^{N} D_{jm} G^{\#,l}(U_{ijk}, U_{imk}), \quad (3.3)$$

$$(\mathcal{L}_\zeta(U))_{ijk}^l \approx \left[H^{*,l}(\xi_i, \eta_j, 1; \underline{n}) - H_{ijN}^l \right] - \left[H^{*,l}(\xi_i, \eta_j, -1; \underline{n}) - H_{ij0}^l \right] + 2 \sum_{m=0}^{N} D_{km} H^{\#,l}(U_{ijk}, U_{ijm}).$$

We show in the next section that the reformulation of the volume integrals in (3.3) plays an important role and that it is possible to generate many variants of DGSEM for the compressible Euler equations. We note that, in principle, it is possible to choose the numerical surface and volume flux differently. However, to restrict the myriads of possible combinations in this work, we couple the choice of the numerical volume flux and the numerical surface flux, detailed in Sec. 3.4 below.

3.2. Split form DSGEM

In this section, we discuss three simple identities used to introduce specific numerical volume flux functions that recover the alternative compressible Euler operator split forms discussed in the introduction. To

formulate these identities we consider three specific choices of the numerical volume flux, namely an arithmetic mean, the product of two arithmetic means and the product of three arithmetic means. We use the typical DG notation

$$\{\{a\}\}_{im} := \frac{1}{2}(a_i + a_m), \tag{3.4}$$

for an arithmetic mean of a generic nodal quantity a_i, $i = 0, ..., N$.

Lemma 1 (Discrete split forms). *Using the arithmetic mean, the product of two arithmetic means, or the product of three arithmetic means in the alternative numerical volume flux form of the DGSEM (3.3), it is possible to recover the standard DGSEM derivative form (3.1), the discrete quadratic split form (1.5) and the discrete cubic split form (1.8) respectively. More precisely, for generic nodal vector fields $\underline{a} = (a_0, ..., a_N)^T$, $\underline{b} = (b_0, ..., b_N)^T$, and $\underline{c} = (c_0, ..., c_N)^T$,*

$$2 \sum_{m=0}^{N} D_{im} \{\{a\}\}_{im} = (\mathbf{D}\,\underline{a})_i,$$

$$2 \sum_{m=0}^{N} D_{im} \{\{a\}\}_{im} \{\{b\}\}_{im} = \frac{1}{2}\left(\mathbf{D}\,\underline{\underline{a}}\,\underline{b} + \underline{\underline{a}}\,\mathbf{D}\,\underline{b} + \underline{\underline{b}}\,\mathbf{D}\,\underline{a}\right)_i,$$

$$2 \sum_{m=0}^{N} D_{im} \{\{a\}\}_{im} \{\{b\}\}_{im} \{\{c\}\}_{im} = \frac{1}{4}\left(\mathbf{D}\,\underline{\underline{a}}\,\underline{\underline{b}}\,\underline{c} + \underline{\underline{a}}\,\mathbf{D}\,\underline{\underline{b}}\,\underline{c} + \underline{\underline{b}}\,\mathbf{D}\,\underline{\underline{a}}\,\underline{c} + \underline{\underline{c}}\,\mathbf{D}\,\underline{\underline{a}}\,\underline{b} + \underline{\underline{b}}\,\underline{\underline{c}}\,\mathbf{D}\,\underline{a} + \underline{\underline{a}}\,\underline{\underline{c}}\,\mathbf{D}\,\underline{b} + \underline{\underline{a}}\,\underline{\underline{b}}\,\mathbf{D}\,\underline{c}\right)_i,$$
$$\tag{3.5}$$

where we introduce the additional notation that double underline denotes a matrix with a vector along the diagonal, e.g., $\underline{\underline{a}} = diag(\underline{a})$.

Proof. The proof of each of the split form identities is straightforward. However, each proof requires some algebraic manipulation unrelated to the focus of this paper. So we have moved the proofs to Appendix A.1. ∎

Remark 2. We note that the right hand sides of (3.5) are straightforward discretisations of the quadratic and the cubic split forms (1.5) and (1.8), respectively.

Remark 3. By combining the split form identities, it is easy to see that the pure advective form of the quadratic product can be generated with the numerical volume flux of the form $\{\{a\}\}_{im} \{\{b\}\}_{im} - \frac{1}{2} \{\{a\,b\}\}_{im}$.

Remark 4. We further note that all the derivations and proofs hold for general diagonal norm SBP operators and thus, for instance, all results in this paper directly carry over to finite difference diagonal norm SBP operators.

Remark 5. Analogously, it is possible to find equivalent forms of quartic products or products with even more terms when necessary.

With the identities presented in Lemma 1, it is now possible to state the second main result of this work, summarised in the following theorem.

Theorem 1 (Equivalent Numerical Volume Fluxes). *A high-order accurate and consistent DGSEM discretisation of the alternative Euler formulations of Morinishi (1.9), Ducros et al. (1.10), Kennedy and Gruber (1.11), and Priozzoli (1.12) can be generated by translating the quadratic and cubic split forms into the corresponding numerical volume flux. To compress the notation for the volume fluxes we note that the averages are taken in each spatial direction. For example, the average density in the x-direction or z-direction are given by*

$$F^{\#,1}(U_{ijk}, U_{mjk}) = \{\{\rho\}\} = \frac{1}{2}\left(\rho_{ijk} + \rho_{mjk}\right),$$

$$H^{\#,1}(U_{ijk}, U_{ijm}) = \{\{\rho\}\} = \frac{1}{2}\left(\rho_{ijk} + \rho_{ijm}\right),$$
$$\tag{3.6}$$

respectively. The volume fluxes for the standard DGSEM in the divergence form and for each alternative formulation of the Euler equations are:

Standard DG [25]:

$$
F^{\#}_{standard}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \{\{\rho u\}\} \\ \{\{\rho u^2\}\} + \{\{p\}\} \\ \{\{\rho uv\}\} \\ \{\{\rho uw\}\} \\ \{\{u(\rho e + p)\}\} \end{bmatrix}, \quad
G^{\#}_{standard}(U_{ijk}, U_{imk}) = \begin{bmatrix} \{\{\rho v\}\} \\ \{\{\rho uv\}\} \\ \{\{\rho v^2\}\} + \{\{p\}\} \\ \{\{\rho vw\}\} \\ \{\{v(\rho e + p)\}\} \end{bmatrix},
$$

(3.7)

$$
H^{\#}_{standard}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \{\{\rho w\}\} \\ \{\{\rho uw\}\} \\ \{\{\rho vw\}\} \\ \{\{\rho w^2\}\} + \{\{p\}\} \\ \{\{w(\rho e + p)\}\} \end{bmatrix}.
$$

Morinishi [31]:

$$
F^{\#}_{MO}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \{\{\rho\,u\}\} \\ \{\{\rho u\}\}\,\{\{u\}\} + \{\{p\}\} \\ \{\{\rho u\}\}\,\{\{v\}\} \\ \{\{\rho u\}\}\,\{\{w\}\} \\ \{\{(\rho\theta + p)\,u\}\} + \{\{\rho\,u^2\}\}\,\{\{u\}\} + \{\{\rho\,u\,v\}\}\,\{\{v\}\} + \{\{\rho\,u\,w\}\}\,\{\{w\}\} - \frac{1}{2}\left(\{\{\rho\,u^3\}\} + \{\{\rho\,u\,v^2\}\} + \{\{\rho\,u\,w^2\}\}\right) \end{bmatrix},
$$

$$
G^{\#}_{MO}(U_{ijk}, U_{imk}) = \begin{bmatrix} \{\{\rho\,v\}\} \\ \{\{\rho v\}\}\,\{\{u\}\} \\ \{\{\rho v\}\}\,\{\{v\}\} + \{\{p\}\} \\ \{\{\rho v\}\}\,\{\{w\}\} \\ \{\{(\rho\theta + p)\,v\}\} + \{\{\rho\,u\,v\}\}\,\{\{u\}\} + \{\{\rho\,v^2\}\}\,\{\{v\}\} + \{\{\rho\,v\,w\}\}\,\{\{w\}\} - \frac{1}{2}\left(\{\{\rho\,u^2\,v\}\} + \{\{\rho\,v^3\}\} + \{\{\rho\,v\,w^2\}\}\right) \end{bmatrix},
$$

(3.8)

$$
H^{\#}_{MO}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \{\{\rho\,w\}\} \\ \{\{\rho w\}\}\,\{\{u\}\} \\ \{\{\rho w\}\}\,\{\{v\}\} + \{\{p\}\} \\ \{\{\rho w\}\}\,\{\{w\}\} \\ \{\{(\rho\theta + p)\,w\}\} + \{\{\rho\,u\,w\}\}\,\{\{u\}\} + \{\{\rho\,v\,w\}\}\,\{\{v\}\} + \{\{\rho\,w^2\}\}\,\{\{w\}\} - \frac{1}{2}\left(\{\{\rho\,u^2\,w\}\} + \{\{\rho\,v^2\,w\}\} + \{\{\rho\,w^3\}\}\right) \end{bmatrix}.
$$

Ducros et al. [10]:

$$
F^{\#}_{DU}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \{\{\rho\}\}\,\{\{u\}\} \\ \{\{\rho u\}\}\,\{\{u\}\} + \{\{p\}\} \\ \{\{\rho v\}\}\,\{\{u\}\} \\ \{\{\rho w\}\}\,\{\{u\}\} \\ (\{\{\rho e\}\} + \{\{p\}\})\,\{\{u\}\} \end{bmatrix}, \quad
G^{\#}_{DU}(U_{ijk}, U_{imk}) = \begin{bmatrix} \{\{\rho\}\}\,\{\{v\}\} \\ \{\{\rho u\}\}\,\{\{v\}\} \\ \{\{\rho v\}\}\,\{\{v\}\} + \{\{p\}\} \\ \{\{\rho w\}\}\,\{\{v\}\} \\ (\{\{\rho e\}\} + \{\{p\}\})\,\{\{v\}\} \end{bmatrix},
$$

(3.9)

$$
H^{\#}_{DU}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \{\{\rho\}\}\,\{\{w\}\} \\ \{\{\rho u\}\}\,\{\{w\}\} \\ \{\{\rho v\}\}\,\{\{w\}\} \\ \{\{\rho w\}\}\,\{\{w\}\} + \{\{p\}\} \\ (\{\{\rho e\}\} + \{\{p\}\})\,\{\{w\}\} \end{bmatrix}.
$$

10

Kennedy and Gruber [22]:

$$F_{KG}^{\#}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \{\{\rho\}\} \{\{u\}\} \\ \{\{\rho\}\} \{\{u\}\}^2 + \{\{p\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{v\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{e\}\} + \{\{p\}\} \{\{u\}\} \end{bmatrix}, \quad G_{KG}^{\#}(U_{ijk}, U_{imk}) = \begin{bmatrix} \{\{\rho\}\} \{\{v\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{v\}\} \\ \{\{\rho\}\} \{\{v\}\}^2 + \{\{p\}\} \\ \{\{\rho\}\} \{\{v\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{v\}\} \{\{e\}\} + \{\{p\}\} \{\{v\}\} \end{bmatrix},$$

(3.10)

$$H_{KG}^{\#}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \{\{\rho\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{v\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{w\}\}^2 + \{\{p\}\} \\ \{\{\rho\}\} \{\{w\}\} \{\{e\}\} + \{\{p\}\} \{\{w\}\} \end{bmatrix}.$$

Pirozzoli [36]:

$$F_{PI}^{\#}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \{\{\rho\}\} \{\{u\}\} \\ \{\{\rho\}\} \{\{u\}\}^2 + \{\{p\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{v\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{h\}\} \end{bmatrix}, \quad G_{PI}^{\#}(U_{ijk}, U_{imk}) = \begin{bmatrix} \{\{\rho\}\} \{\{v\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{v\}\} \\ \{\{\rho\}\} \{\{v\}\}^2 + \{\{p\}\} \\ \{\{\rho\}\} \{\{v\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{v\}\} \{\{h\}\} \end{bmatrix},$$

(3.11)

$$H_{PI}^{\#}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \{\{\rho\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{u\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{v\}\} \{\{w\}\} \\ \{\{\rho\}\} \{\{w\}\}^2 + \{\{p\}\} \\ \{\{\rho\}\} \{\{w\}\} \{\{h\}\} \end{bmatrix}.$$

Proof. The proof follows directly when applying the results from Lem. 1 to the respective alternative Euler formulations. All of the alternative formulations are built from either the divergence form, the advective form, the quadratic or cubic split forms. We note that we use the first part of Lem. 1 to generate the standard flux divergence form. Furthermore, we need the pure advective form in the energy equation of the Morinishi (MO) form. To obtain the pure advective form we use the quadratic split form and subtract the divergence form scaled by one half, see Rem. 3. The other quadratic and cubic split forms are directly translated into their flux form using the second and third identity from the Lem. 1. Note that all fluxes are consistent, i.e.

$$F^{\#}(U_{ijk}, U_{ijk}) = F(U_{ijk}),$$

(3.12)

and symmetric in their arguments, e.g.

$$F^{\#}(U_{ijk}, U_{mjk}) = F^{\#}(U_{mjk}, U_{ijk}).$$

(3.13)

In the second part of the proof by Fisher et al. [12] it was shown via Taylor expansion (second part of the proof of Thm. 3.1 on pages 15 and 16) that for general two-point entropy conserving flux functions the volume flux differencing approximation (3.2) gives a high-order accurate scheme. However, the only properties that are used to prove high-order accuracy are consistency of the numerical volume flux and its symmetry. Thus, we can directly extend the proof by Fisher et al. to the numerical volume fluxes presented in Thm. 1, as all numerical volume fluxes are indeed symmetric and consistent in their arguments, as noted above. Another way of proving the high-order accuracy is to use again the identities of Lem. 1, as it is clear that the equivalent split form discretisations (right hand side of the identities in Lem. 1) are high-order accurate discretisations of the continuous split forms with the high-order accurate derivative matrix **D**. ∎

Remark 6. For central finite difference discretisations, Pirozzoli [35] already found a way to rewrite his alternative formulation (1.12) into an equivalent flux differencing form. In fact, ignoring boundary conditions, central finite difference schemes considered by Pirozzoli have the SBP property and thus the numerical volume flux described here (3.11) coincides with the one found by Pirozzoli.

Remark 7. To compare the split form DGSEM (3.3) to entropy conservative methods, we use the variants introduced by Carpenter et al. [2]. As stated above, the motivation of the work by Fisher et al. [12] and Carpenter et al. [2] is the construction of an entropy stable discretisation. The volume terms of the DGSEM are entropy conserving when using a two-point entropy conserving flux function in the volume flux differencing formulation [2]. The numerical surface fluxes can then be used to control the dissipation of the discretisation and to guarantee that entropy is decreasing, i.e. entropy stability. To generate two DGSEM variants with entropy conserving volume terms, we choose the following two numerical volume fluxes:

Ismail and Roe [19]: Introduce the parameter vector

$$\underline{z} = \left[\sqrt{\frac{\rho}{p}}, \ \sqrt{\frac{\rho}{p}}u, \ \sqrt{\frac{\rho}{p}}v, \ \sqrt{\frac{\rho}{p}}w, \ \sqrt{\rho p} \right]^T, \tag{3.14}$$

and the averaged quantities for the primitive variables

$$\hat{\rho} = \{\{z_1\}\} z_5^{\ln}, \quad \hat{u} = \frac{\{\{z_2\}\}}{\{\{z_1\}\}}, \quad \hat{v} = \frac{\{\{z_3\}\}}{\{\{z_1\}\}}, \quad \hat{w} = \frac{\{\{z_4\}\}}{\{\{z_1\}\}},$$

$$\hat{p}_1 = \frac{\{\{z_5\}\}}{\{\{z_1\}\}}, \quad \hat{p}_2 = \frac{\gamma+1}{2\gamma}\frac{z_5^{\ln}}{z_1^{\ln}} + \frac{\gamma-1}{2\gamma}\frac{\{\{z_5\}\}}{\{\{z_1\}\}}, \quad \hat{h} = \frac{\gamma \hat{p}_2}{\hat{\rho}(\gamma-1)} + \frac{1}{2}(\hat{u}^2 + \hat{v}^2 + \hat{w}^2), \tag{3.15}$$

where we introduce the logarithmic mean

$$(\cdot)^{\ln} := \frac{(\cdot)_L - (\cdot)_R}{\ln((\cdot)_L) - \ln((\cdot)_R)}, \tag{3.16}$$

needed for the entropy conserving flux functions. We note that a numerically stable procedure to compute the logarithmic mean is described by Ismail and Roe [19, Appendix B]. The entropy conserving fluxes in each physical direction read as

$$F_{IR}^{\#}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \hat{\rho}\hat{u} \\ \hat{\rho}\hat{u}^2 + \hat{p}_1 \\ \hat{\rho}\hat{u}\hat{v} \\ \hat{\rho}\hat{u}\hat{w} \\ \hat{\rho}\hat{u}\hat{h} \end{bmatrix}, \quad G_{IR}^{\#}(U_{ijk}, U_{imk}) = \begin{bmatrix} \hat{\rho}\hat{v} \\ \hat{\rho}\hat{u}\hat{v} \\ \hat{\rho}\hat{v}^2 + \hat{p}_1 \\ \hat{\rho}\hat{v}\hat{w} \\ \hat{\rho}\hat{v}\hat{h} \end{bmatrix}, \quad H_{IR}^{\#}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \hat{\rho}\hat{w} \\ \hat{\rho}\hat{u}\hat{w} \\ \hat{\rho}\hat{v}\hat{w} \\ \hat{\rho}\hat{w}^2 + \hat{p}_1 \\ \hat{\rho}\hat{w}\hat{h} \end{bmatrix}. \tag{3.17}$$

Chandrashekar [4]: Introduce notation for the inverse of the temperature

$$\beta = \frac{1}{RT} = \frac{\rho}{2p}, \tag{3.18}$$

and the average pressure and enthalpy

$$\hat{p} = \frac{\{\{\rho\}\}}{2\{\{\beta\}\}}, \quad \hat{h} = \frac{1}{2\beta^{\ln}(\gamma-1)} - \frac{1}{2}\left(\{\{u^2\}\} + \{\{v^2\}\} + \{\{w^2\}\}\right) + \frac{\hat{p}}{\rho^{\ln}} + \{\{u\}\}^2 + \{\{v\}\}^2 + \{\{w\}\}^2, \tag{3.19}$$

then we have the entropy conserving and kinetic energy preserving flux components

$$F_{CH}^{\#}(U_{ijk}, U_{mjk}) = \begin{bmatrix} \rho^{\ln}\{\{u\}\} \\ \rho^{\ln}\{\{u\}\}^2 + \hat{p} \\ \rho^{\ln}\{\{u\}\}\{\{v\}\} \\ \rho^{\ln}\{\{u\}\}\{\{w\}\} \\ \rho^{\ln}\{\{u\}\}\hat{h} \end{bmatrix}, \quad G_{CH}^{\#}(U_{ijk}, U_{imk}) = \begin{bmatrix} \rho^{\ln}\{\{v\}\} \\ \rho^{\ln}\{\{u\}\}\{\{v\}\} \\ \rho^{\ln}\{\{v\}\}^2 + \hat{p} \\ \rho^{\ln}\{\{v\}\}\{\{w\}\} \\ \rho^{\ln}\{\{v\}\}\hat{h} \end{bmatrix}, \quad H_{CH}^{\#}(U_{ijk}, U_{ijm}) = \begin{bmatrix} \rho^{\ln}\{\{w\}\} \\ \rho^{\ln}\{\{u\}\}\{\{w\}\} \\ \rho^{\ln}\{\{v\}\}\{\{w\}\} \\ \rho^{\ln}\{\{w\}\}^2 + \hat{p} \\ \rho^{\ln}\{\{w\}\}\hat{h} \end{bmatrix}. \tag{3.20}$$

3.3. Kinetic energy preservation of the split forms

In [20] Jameson derived a general condition on a numerical flux function for finite volume schemes to generate kinetic energy preserving discretisations. The kinetic energy is

$$\kappa := \frac{1}{2}\rho\,(u^2 + v^2 + w^2) = \frac{(\rho\,u)^2 + (\rho\,v)^2 + (\rho\,w)^2}{2\,\rho}. \tag{3.21}$$

The kinetic energy balance can be directly derived from the compressible Euler equations and reads

$$\frac{\partial\kappa}{\partial t} + \left(\tfrac{1}{2}\rho\,u\,(u^2+v^2+w^2)\right)_x + \left(\tfrac{1}{2}\rho\,v\,(u^2+v^2+w^2)\right)_y + \left(\tfrac{1}{2}\rho\,w\,(u^2+v^2+w^2)\right)_z + u\,p_x + v\,p_y + w\,p_z = 0. \tag{3.22}$$

Note, that the influence of the advective terms of the momentum flux $\rho\,u^2$, $\rho\,u\,v$, $\rho\,u\,w, \dots$ can be re-cast into flux form, whereas the pressure gradient in the momentum flux yields a non-conservative term (pressure work) in the kinetic energy balance (3.22). It is exactly this structure of the kinetic energy balance that is reflected in the definition of kinetic energy preserving by Jameson: Ignoring boundary conditions, a discretisation is termed kinetic energy preserving when the (discrete) integral of the kinetic energy is not changed by the advective terms, but only by the pressure work.

For finite volume schemes, Jameson [20] found the following general form for the components of the numerical surface fluxes of the momentum equations that give kinetic energy preservation:

$$\begin{aligned}
F^{*,2} &= F^{*,1}\,\{\{u\}\} + \widetilde{p}, & F^{*,3} &= F^{*,1}\,\{\{v\}\}, & F^{*,4} &= F^{*,1}\,\{\{w\}\}, \\
G^{*,2} &= G^{*,1}\,\{\{u\}\}, & G^{*,3} &= G^{*,1}\,\{\{v\}\} + \widetilde{p}, & G^{*,4} &= G^{*,1}\,\{\{w\}\}, \\
H^{*,2} &= H^{*,1}\,\{\{u\}\}, & H^{*,3} &= H^{*,1}\,\{\{v\}\}, & H^{*,4} &= H^{*,1}\,\{\{w\}\} + \widetilde{p},
\end{aligned} \tag{3.23}$$

where $\widetilde{p}$ can be any consistent numerical trace approximation of the pressure. This structure is enough to guarantee that the advective terms in the resulting discrete kinetic energy balance are consistent and in conservative form. We note that in the original conditions for kinetic energy preservation of Jameson, the pressure discretisation $\widetilde{p}$ can be any consistent approximation. However, we will show in the numerical results section that the discretisation of the pressure plays an important role for the discrete kinetic energy preservation. We will show numerically that the best results for discrete kinetic energy preservation are achieved when an arithmetic mean is used for $\widetilde{p}$.

If we inspect the numerical volume flux functions presented in Thm. 1, the conditions (3.23) hold for the variants MO, KG, PI and CH. We will demonstrate that the conditions (3.23) are enough to guarantee discrete kinetic energy preservation for the flux difference formulation (3.3).

Gassner [14] showed that Morinishi's alternative formulation allows one to construct a kinetic energy preserving DG discretisation. Using the same approach, it is possible to prove that KG and PI are kinetic energy preserving as well. In contrast, the method of (3.23) cannot be applied to the CH formulation, as no explicit split form is known. However, it is possible to make use of the flux differencing form of Sec. 3.1 to prove that numerical volume fluxes satisfying the conditions (3.23) generate high-order accurate schemes that are kinetic energy preserving in the sense of Jameson, i.e. that the advective terms can be re-cast into a conservative form and only the approximations of the pressure work is in non-conservative form. We summarise the third main result of this work in the following theorem.

Theorem 2 (Kinetic energy preservation)**.** *If the numerical volume flux functions $F^{\#}$, $G^{\#}$, and $H^{\#}$ satisfy the condition* (3.23)*, the volume terms of the resulting flux differencing discretisation* (3.3) *are kinetic energy preserving in the sense of Jameson [20], i.e. that the advective terms can be re-cast into a conservative form and only the approximations of the pressure work is in non-conservative form. More precisely, the volume terms of the discrete kinetic energy balance are given by*

$$\begin{aligned}
\left(\frac{\partial\kappa}{\partial t}\right)_{ijk} \approx\, & -2\sum_{m=0}^{N} D_{im}\,f^{\#,\kappa}(U_{ijk}, U_{mjk}) + D_{jm}\,g^{\#,\kappa}(U_{ijk}, U_{imk}) + D_{km}\,h^{\#,\kappa}(U_{ijk}, U_{ijm}) \\
& -2\sum_{m=0}^{N} u_{ijk}\,D_{im}\,\widetilde{p}(U_{ijk}, U_{mjk}) + v_{ijk}\,D_{jm}\,\widetilde{p}(U_{ijk}, U_{imk}) + w_{ijk}\,D_{km}\,\widetilde{p}(U_{ijk}, U_{ijm}),
\end{aligned} \tag{3.24}$$

13

with a consistent and symmetric pressure discretisation $\widetilde{p}$ and with the consistent and symmetric advective fluxes of the discrete kinetic energy

$$
\begin{aligned}
f^{\#,\kappa}(U_{ijk}, U_{mjk}) &:= \frac{1}{2}\, F^{\#,1}(U_{ijk}, U_{mjk})\,(u_{ijk}\, u_{mjk} + v_{ijk}\, v_{mjk} + w_{ijk}\, w_{mjk})\,, \\
g^{\#,\kappa}(U_{ijk}, U_{imk}) &:= \frac{1}{2}\, G^{\#,1}(U_{ijk}, U_{imk})\,(u_{ijk}\, u_{imk} + v_{ijk}\, v_{imk} + w_{ijk}\, w_{imk})\,, \\
h^{\#,\kappa}(U_{ijk}, U_{ijm}) &:= \frac{1}{2}\, H^{\#,1}(U_{ijk}, U_{ijm})\,(u_{ijk}\, u_{ijm} + v_{ijk}\, v_{ijm} + w_{ijk}\, w_{ijm})\,.
\end{aligned}
\tag{3.25}
$$

Proof. Again, we move the proof to the Appendix A.2. ∎

Remark 8. Note, that the terms of the discrete kinetic energy balance (3.24) are consistent to the continuous kinetic energy balance (3.22) and that the advective terms are in conservative form and the pressure work is in non-conservative form. This property guarantees that the discrete total kinetic energy is not dissipated by the advective terms. Hence, this property is important to construct discretisations that have low artificial dissipation of kinetic energy.

Remark 9. The kinetic energy preserving flux function proposed by Jameson is identical to the *PI* flux (3.11).

Corollary 1. *In Gassner [14] it was shown that when the volume terms of the DGSEM are kinetic energy preserving and the numerical surface flux function at element interfaces satisfies the general conditions (3.23) then the multi-element discretisation is kinetic energy preserving as well. Thus, if the numerical volume and surface flux are the same it follows that the DGSEM approximation satisfies the kinetic energy preservation condition (3.23). In particular, the variants MO, KG, PI and CH are kinetic energy preserving in the sense that the (discrete) integral of the kinetic energy is only changed by the pressure work (ignoring boundary conditions).*

3.4. Numerical surface flux functions

As mentioned above, we connect the choice of the volume numerical flux to the choice of the surface numerical flux. As a blueprint, we use the local Lax-Friedrichs numerical flux function, e.g.

$$
F^*(U_-, U_+) := \frac{1}{2}\left(F(U_-) + F(U_+)\right) - \frac{1}{2}\lambda_{max}\left[U_+ - U_-\right],
\tag{3.26}
$$

where λ_{max} is an estimate of the fastest wave speed at the interface. Note that the structure of the local Lax-Friedrichs flux (LLF) is a consistent symmetric part $\frac{1}{2}\left(F(U_-) + F(U_+)\right)$ and a stabilisation term $-\frac{1}{2}\lambda_{max}\left[U_+ - U_-\right]$, which introduces dissipation that depends on the jump of the approximation at the interface.

Analogously, we replace the consistent symmetric part by the consistent symmetric numerical volume flux function $F^{\#}$, which is also used to generate the volume terms of the DGSEM. We add a stabilisation term $-Stab(U_-, U_+)$, that we design so that it introduces dissipation at the interface depending on the jump of the approximate solution

$$
F^*(U_-, U_+) := F^{\#}(U_-, U_+) - Stab(U_-, U_+).
\tag{3.27}
$$

For the variants *MO*, *DU*, *KG*, and *PI* we choose the same stabilisation term as in the local Lax Friedrichs flux

$$
Stab_{MO}(U_-, U_+) = Stab_{DU}(U_-, U_+) = Stab_{KG}(U_-, U_+) = Stab_{PI}(U_-, U_+) := \frac{1}{2}\lambda_{max}\left[U_+ - U_-\right], \tag{3.28}
$$

where λ_{max} is the maximum of the maximum wave speeds from the left and right at the interface. The stabilisation term for *CH* is a direct 3D extension of the term introduced in [4] and is identical to (3.28) for

14

the first four components, but differs in the fifth component (energy equation):

$$Stab_{CH}^5(U_-, U_+) := \frac{1}{2}\lambda_{max}\left\{\left[\frac{1}{2(\gamma-1)\beta^{ln}} + (u^+u^- + v^+v^- + w^+w^-)\right][\rho^+ - \rho^-]\right.$$
$$\left. + \{\{\rho\}\}\{\{u\}\}[u^+ - u^-] + \{\{\rho\}\}\{\{v\}\}[v^+ - v^-] + \{\{\rho\}\}\{\{w\}\}[w^+ - w^-] + \frac{\{\{\rho\}\}}{2(\gamma-1)}\left[\frac{1}{\beta^+} - \frac{1}{\beta^-}\right]\right\}, \quad (3.29)$$

where β is introduced in (3.18).

Note that the standard LLF type dissipation terms in the first four components ensures guaranteed kinetic energy dissipation [4], whereas the specific fifth component of CH stabilisation term additionally ensures guaranteed entropy dissipation [4]. For the IR stabilisation, we use the term introduced in [2]

$$Stab_{IR}(U_-, U_+) = \frac{1}{2}\lambda_{max}\,\mathcal{H}\,[V^+ - V^-], \quad (3.30)$$

where $V = [\frac{\gamma-s}{\gamma-1} - \frac{\rho\|u\|^2}{2\,p}, \frac{\rho\,u}{p}, \frac{\rho\,v}{p}, \frac{\rho\,w}{p}, -\frac{\rho}{p}]^T$ are the entropy variables and $s = \ln(p) - \gamma\ln(\rho)$ is the specific thermodynamic entropy. The matrix $\mathcal{H}$ is the symmetric positive definite entropy Jacobian of the conservative variables

$$\mathcal{H} := \frac{\partial U}{\partial V} = \begin{bmatrix} \rho & \rho u & \rho v & \rho w & \rho e \\ \rho u & \rho u^2 + p & \rho uv & \rho uw & \rho hu \\ \rho v & \rho uv & \rho v^2 + p & \rho vw & \rho hv \\ \rho w & \rho uw & \rho vw & \rho w^2 + p & \rho hw \\ \rho e & \rho hu & \rho hv & \rho hw & \rho h^2 - \frac{a^2 p}{\gamma-1} \end{bmatrix}, \quad (3.31)$$

where the sound speed is defined as $a^2 = \frac{\gamma p}{\rho}$. The maximum wave speed λ_{max} and the entries of the entropy Jacobian $\mathcal{H}$ are evaluated at the average states used to compute the IR numerical flux functions (3.17). Again, this stabilisation term is designed so that the discrete entropy is dissipated at element interfaces. Thus, the variants IR and CH with stabilisation terms are entropy stable.

4. Numerical Investigations

The main purpose of this section is to investigate the robustness of the new split form DGSEM with respect to different choices of grid resolution and/or polynomial degree. We compare the robustness and behaviour of the standard DGSEM, the variant DU, the new split form variants with kinetic energy preserving volume terms MO, KG and PI, the variant with entropy conserving volume terms IR (Carpenter et al. [2]) and the variant with entropy conserving and kinetic energy preserving volume terms CH.

We start however with an investigation of the experimental order of convergence, discrete entropy conservation and kinetic energy preservation. All simulation results are obtained with an explicit five stage fourth order accurate low storage Runge-Kutta scheme, where the time step is computed according to a CFL type condition with the local maximum wave speed and the relative grid size $\Delta x/(N+1)$, where Δx is the element size. If not specified otherwise, we use $CFL = 0.5$ for all computations. The adiabatic coefficient is chosen as $\gamma = 1.4$.

4.1. h- and p-convergence

We first investigate the *h*- and *p*-convergence properties of the schemes. We choose the following manufactured solution for the unsteady compressible Euler equations

$$\rho = 2 + \frac{1}{10} \sin(\pi (x + y + z - 2t)),$$
$$u = 1,$$
$$v = 1,$$
$$w = 1,$$
$$\rho e = \left(2 + \frac{1}{10} \sin(\pi (x + y + z - 2t))\right)^2,$$

(4.1)

with the corresponding source term

$$q_\rho = c_1 \cos(\pi (x + y + z - 2t))$$
$$q_{\rho u} = c_2 \cos(\pi (x + y + z - 2t)) + c_3 \cos(2\pi (x + y + z - 2t)),$$
$$q_{\rho v} = c_2 \cos(\pi (x + y + z - 2t)) + c_3 \cos(2\pi (x + y + z - 2t)),$$
$$q_{\rho w} = c_2 \cos(\pi (x + y + z - 2t)) + c_3 \cos(2\pi (x + y + z - 2t)),$$
$$q_{\rho e} = c_4 \cos(\pi (x + y + z - 2t)) + c_5 \cos(2\pi (x + y + z - 2t)),$$

(4.2)

where $c_1 = \frac{\pi}{10}$, $c_2 = -\frac{1}{5}\pi + \frac{1}{20}\pi(1 + 5\gamma)$, $c_3 = \frac{\pi}{100}(\gamma - 1)$, $c_4 = \frac{1}{20}(-16\pi + \pi(9 + 15\gamma))$, and $c_5 = \frac{1}{100}(3\pi\gamma - 2\pi)$. The end time is set to $t_{end} = 10$. The computational domain is a fully periodic box with the size $[-1, 1]^3$.

As stated above, derivations in this work are valid for all diagonal norm SBP operators, such as those from the finite difference community. A conceptional difference of a spectral element method to a finite difference approximation is the ability for *p*-convergence. In *p*-convergence studies the grid is fixed and the number of the nodes inside an element is increased. In a DGSEM, increasing the number of element nodes automatically increases the polynomial degree of the approximation and hence increases the approximation order in tandem with the resolution. The resulting *spectral* convergence can be observed in Fig. 1, where all schemes are investigated with and without interface stabilisation on a regular 4^3 grid. As an example, the L_2-errors in density are shown. All other quantities show a similar behaviour. The L_2-norm is computed discretely with the collocated GL quadrature used for the scheme.

For this test, the errors of the different schemes are remarkably close, especially when interface stabilisation is activated. Note that without interface stabilisation, the numerical surface flux is symmetric in its arguments. This causes an odd/even effect, which can be clearly observed in the left plot of Fig. 1 and is in accordance to similar observations [13, 14, 16]. The simulations for higher polynomial degree N plateau out at about 10^{-8} in the right plot due to the accuracy of the time integration scheme. By decreasing the CFL from 0.5 to 0.25, we can see in the right plot of Fig. 1 that the level of the plateau decreases exactly by a factor of 16 as expected for the fourth order accurate RK method.

The next figure, Fig. 2, shows the *h*-convergence behaviour for the different split form DGSEM schemes without stabilisation for polynomial degree $N = 3$ and $N = 4$. Without interface stabilisation terms, we again obtain the odd/even behaviour: for odd polynomial degrees N we do not get the optimal convergence rate $N + 1$, but only a convergence rate of N.

If we switch on the interface stabilisation terms, again, the errors produced by the different fluxes are almost the same. All the schemes show now the optimal order of convergence, i.e. for a polynomial of degree N, we get $N + 1$st order convergence in *h* as can be seen in Fig. 3.

In summary, all schemes show the expected *p*- and *h*-convergence behaviour.

4.2. Entropy conservation and kinetic energy preservation

All presented schemes conserve mass, momentum, and total energy by construction. In this section, we focus on the auxiliary conservation properties of the methods and investigate the entropy conservation

Figure 1: p-convergence for all the schemes on a regular 4^3 grid. Plot of the L_2-error in density is shown. Left: without interface stabilisation. Right: with interface stabilisation.

Figure 2: h-convergence for all the schemes without interface stabilisation. Grid sequence goes from 2^3 up to 16^3. Plot of the L_2-error in density is shown. The odd/even effect can be seen. Left: $N = 3$, experimental order of convergence ≈ 3. Right: $N = 4$, experimental order of convergence ≈ 5.

Figure 3: h-convergence for the all schemes with interface stabilisation. Grid sequence goes from 2^3 up to 16^3. Plot of the L_2-error in density is shown. For polynomial degree N, we get $N+1$ convergence in h. Left: $N=3$. Right: $N=4$.

and kinetic energy preservation of the schemes. To investigate the auxiliary conservation properties, we deactivate the numerical dissipation introduced by the surface stabilisation terms and only use the symmetric and consistent parts at element interfaces.

In contrast to the last section, it does not make sense to investigate the auxiliary conservation properties with well resolved smooth solutions, as in this case all methods would converge spectrally fast to them, if the solution supports it. To make this investigation challenging, we consider the inviscid Taylor-Green vortex. The initial condition in the periodic $[0, 2\pi]^3$ box is

$$
\begin{aligned}
\rho &= 1, \\
u &= \sin(x)\,\cos(y)\,\cos(z), \\
v &= -\cos(x)\,\sin(y)\,\cos(z), \\
w &= 0, \\
p &= \frac{100}{\gamma} + \frac{1}{16}\left(\cos(2\,x)\,\cos(2\,z) + 2\,\cos(2\,y) + 2\,\cos(2\,x) + \cos(2\,y)\,\cos(2\,z)\right).
\end{aligned}
\tag{4.3}
$$

The evolution of these simple initial conditions is quite challenging due to the non-linear interaction of scales as well as the transition to a turbulence like flow field that occurs for large enough times. Without physical viscosity, there is no small scale limit and thus approximations of the inviscid Taylor-Green vortex are always under-resolved for large enough times. This behaviour makes this test case a challenge for the robustness of high-order methods.

Before we investigate the robustness of the schemes, we use this test setup to investigate their auxiliary conservation/preservation properties. The inviscid Taylor-Green vortex solution conserves both the total kinetic energy and the total entropy for all times. Although total kinetic energy is in general not a conserved quantity, it is for the specific inviscid Taylor-Green vortex setup in fully periodic domains. Due to periodicity, the pressure work when integrated over the domain cancels out and does not change the total kinetic energy. This allows us to experimentally investigate the actual kinetic energy behaviour of the different schemes. Note, that the kinetic energy preservation property of Jameson is only a statement for the influence of the advective terms on the total kinetic energy. However, we will show that discretisation errors in the pressure and velocity naturally cause errors in the pressure work contribution to the discrete total kinetic energy

18

as well, i.e. the discrete kinetic energy is not conserved discretely to machine precision for this case. Our investigations clearly show that the discretisation of the pressure matters and that the simple arithmetic mean yields the best results.

As mentioned above, these investigations only make sense when the interface stabilisation is omitted. Otherwise, the numerical viscosity would dissipate kinetic energy and entropy when the approximation is under-resolved. However, without interface stabilisation the resulting discretisations are basically dissipation free and the robustness is fragile. A general observation is that the higher the polynomial degree and the higher the overall number of spatial degrees of freedom, the harder it is to successfully run the simulations until the final time. Without interface stabilisation, the highest polynomial degree we could choose and still get meaningful results is $N = 3$. For $N = 3$ it is interesting to note that all proposed split form schemes run for all tested resolutions (up to 256^3), except for the standard and MO variants, which crash almost immediately. For higher polynomial degrees, the general trend is that low resolutions with 16^3 or 32^3 might run, but all schemes crash for resolutions greater than or equal to 64^3.

Nevertheless, the choice of polynomial degree $N = 3$ and 16^3 grid cells (64^3 degrees of freedom) allows us to investigate the auxiliary conservation properties. The left part of Fig. 4 shows the time evolution of the total entropy density

$$S = -\frac{\rho s}{\gamma - 1}, \quad s = \ln(p) - \gamma \ln(\rho), \tag{4.4}$$

for the different schemes. The right part shows the temporal evolution of the total kinetic energy.

Figure 4: Time evolution of the discrete total entropy and total kinetic energy for the case $N = 3$ and 16^3 grid cells. All results are obtained with $CFL = 0.1$.

Focusing first on the entropy conservation, as expected the IR and CH schemes conserve entropy basically to machine precision, whereas the other schemes show a small decay of the total entropy. However, the $y-$axis shows that the entropy dissipation of the other formulations is very small and arguably negligible.

The entropy conservation results lie in stark contrast to the kinetic energy preservation where substantial differences are observed. Note, that we have three schemes KG, PI and CH which, by construction, preserve the kinetic energy in the sense that the discrete pressure work changes the total kinetic energy, not the advective terms. We can see in the right part of Fig. 4 that KG and PI give similar results and indeed best preserve the total kinetic energy. Especially in comparison to the other discretisations, which show a loss of up to 10% of total kinetic energy. The IR scheme performs the worst.

There are two noteworthy remarks. First, the DU scheme conserves neither entropy nor preserves kinetic energy. However, this scheme has a lower loss of total kinetic energy compared to the entropy conserving

schemes IR and CH. Second, although the CH scheme is formally entropy conserving *and* kinetic energy preserving, the results clearly show that the loss of kinetic energy is much larger than for the KG and PI schemes. This is an unexpected result and demands further investigation. It turns out that although the advective terms do not formally contribute to the evolution of the kinetic energy as desired, the discretisation of the pressure plays a crucial role for the kinetic energy. Both, KG and PI use a simple arithmetic mean for the pressure and it turns out that this seems to be important for the kinetic energy evolution.

The CH scheme chooses the discretisation of the pressure in such a way that entropy is discretely conserved. The pressure discretisation is much more complicated than the simple arithmetic mean and a possible explanation is that this introduces additional errors that affect the discrete total kinetic energy. To support this claim, we modify the CH scheme by changing the pressure discretisation to the simple arithmetic mean. The right part of Fig. 5 shows the evolution of the discrete total kinetic energy. The CH scheme with the pressure fix now behaves like the KG and PI scheme and significantly reduces the loss of kinetic energy in comparison to the original CH scheme. However, the CH scheme with pressure fix is, of course, not entropy conservative anymore as shown in the left part of Fig. 5.

Figure 5: Comparison of original CH scheme and the CH scheme where the pressure discretisation is changed to a simple arithmetic mean, as in the KG and PI variants. The plot shows the time evolution of the discrete total entropy (left) and the discrete total kinetic energy (right) for the case $N = 3$ and 16^3 grid cells. All results are obtained with $CFL = 0.1$.

Summarising this section, the IR and CH schemes show machine precision entropy conservation as expected. The KG and PI schemes show near discrete kinetic energy conservation for the inviscid Taylor-Green vortex. The CH scheme shows discrete kinetic energy conservation only with a modification of the pressure discretisation, which demonstrates that for actual kinetic energy preservation the discretisation of the pressure has a strong impact. While the DU scheme does not possess any specific auxiliary conservation properties, it is still worth pointing out that it lost less of the total kinetic energy compared to the entropy conserving schemes.

4.3. Robustness

In this main results section, the robustness of the schemes is investigated. In contrast to the former investigations, we now activate the interface stabilisation. The interface stabilisation terms depend on the jump of the solution at the interface and introduce dissipation. The main result of the numerical investigations is that with the stabilisation terms, all split schemes except the MO variant are robust in

the sense that even for very high polynomial degrees the simulations do not crash even with severe under-resolution.

To make our point, we mimic a table presented in Moura et al. [32], where the stability of an over-integrated DG method for the inviscid Taylor-Green vortex is reported. It is important to note that for high resolutions, the over-integrated DG discretisations crashed and it was concluded that the dissipation from the surface integrals is not enough to stabilise the discretisation. In contrast to the over-integration results, all simulations performed with the DGSEM split forms listed in Tbl. 1 for the schemes KG, IR, PI, CH, and DU successfully run up to the final time of $t = 14$. Note that the configurations marked with a box are the ones that run to the final time when over-integration is used [32]. All other configurations could not be successfully completed, even when decreasing the time step and increasing the number of integration nodes up to a factor of four in each spatial direction [personal communication with Rodrigo Moura]. Thus, the entropy stable schemes and the new split form schemes offer a substantial advantage with respect to robustness compared to fully integrated DG.

DOFs/N	3	4	5	6	7
113^3	28	23	19	16	14
159^3	39	32	28	23	19
227^3	56	45	39	32	28
320^3	80	64	53	46	40
450^3	113	90	75	64	56

Table 1: List of grid cell and polynomial degree configurations for KG, PI, IR, DU, CH schemes that were investigated. The entries refer to the numbers of elements in one direction. None of these schemes crashed. The number of DOFs was chosen to keep a factor of $\sqrt{2}$ between consecutive DOFs while allowing for (approximately) integer numbers of elements across the range of polynomial degrees considered. The configurations marked by a box are the ones that are stable with over-integration, whereas the other configuration could not be successfully finished with over-integration alone [32].

To further assess the behaviour of the schemes for the inviscid Taylor-Green vortex, we investigate the evolution of the total kinetic energy dissipation rate, $-d\kappa/dt$, and the enstrophy

$$\sigma := \frac{1}{|\Omega|} \int_\Omega \frac{\rho}{2} \omega \cdot \omega \, d\Omega, \tag{4.5}$$

where ω is the vorticity vector. In the left part of Fig. 6, the resulting dissipation rate for the configuration $N = 3$ with 16^3 grid cells (64^3 degrees of freedom) for all schemes is plotted. Although there is a difference between the split form variants, it is fairly small. In the right part of Fig. 6 we focus on the KG variant only and increase the number of degrees of freedom to compare the 64^3 results to the configuration with 32^3 grid cells. It is interesting to note that again, the dissipation rate does not change significantly.

It seems that the dissipation introduced by the numerical surface flux function has a certain limit on the dissipation of the integrated kinetic energy. This is further supported by the results in Fig. 7, where for configurations with fixed 64^3 degrees of freedom the polynomial degrees are increased to $N = 7$ and $N = 15$. Note, that the dissipation rates do not significantly change.

In contrast to the behaviour of the dissipation rates, the evolution of the enstrophy strongly depends on the spatial resolution and on the polynomial degree. Figure 8 shows the enstrophy as a function of time for a fixed 64^3 degrees of freedom in the left part. The difference between the split forms is again small, however the impact of the polynomial degree increases the magnitude of the total of enstrophy by an order of magnitude. The right part of Fig. 8 considers again the KG scheme with $N = 3$ and 16^3 and 32^3 grid cells, respectively. By comparing the left and right plot, it is interesting to note that the enstrophy of the configuration with $N = 7$ and 8^3 grid cells (64^3 degrees of freedom) has a higher maximum than the configuration with $N = 3$ and 32^3 grid cells (128^3 degrees of freedom).

Figure 6: Plot of the total kinetic energy dissipation rate. Left: $N = 3$ with 64^3 degrees of freedom for all stable split forms. Right: KG scheme with $N = 3$ and 16^3 and 32^3 grid cells respectively.

Figure 7: Plot of the total kinetic energy dissipation rate. Left: $N = 7$ with 8^3 grid cells for all stable split forms. Right: $N = 15$ with 4^3 grid cells for all stable split forms.

Figure 8: Temporal evolution of the enstrophy. Left: Fixed 64^3 degrees of freedom for all stable split forms for configurations with polynomial degree $N = 3, 7, 15$. Right: Enstrophy of the KG scheme with $N = 3$ and 16^3 and 32^3 grid cells respectively.

The significant impact of the polynomial degree on the evolution of the enstrophy has a direct consequence for the estimated numerical viscosity of the discretisations. For solutions of the incompressible viscous Taylor-Green vortex, the dissipation rate is directly linked to the enstrophy via the physical viscosity μ [15]

$$-\frac{d\,\kappa}{dt} = 2\,\mu\,\sigma. \tag{4.6}$$

We exploit this relationship to find an estimate for the viscosity introduced by the numerical discretisation. By relating the discrete dissipation rate and the enstrophy over time, we get an evolution of the numerical dissipation of the scheme

$$\mu_{num} \approx \frac{-\frac{d\,\kappa}{dt}}{2\,\sigma}. \tag{4.7}$$

The numerical viscosity estimate is plotted in the left part of Fig. 9, where again the degrees of freedom are fixed to 64^3. Due to the significantly higher enstrophy for the high polynomial degree discretisations, the estimate of the numerical viscosity is much lower. This fits to the apparent "high-order schemes have lower dissipation" paradigm. However, it is interesting to note that the actual dissipation rate of the kinetic energy does not vary much. In the right part of Fig. 9, we again compare the KG scheme for the $N = 3$ and 16^3 configuration. We directly compare the impact of the interface stabilisation terms on the numerical viscosity and plot results from the simulation with interface stabilisation terms and without. It can be clearly observed that the no stabilisation run has almost zero numerical viscosity, while being stable for this severely under-resolved test case.

At the end of this section, we note that the variant KG (and PI) with cubic split forms was introduced by Kennedy and Gruber [22] to account for large density variations. Thus, it is interesting to investigate the difference of DU and KG/PI for a test configuration with higher compressibility. To do so we change the initial pressure for inviscid the Taylor-Green vortex so that the Mach number increases to $Ma = 0.4$. This introduces more compressibility effects and as shown in Fig. 10 and 11, this causes problems for the DU variant. For the configuration $N = 7$ with 16^3 grid cells, we can observe that DU crashes at $t \approx 10.8$, even with very small CFL numbers. The variants KG, PI and IR, CH all run until the final time, demonstrating increased robustness of those variants for problems with higher density variations.

23

Figure 9: Plot of the numerical viscosity over time. Left: Fixed 64^3 degrees of freedom for all stable split forms for configurations with polynomial degree $N = 3, 7, 15$. Right: Comparison of KG results with and without stabilisation for $N = 3$ and 16^3 grid cells.

Figure 10: Results of simulation with 16^3 grid cells and $N = 7$ (128^3 degrees of freedom) for the Taylor-Green vortex with a Mach number of $Ma = 0.4$. Left: Plot of total enstrophy. Right: Plot of total kinetic energy. Note that DU crashes at $t \approx 10.8$, indicated by an extreme spike of the quantities.

Figure 11: Results of simulation with 16^3 grid cells and $N = 7$ (128^3 degrees of freedom) for the Taylor-Green vortex with a Mach number of $Ma = 0.4$. Left: Plot of dissipation rate of total kinetic energy. Right: Plot of Numerical viscosity estimation. Note that DU crashes at $t \approx 10.8$, indicated by an extreme spike of the quantities.

Summarising the investigations of this section, the most important result is that all split form schemes, except the variant MO, significantly improve the robustness of high-order DG discretisations compared to polynomial de-aliasing with over-integration. We could demonstrate that the cubic split forms are more robust in comparison to the quadratic splitting when the flow is compressible. Furthermore, the influence of the numerical viscosity can be directly observed and tracked by comparison to nearly dissipation free variants of the scheme. This seems to be a beneficial and interesting feature for future investigations on turbulence modelling.

5. Final Remarks

5.1. Discussion

As a final discussion, the identities in Lem. 1 form the basis for a dictionary of a translation from split forms into flux forms and vice versa. We used these to translate known alternative formulations of the non-linear compressible Euler terms. However, this dictionary can be used to generate a multitude of new split forms by choosing symmetric and consistent flux approximations. To demonstrate this, we consider an example based on the Roe variables

$$
\begin{aligned}
q_1 &= \sqrt{\rho}, \\
q_2 &= \sqrt{\rho}\, u, \\
q_3 &= \sqrt{\rho}\, v, \\
q_4 &= \sqrt{\rho}\, w, \\
q_5 &= \sqrt{\rho}\, h.
\end{aligned}
\tag{5.1}
$$

25

This specific set of variables is interesting, as it reformulates the flux into only quadratic products, e.g. the flux in the $x-$direction is

$$F(U) = \begin{pmatrix} q_1\, q_2 \\ q_2^2 + \frac{\gamma-1}{\gamma}\left(q_1\, q_5 - \frac{1}{2}\left(q_2^2 + q_3^2 + q_4^2\right)\right) \\ q_2\, q_3 \\ q_2\, q_4 \\ q_2\, q_5 \end{pmatrix}. \tag{5.2}$$

Assuming now a quadratic splitting of the products, we can directly translate this into an equivalent numerical volume flux QU, based on the quadratic identities of Lem. 1

$$F_{QU}^{\#}(U_{ijk}, U_{mjk}) = \begin{pmatrix} \{\{q_1\}\}\,\{\{q_2\}\} \\ \{\{q_2\}\}^2 + \frac{\gamma-1}{\gamma}\left(\{\{q_1\}\}\,\{\{q_5\}\} - \frac{1}{2}\left(\{\{q_2\}\}^2 + \{\{q_3\}\}^2 + \{\{q_4\}\}^2\right)\right) \\ \{\{q_2\}\}\,\{\{q_3\}\} \\ \{\{q_2\}\}\,\{\{q_4\}\} \\ \{\{q_2\}\}\,\{\{q_5\}\} \end{pmatrix}, \tag{5.3}$$

and thus, generate a new scheme. In Fig. 12 we provide a comparison of the dissipation rates of the new scheme against the previous results. With this simple construction and translation guide, it is straightforward

Figure 12: Plot of dissipation rates for all variants including the Roe variable based splitting variant QU for $N = 3$ and 16^3 grid cells.

to introduce novel splittings, as demonstrated, and perhaps find forms that offer benefits over the ones currently available in the literature.

5.2. Conclusion

In the first part of this paper, we exploit the volume flux difference form introduced by Carpenter et al. and Fisher et al. [2, 12] to construct a unified formulation for split forms of quadratic and cubic products. From a simple relationship between split forms for quadratic and cubic products, we showed that it is possible to directly translate known alternative compressible Euler formulations into the volume flux difference form (3.3) by introducing a numerical volume flux $F^{\#}$, $G^{\#}$, $H^{\#}$. It is possible to simply change the volume flux

and generate a new DGSEM. We prove that the kinetic energy preserving numerical volume fluxes generate high-order accurate discretisations that also preserve kinetic energy.

Several alternative formulations of the compressible Euler equations from the literature are investigated in the numerical results sections. All schemes show expected convergence behaviour and expected auxiliary conservation properties. An outcome of the numerical assessment is that the new DG schemes are more robust than DG with over-integration. Configurations that crash with over-integration can be successfully finished with the split form DGSEM as well as with the entropy stable DGSEM introduced by Carpenter et al. [2]. The numerical volume fluxes of the KG, PI and DU variants offer a computational advantage compared to the more complex entropy stable variants IR and CH. We note that the volume flux differencing form is computationally more intense than the standard DGSEM. On the other hand, we found that for the inviscid Taylor-Green vortex, all configurations for the standard DGSEM (without over-integration) immediately crash.

It is also worth noting that not all alternative formulations automatically improve the robustness, even if the split form is kinetic energy preserving, such as the MO variant. Similar to the standard DGSEM, this variant crashes for all configurations after a short simulation time. Whereas the DU variant is as stable as the KG, PI, IR and CH variants for low Mach numbers, for higher Mach numbers (more compressibility effects), DU is less robust. A conclusion so far is that a good balance between robustness and computational effort is offered by the variants KG and PI. However, it is clear that with the newly established framework for de-aliasing split forms of DG for the compressible Euler equations [10, 22, 28, 29] more assessment of the many interesting aspects, such as robustness, accuracy and efficiency is necessary.

Additionally, the numerical viscosity of the different discretisations is investigated and it was possible to directly trace the influence of the stabilisation terms. The key for this is that even without interface stabilisation terms, some configurations of the schemes were stable. This offers interesting possibilities in the control and design of dissipation tailored for turbulence modelling.

Appendix A. Proofs

Appendix A.1. Proof of Lem. 1 [Discrete split forms]

Proof. First we consider and make use of the relation $\mathbf{D} = \mathbf{M}^{-1}\mathbf{Q}$ to get

$$
\begin{aligned}
2\sum_{m=0}^{N} Q_{im}\,\{\!\{a\}\!\}_{im} &= \sum_{m=0}^{N} Q_{im}(a_i + a_m), \\
&= a_i\sum_{m=0}^{N} Q_{im} + \sum_{m=0}^{N} Q_{im}a_m, \\
&= \sum_{m=0}^{N} Q_{im}a_m,
\end{aligned}
\tag{A.1}
$$

for $i = 0, \ldots, N$ and we have used the fact that the rows of $\mathbf{Q}$ sum to zero [37]. We compress the result (A.1) into the matrix vector product form

$$
2\sum_{m=0}^{N} D_{im}\,\{\!\{a\}\!\}_{im} = (\mathbf{D}\,\underline{a})_i.
\tag{A.2}
$$

27

Next, we prove the identity for the product of two averages

$$2 \sum_{m=0}^{N} Q_{im} \, \{\!\{a\}\!\}_{im} \, \{\!\{b\}\!\}_{im} = \frac{1}{2} \sum_{m=0}^{N} Q_{im}(a_i + a_m)(b_i + b_m),$$

$$= \frac{1}{2} \sum_{m=0}^{N} Q_{im} a_m \, b_m + \frac{a_i}{2} \sum_{m=0}^{N} Q_{im} \, b_m + \frac{b_i}{2} \sum_{m=0}^{N} Q_{im} \, a_m + \frac{a_i \, b_i}{2} \sum_{m=0}^{N} Q_{im}, \quad \text{(A.3)}$$

$$= \frac{1}{2} \left\{ \sum_{m=0}^{N} Q_{im} \, a_m \, b_m + a_i \sum_{m=0}^{N} Q_{im} \, b_m + b_i \sum_{m=0}^{N} Q_{im} \, a_m \right\},$$

for $i = 0, \ldots, N$. Again, we have used the property that the sum of each row of the SBP matrix $\mathbf{Q}$ vanishes. We see that after scaling by i-th term of $\mathbf{M}^{-1}$ the flux difference result (A.3) is the ith row of the split form

$$2 \sum_{m=0}^{N} D_{im} \, \{\!\{a\}\!\}_{im} \, \{\!\{b\}\!\}_{im} = \frac{1}{2} \left(\mathbf{D} \, \underline{a} \, \underline{b} + \underline{\underline{a}} \, \mathbf{D} \, \underline{b} + \underline{\underline{b}} \, \mathbf{D} \, \underline{a} \right)_i. \quad \text{(A.4)}$$

We prove the final identity of (3.5) by considering a triple product of averages

$$2 \sum_{m=0}^{N} Q_{im} \, \{\!\{a\}\!\}_{im} \, \{\!\{b\}\!\}_{im} \, \{\!\{c\}\!\}_{im} = \frac{1}{4} \sum_{m=0}^{N} Q_{im}(a_i + a_m)(b_i + b_m)(c_i + c_m),$$

$$= \frac{1}{4} \sum_{m=0}^{N} Q_{im} \, (a_i b_i c_i + a_i b_i c_m + a_i b_m c_i + a_i b_m c_m$$

$$+ a_m \, b_i \, c_i + a_m \, b_i \, c_m + a_m \, b_m \, c_i + a_m \, b_m \, c_m),$$

$$= \frac{a_i \, b_i \, c_i}{4} \sum_{m=0}^{N} Q_{im} + \frac{a_i \, b_i}{4} \sum_{m=0}^{N} Q_{im} \, c_m + \frac{a_i \, c_i}{4} \sum_{m=0}^{N} Q_{im} \, b_m + \frac{a_i}{4} \sum_{m=0}^{N} Q_{im} \, b_m \, c_m \quad \text{(A.5)}$$

$$+ \frac{b_i \, c_i}{4} \sum_{m=0}^{N} Q_{im} \, a_m + \frac{b_i}{4} \sum_{m=0}^{N} Q_{im} \, a_m \, c_m + \frac{c_i}{4} \sum_{m=0}^{N} Q_{im} \, a_m \, b_m + \sum_{m=0}^{N} Q_{im} \, a_m \, b_m \, c_m,$$

$$= \frac{1}{4} \left\{ \sum_{m=0}^{N} Q_{im} \, a_m \, b_m \, c_m + a_i \sum_{m=0}^{N} Q_{im} \, b_m \, c_m + b_i \sum_{m=0}^{N} Q_{im} \, a_m \, c_m + c_i \sum_{m=0}^{N} Q_{im} \, a_m \, b_m \right.$$

$$\left. + b_i \, c_i \sum_{m=0}^{N} Q_{im} \, a_m + a_i \, c_i \sum_{m=0}^{N} Q_{im} \, b_m + a_i \, b_i \sum_{m=0}^{N} Q_{im} \, c_m \right\},$$

for $i = 0, \ldots, N$. The vanishing sum property of the rows of $\mathbf{Q}$ is, again, applied. We condense the final result of (A.5) into matrix-vector form to obtain the final split form identity

$$2 \sum_{m=0}^{N} D_{im} \, \{\!\{a\}\!\}_{im} \, \{\!\{b\}\!\}_{im} \, \{\!\{c\}\!\}_{im} = \frac{1}{4} \left(\mathbf{D} \, \underline{a} \, \underline{b} \, \underline{c} + \underline{\underline{a}} \, \mathbf{D} \, \underline{b} \, \underline{c} + \underline{\underline{b}} \, \mathbf{D} \, \underline{a} \, \underline{c} + \underline{\underline{c}} \, \mathbf{D} \, \underline{a} \, \underline{b} + \underline{\underline{bc}} \, \mathbf{D} \, \underline{a} + \underline{\underline{ac}} \, \mathbf{D} \, \underline{b} + \underline{\underline{ab}} \, \mathbf{D} \, \underline{c} \right)_i.$$

$$\text{(A.6)}$$

$\blacksquare$

Appendix A.2. Proof of Thm. 2 [Kinetic energy preservation]

Proof. The kinetic energy is calculated as

$$\kappa := \frac{1}{2} \rho \left(u^2 + v^2 + w^2 \right) = \frac{(\rho \, u)^2 + (\rho \, v)^2 + (\rho \, w)^2}{2 \, \rho}. \quad \text{(A.7)}$$

The kinetic energy balance for the compressible Euler equations is

$$\frac{\partial \kappa}{\partial t} + \left(\frac{1}{2} \rho \, u \left(u^2 + v^2 + w^2 \right) \right)_x + u \, p_x + \left(\frac{1}{2} \rho \, v \left(u^2 + v^2 + w^2 \right) \right)_y + v \, p_y + \left(\frac{1}{2} \rho \, w \left(u^2 + v^2 + w^2 \right) \right)_z + w \, p_z = 0. \quad \text{(A.8)}$$

We can also express the time derivative of the kinetic energy in terms of the time derivatives of the mass and momentum

$$\frac{\partial \kappa}{\partial t} := \frac{\partial k}{\partial \rho} (\rho)_t + \frac{\partial k}{\partial (\rho \, u)} (\rho \, u)_t + \frac{\partial k}{\partial (\rho \, v)} (\rho \, v)_t + \frac{\partial k}{\partial (\rho \, w)} (\rho \, w)_t,$$
$$= -\left(\frac{u^2 + v^2 + w^2}{2} \right) (\rho)_t + (u) \, (\rho \, u)_t + (v) \, (\rho \, v)_t + (w) \, (\rho \, w)_t. \tag{A.9}$$

We focus on the volume parts of the continuity equation and consider a single node ijk to get

$$-\left(\frac{u^2 + v^2 + w^2}{2} \right) (\rho)_t \approx -\left(\frac{(u_{ijk})^2 + (v_{ijk})^2 + (w_{ijk})^2}{2} \right) 2 \sum_{m=0}^{N} D_{im} \, F^{\#,1}(U_{ijk}, U_{mjk}) + D_{jm} \, G^{\#,1}(U_{ijk}, U_{imk}) + D_{km} \, H^{\#,1}(U_{ijk}, U_{ijm}), \tag{A.10}$$

and

$$(u) \, (\rho \, u)_t \approx u_{ijk} \, 2 \sum_{m=0}^{N} D_{im} \, F^{\#,1}(U_{ijk}, U_{mjk}) \, \{\!\{u\}\!\}_{(i,m)jk} + D_{jm} \, G^{\#,1}(U_{ijk}, U_{imk}) \, \{\!\{u\}\!\}_{i(j,m)k} + D_{km} \, H^{\#,1}(U_{ijk}, U_{ijm}) \, \{\!\{u\}\!\}_{ij(k,m)}$$
$$+ u_{ijk} \, 2 \sum_{m=0}^{N} D_{im} \widetilde{p}(U_{ijk}, U_{mjk}),$$

$$(v) \, (\rho \, v)_t \approx v_{ijk} \, 2 \sum_{m=0}^{N} D_{im} \, F^{\#,1}(U_{ijk}, U_{mjk}) \, \{\!\{v\}\!\}_{(i,m)jk} + D_{jm} \, G^{\#,1}(U_{ijk}, U_{imk}) \, \{\!\{v\}\!\}_{i(j,m)k} + D_{km} \, H^{\#,1}(U_{ijk}, U_{ijm}) \, \{\!\{v\}\!\}_{ij(k,m)} \tag{A.11}$$
$$+ v_{ijk} \, 2 \sum_{m=0}^{N} D_{jm} \widetilde{p}(U_{ijk}, U_{imk}),$$

$$(w) \, (\rho \, w)_t \approx w_{ijk} \, 2 \sum_{m=0}^{N} D_{im} \, F^{\#,1}(U_{ijk}, U_{mjk}) \, \{\!\{w\}\!\}_{(i,m)jk} + D_{jm} \, G^{\#,1}(U_{ijk}, U_{imk}) \, \{\!\{w\}\!\}_{i(j,m)k} + D_{km} \, H^{\#,1}(U_{ijk}, U_{ijm}) \, \{\!\{w\}\!\}_{ij(k,m)}$$
$$+ w_{ijk} \, 2 \sum_{m=0}^{N} D_{km} \widetilde{p}(U_{ijk}, U_{ijm}),$$

where we used a notation of the average of two states, e.g. $\{\!\{u\}\!\}_{(i,m)jk} := \frac{1}{2}(u_{ijk} + u_{mjk})$ and inserted the kinetic energy preserving structure of the numerical flux functions, see (3.23).

The next step is to sum all four expressions according to (A.9). We first note that as long as the numerical trace approximation of the pressure $\widetilde{p}$ is consistent and symmetric, it follows from the work of Fisher et al. and Carpenter et al. [2, 12] that the resulting discretisation of $u \, p_x + v \, p_y + w \, p_z$ of the pressure work is consistent and high-order accurate.

If we look at the advective terms, we note that the values u_{ijk}, v_{ijk}, and w_{ijk} do not depend on the summation index m and thus can be pulled into the sum. By doing this, we inspect the product of the velocities and the averages. Looking at the terms involving $F^{\#,1}(U_{ijk}, U_{mjk})$,

$$u_{ijk} \, \{\!\{u\}\!\}_{(i,m)jk} \, F^{\#,1}(U_{ijk}, U_{mjk}) = u_{ijk} \frac{1}{2}(u_{ijk} + u_{mjk}) \, F^{\#,1}(U_{ijk}, U_{mjk}) = \left[\frac{(u_{ijk})^2}{2} + \frac{u_{ijk} \, u_{mjk}}{2} \right] F^{\#,1}(U_{ijk}, U_{mjk}),$$

$$v_{ijk} \, \{\!\{v\}\!\}_{(i,m)jk} \, F^{\#,1}(U_{ijk}, U_{mjk}) = v_{ijk} \frac{1}{2}(v_{ijk} + v_{mjk}) \, F^{\#,1}(U_{ijk}, U_{mjk}) = \left[\frac{(v_{ijk})^2}{2} + \frac{v_{ijk} \, v_{mjk}}{2} \right] F^{\#,1}(U_{ijk}, U_{mjk}), \tag{A.12}$$

$$w_{ijk} \, \{\!\{w\}\!\}_{(i,m)jk} \, F^{\#,1}(U_{ijk}, U_{mjk}) = w_{ijk} \frac{1}{2}(w_{ijk} + w_{mjk}) \, F^{\#,1}(U_{ijk}, U_{mjk}) = \left[\frac{(w_{ijk})^2}{2} + \frac{w_{ijk} \, w_{mjk}}{2} \right] F^{\#,1}(U_{ijk}, U_{mjk}).$$

Summing these three terms gives

$$\left(u_{ijk} \, \{\!\{u\}\!\}_{(i,m)jk} + v_{ijk} \, \{\!\{v\}\!\}_{(i,m)jk} + w_{ijk} \, \{\!\{w\}\!\}_{(i,m)jk} \right) F^{\#,1}(U_{ijk}, U_{mjk}) = \frac{(u_{ijk})^2 + (v_{ijk})^2 + (w_{ijk})^2}{2} F^{\#,1}(U_{ijk}, U_{mjk})$$
$$+ \frac{u_{ijk} \, u_{mjk} + v_{ijk} \, v_{mjk} + w_{ijk} \, w_{mjk}}{2} F^{\#,1}(U_{ijk}, U_{mjk}). \tag{A.13}$$

Note, that the first term is identical to the first term on the right hand side of (A.10), except for the sign. Thus, when adding, this term cancels and we are left with the remainder term

$$f^{\#,\kappa}(U_{ijk}, U_{mjk}) := \frac{1}{2} \, F^{\#,1}(U_{ijk}, U_{mjk}) \, (u_{ijk} \, u_{mjk} + v_{ijk} \, v_{mjk} + w_{ijk} \, w_{mjk}), \tag{A.14}$$

29

which is symmetric in the arguments and is consistent with the advective flux of the kinetic energy balance $f = \frac{1}{2} \rho \, u \, (u^2 + v^2 + w^2)$.

Analogously, if we focus on the terms involving $G^{\#,1}(U_{ijk}, U_{imk})$, we get

$$g^{\#,\kappa}(U_{ijk}, U_{imk}) := \frac{1}{2} \, G^{\#,1}(U_{ijk}, U_{imk}) \, (u_{ijk} \, u_{imk} + v_{ijk} \, v_{imk} + w_{ijk} \, w_{imk}), \tag{A.15}$$

which is again symmetric in the arguments and consistent to $g = \frac{1}{2} \rho \, v \, (u^2 + v^2 + w^2)$. For the terms involving $H^{\#,1}(U_{ijk}, U_{ijm})$ we get

$$h^{\#,\kappa}(U_{ijk}, U_{ijm}) := \frac{1}{2} \, H^{\#,1}(U_{ijk}, U_{ijm}) \, (u_{ijk} \, u_{ijm} + v_{ijk} \, v_{ijm} + w_{ijk} \, w_{ijm}), \tag{A.16}$$

which is also symmetric and consistent to $h = \frac{1}{2} \rho \, w \, (u^2 + v^2 + w^2)$.

Thus, summarising, we get the following volume term in our discrete kinetic energy balance

$$\left(\frac{\partial \kappa}{\partial t}\right)_{ijk} \approx -2 \sum_{m=0}^{N} D_{im} \, f^{\#,\kappa}(U_{ijk}, U_{mjk}) + D_{jm} \, g^{\#,\kappa}(U_{ijk}, U_{imk}) + D_{km} \, h^{\#,\kappa}(U_{ijk}, U_{ijm})$$
$$-2 \sum_{m=0}^{N} u_{ijk} \, D_{im} \, \widetilde{p}(U_{ijk}, U_{mjk}) + v_{ijk} \, D_{jm} \, \widetilde{p}(U_{ijk}, U_{imk}) + w_{ijk} \, D_{km} \, \widetilde{p}(U_{ijk}, U_{ijm}), \tag{A.17}$$

which is a consistent and high-order accurate approximation of the continuous kinetic energy balance (A.8), with the advective terms in conservative form. ∎

Appendix B. Curvilinear flux differencing form

Now the computational domain is divided into non-overlapping *curved* hexahedral elements C. We create a polynomial transformation $\underline{X}(\underline{\xi})$ to map computational coordinates in the reference cube $\underline{\xi} = (\xi^1, \xi^2, \xi^3) = (\xi, \eta, \zeta)$ to physical coordinates $\underline{x} = (x, y, z) \in C$, for details see [25].

The curvilinear coordinate system for $\underline{\xi}$ on the reference cube has three covariant basis vectors, $\underline{a}_i$, computed directly from the transformation

$$\underline{a}_i = \frac{\partial \underline{X}}{\partial \xi^i}, \quad i = 1, 2, 3. \tag{B.1}$$

From the covariant basis vectors we derive the contravariant basis vectors $\underline{a}^i$, scaled by the Jacobian of the transformation J

$$J\underline{a}^i = J\nabla\xi^i = \underline{a}_j \times \underline{a}_k, \quad (i, j, k) \text{ cyclic.} \tag{B.2}$$

Alternatively, there is an explicitly divergence-free form of the contravariant basis vectors derived in [24]

$$Ja_n^i = -\hat{x}_i \cdot \nabla_\xi \times (x_l \nabla_\xi x_m), \quad i = 1, 2, 3; \; n = 1, 2, 3; \; (n, m, l) \text{ cyclic.} \tag{B.3}$$

The divergence-free form (B.3) of the contravariant basis vectors is particularly important to prevent spurious oscillations in the solution on curved sided hexahedral elements [24]. However, for two dimensional problems and straight-sided hexahedral meshes (e.g. Cartesian meshes) the cross product formulation (B.2) is sufficient to prevent the generation of spurious waves by a mesh [24].

The conservation law in the physical domain transforms to a conservation law equation in the reference domain of the form

$$JU_t + \widetilde{\mathcal{L}}_\xi^{div}(U) + \widetilde{\mathcal{L}}_\eta^{div}(U) + \widetilde{\mathcal{L}}_\zeta^{div}(U) = 0, \tag{B.4}$$

where the Jacobian values of the transformation are computed from the covariant basis vectors (B.1) and the contravariant operators incorporate the metric terms (B.2)

$$
\begin{aligned}
J &= \underline{a}_1 \cdot (\underline{a}_2 \times \underline{a}_3), \\
\widetilde{\mathcal{L}}_\xi^{div}(U) &= \left(Ja_1^1\, F(U) + Ja_2^1\, G(U) + Ja_3^1\, H(U) \right)_\xi, \\
\widetilde{\mathcal{L}}_\eta^{div}(U) &= \left(Ja_1^2\, F(U) + Ja_2^2\, G(U) + Ja_3^2\, H(U) \right)_\eta, \\
\widetilde{\mathcal{L}}_\zeta^{div}(U) &= \left(Ja_1^3\, F(U) + Ja_2^3\, G(U) + Ja_3^3\, H(U) \right)_\zeta.
\end{aligned}
\tag{B.5}
$$

Next, consider the component l of the mapped system (B.4) and an GL node (i,j,k), the DGSEM approximation in strong form is

$$
\left(\widetilde{\mathcal{L}}_\xi^{div}(U) \right)_{ijk}^l \approx \left[\widetilde{F}^{*,l}(1, \eta_j, \zeta_k; \underline{n}) - \widetilde{F}_{Njk}^l \right] - \left[\widetilde{F}^{*,l}(-1, \eta_j, \zeta_k; \underline{n}) - \widetilde{F}_{0jk}^l \right] + \sum_{m=0}^N D_{im} \widetilde{F}_{mjk}^l,
\tag{B.6}
$$

$$
\widetilde{F}_{ijk}^l \approx Ja_1^1\, F_{ijk}^l + Ja_2^1\, G_{ijk}^l + Ja_3^1\, H_{ijk}^l,
$$

where we use collocation for the non-linear flux functions, e.g. $F_{ijk}^l := F^l(U_{ijk})$ and denote the outward pointing normal vector by $\underline{n}$. Note that the terms $\left(\widetilde{\mathcal{L}}_\eta^{div}(U) \right)_{ijk}^l$ and $\left(\widetilde{\mathcal{L}}_\zeta^{div}(U) \right)_{ijk}^l$ have an analogous structure to (B.6).

Appendix B.1. Split form stabilisation for curvilinear DGSEM

Just as in Sec. 3.1, we consider alternative formulations for the volume part of the discretisation (B.6). Again, this means starting from a standard strong form DGSEM implementation and modifying the volume integrals. We note that for curvilinear meshes the metric terms are now polynomials. Thus, the non-linearity in the contravariant terms increases. The increased non-linearity due to the curvilinear grid is an additional cause for aliasing driven non-linear instabilities. For a detailed algorithmic description of the curvilinear flux difference formulation see [39].

In his PhD thesis, Fisher [11] extended the de-aliasing flux differencing technique (3.2) to curvilinear grids with corresponding element-wise mappings. For this extension, the contravariant operators in each direction are needed. We collect the main result below and proofs can be found in [11]. In general curvilinear coordinates the high-order flux difference form has the structure

$$
\frac{\overline{\widetilde{F^l}}_{(i+1)jk} - \overline{\widetilde{F^l}}_{(i)jk}}{\omega_i} \approx 2 \sum_{m=0}^N D_{im} \Big[F^{\#,l}(U_{ijk}, U_{mjk}) \left\{\!\left\{ Ja_1^1 \right\}\!\right\}_{(i,m)jk}
$$
$$
+ G^{\#,l}(U_{ijk}, U_{mjk}) \left\{\!\left\{ Ja_2^1 \right\}\!\right\}_{(i,m)jk}
\tag{B.7}
$$
$$
+ H^{\#,l}(U_{ijk}, U_{mjk}) \left\{\!\left\{ Ja_3^1 \right\}\!\right\}_{(i,m)jk} \Big],
$$

where

$$
\left\{\!\left\{ Ja_r^1 \right\}\!\right\}_{(i,m)jk} = \frac{1}{2} \left[\left(Ja_r^1 \right)_{ijk} + \left(Ja_r^1 \right)_{mjk} \right], \quad r = 1, 2, 3
\tag{B.8}
$$

are the average of the metric term components. For the approximation of the $\overline{\widetilde{G^l}}$ and $\overline{\widetilde{H^l}}$ contravariant flux differences the superscript on the metric term averages in (B.8) change to 2 and 3 respectively. Each of the volume flux functions in (B.7) are consistent and symmetric. Also, the resulting curvilinear approximation is high-order accurate and conservative in the Lax-Wendroff sense. We use the form of the volume discretisation

(B.7) rather than the standard discretisation (B.6)

$$
\left(\widetilde{\mathcal{L}}^{div}_{\xi}(U)\right)^l_{ijk} \approx \left[\widetilde{F}^{*,l}(1,\eta_j,\zeta_k;\underline{n}) - \tilde{F}^l_{Njk}\right] - \left[\widetilde{F}^{*,l}(-1,\eta_j,\zeta_k;\underline{n}) - \widetilde{F}^l_{0jk}\right] + 2\sum_{m=0}^{N} D_{im}\left[F^{\#,l}(U_{ijk},U_{mjk})\left\{\!\left\{Ja^1_1\right\}\!\right\}_{(i,m)jk}\right.
$$
$$
+ G^{\#,l}(U_{ijk},U_{mjk})\left\{\!\left\{Ja^1_2\right\}\!\right\}_{(i,m)jk}
$$
$$
\left. + H^{\#,l}(U_{ijk},U_{mjk})\left\{\!\left\{Ja^1_3\right\}\!\right\}_{(i,m)jk}\right],
$$

$$
\left(\widetilde{\mathcal{L}}^{div}_{\eta}(U)\right)^l_{ijk} \approx \left[\widetilde{G}^{*,l}(\xi_i,1,\zeta_k;\underline{n}) - \tilde{G}^l_{iNk}\right] - \left[\widetilde{G}^{*,l}(\xi_i,-1,\zeta_k;\underline{n}) - \widetilde{G}^l_{i0k}\right] + 2\sum_{m=0}^{N} D_{jm}\left[F^{\#,l}(U_{ijk},U_{imk})\left\{\!\left\{Ja^2_1\right\}\!\right\}_{i(j,m)k}\right.
$$
$$
+ G^{\#,l}(U_{ijk},U_{imk})\left\{\!\left\{Ja^2_2\right\}\!\right\}_{i(j,m)k}
$$
$$
\left. + H^{\#,l}(U_{ijk},U_{imk})\left\{\!\left\{Ja^2_3\right\}\!\right\}_{i(j,m)k}\right],
$$

$$
\left(\widetilde{\mathcal{L}}^{div}_{\zeta}(U)\right)^l_{ijk} \approx \left[\widetilde{H}^{*,l}(\xi_i,\eta_j,1;\underline{n}) - \tilde{H}^l_{ijN}\right] - \left[\widetilde{H}^{*,l}(\xi_i,\eta_j,-1;\underline{n}) - \widetilde{H}^l_{ij0}\right] + 2\sum_{m=0}^{N} D_{km}\left[F^{\#,l}(U_{ijk},U_{ijm})\left\{\!\left\{Ja^3_1\right\}\!\right\}_{ij(k,m)}\right.
$$
$$
+ G^{\#,l}(U_{ijk},U_{ijm})\left\{\!\left\{Ja^3_2\right\}\!\right\}_{ij(k,m)}
$$
$$
\left. + H^{\#,l}(U_{ijk},U_{ijm})\left\{\!\left\{Ja^3_3\right\}\!\right\}_{ij(k,m)}\right],
$$
$$(B.9)$$

where the choice of the numerical surface fluxes in the normal direction, $\underline{n}$, for each of the # variants of DGSEM are identical to those discussed in Sec. 3.4.

References

References

[1] G. A. Blaisdell, E. T. Spyropoulos, and J. H. Qin. The effect of the formulation of nonlinear terms on aliasing errors in spectral methods. *Applied Numerical Mathematics*, 21(3):207–219, 1996.

[2] M. Carpenter, T. Fisher, E. Nielsen, and S. Frankel. Entropy stable spectral collocation schemes for the Navier–Stokes equations: Discontinuous interfaces. *SIAM Journal on Scientific Computing*, 36(5):B835–B867, 2014.

[3] Mark H. Carpenter and Christopher A. Kennedy. Fourth-order 2N-storage Runge- Kutta schemes. Technical Report NASA TM 109111, NASA Langley Research Center, 1994.

[4] Praveen Chandrashekar. Kinetic energy preserving and entropy stable finite volume schemes for compressible Euler and Navier-Stokes equations. *Communications in Computational Physics*, 14(5):1252–1286, 2013.

[5] B. Cockburn, S. Hou, and C. W. Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case. *Math. Comput.*, 54:545–581, 1990.

[6] B. Cockburn, S. Y. Lin, and C.W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems. *J. Comput. Phys.*, 84:90–113, 1989.

[7] B. Cockburn and C. W. Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework. *Math. Comput.*, 52:411–435, 1989.

[8] B. Cockburn and C. W. Shu. The Runge-Kutta discontinuous Galerkin method for conservation laws V: Multidimensional systems. *J. Comput. Phys.*, 141:199–224, 1998.

[9] B. Cockburn and C.W. Shu. The Runge-Kutta local projection p^1-discontinuous Galerkin method for scalar conservation laws. M^2AN, 25:337–361, 1991.

[10] F. Ducros, F. Laporte, T. Soulères, V. Guinot, P. Moinat, and B. Caruelle. High-order fluxes for conservative skew-symmetric-like schemes in structured meshes: Application to compressible flows. *Journal of Computational Physics*, 161:114–139, 2000.

[11] Travis C. Fisher. *High-order L_2 stable multi-domain finite difference method for compressible flows*. PhD thesis, Purdue University, 2012.

[12] Travis C. Fisher and Mark H. Carpenter. High-order entropy stable finite difference schemes for nonlinear conservation laws: Finite domains. *Journal of Computational Physics*, 252:518–557, 2013.

[13] Gregor J. Gassner. A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods. *SIAM Journal on Scientific Computing*, 35(3):A1233–A1253, 2013.

[14] Gregor J. Gassner. A kinetic energy preserving nodal discontinuous Galerkin spectral element method. *International Journal for Numerical Methods in Fluids*, 76(1):28–50, 2014.

[15] Gregor J. Gassner and Andrea D. Beck. On the accuracy of high-order discretizations for underresolved turbulence simulations. *Theoretical and Computational Fluid Dynamics*, 2012. DOI: 10.1007/s00162-011-0253-7.

[16] Gregor J. Gassner, Andrew R. Winters, and David A. Kopriva. A well balanced and entropy conservative discontinuous Galerkin spectral element method for the shallow water equations. *Applied Mathematics and Computation*, page http://dx.doi.org/10.1016/j.amc.2015.07.014, 2015.

[17] Andreas Hiltebrand and Siddhartha Mishra. Entropy stable shock capturing space-time discontinuous galerkin schemes for systems of conservation laws. *Numerische Mathematik*, 126(1):103–151, 2014.

[18] F. Hindenlang, G. Gassner, C. Altmann, A. Beck, M. Staudenmaier, and C.-D. Munz. Explicit discontinuous Galerkin methods for unsteady problems. *Computers and Fluids*, 61:86–93, 2012.

[19] Farzad Ismail and Philip L. Roe. Affordable, entropy-consistent Euler flux functions II: Entropy production at shocks. *Journal of Computational Physics*, 228(15):5410–5436, 2009.

[20] Antony Jameson. Formulation of kinetic energy preserving conservative schemes for gas dynamics and direct numerical simulation of one-dimensional viscous compressible flow in a shock tube using entropy and kinetic energy preserving schemes. *Journal of Scientific Computing*, 34(3):188–208, 2008.

[21] Guangshan Jiang and Chi-Wang Shu. On a cell entropy inequality for discontinuous galerkin methods. *Mathematics of Computation*, 62(206):pp. 531–538, 1994.

[22] Christopher A. Kennedy and Andrea Gruber. Reduced aliasing formulations of the convective terms within the Navier–Stokes equations for a compressible fluid. *Journal of Computational Physics*, 227:1676–1700, 2008.

[23] R.M. Kirby and G.E. Karniadakis. De-aliasing on non-uniform grids: algorithms and applications. *Journal of Computational Physics*, 191:249–264, 2003.

[24] David A. Kopriva. Metric identities and the discontinuous spectral element method on curvilinear meshes. *Journal of Scientific Computing*, 26(3):301–327, March 2006.

[25] David A. Kopriva. *Implementing Spectral Methods for Partial Differential Equations: Algorithms for Scientists and Engineers.* Springer Publishing Company, Incorporated, 1st edition, 2009.

[26] David A. Kopriva and Gregor J. Gassner. On the quadrature and weak form choices in collocation type discontinuous Galerkin spectral element methods. *Journal of Scientific Computing*, 44(2):136–155, 2010-08-01.

[27] David A. Kopriva and Gregor J. Gassner. An energy stable discontinuous Galerkin spectral element discretization for variable coefficient advection problems. *SIAM Journal on Scientific Computing*, 36(4):2076–2099, 2014.

[28] A.G. Kravchenko and P. Moin. On the effect of numerical errors in large eddy simulations of turbulent flows. *Journal of Computational Physics*, 131(2):310 – 322, 1997.

[29] J Larsson, S. K. Lele, and Moin P. Effect of numerical dissipation on the predicted spectra for compressible turbulence. *Annual Research Briefs*, pages 47–57, 2007.

[30] G Mengaldo, D De Grazia, D Moxey, P. E. Vincent, and S. J. Sherwin. Dealiasing techniques for high-order spectral element methods on regular and irregular grids. *Journal of Computational Physics*, 299:56–81, 2015.

[31] Yohei Morinishi. Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-Mach number flows. *Journal of Computational Physics*, 229(2):276 – 300, 2010.

[32] Rodrigo Costa Moura, Spencer J. Sherwin, and Joaquim Peiro. On DG-based iLES approaches at very high Reynolds numbers. Report, Research Gate, 2015.

[33] Matteo Parsani, Mark H Carpenter, and Eric J Nielsen. Entropy stable discontinuous interfaces coupling for the three-dimensional compressible Navier-Stokes equations. *Journal of Computational Physics*, 290:132–138, 2015.

[34] Matteo Parsani, Mark H Carpenter, and Eric J Nielsen. Entropy stable wall boundary conditions for the three-dimensional compressible Navier-Stokes equations. *Journal of Computational Physics*, 292:88–113, 2015.

[35] Sergio Pirozzoli. Generalized conservative approximations of split convective derivative operators. *Journal of Computational Physics*, 229(19):7180–7190, 2010.

[36] Sergio Pirozzoli. Numerical methods for high-speed flows. *Annual Review of Fluid Mechanics*, 43:163–194, 2011.

[37] Bo Strand. Summation by parts for finite difference approximations for d/dx. *Journal of Computational Physics*, 110(1):47 – 67, 1994.

[38] Magnus Svärd and Hatice Özcan. Entropy-stable schemes for the Euler equations with far-field and wall boundary conditions. *Journal of Scientific Computing*, 58(1):61–89, 2014.

[39] Niklas Wintermeyer, Andrew R. Winters, Gregor J. Gassner, and David A. Kopriva. An entropy stable nodal discontinuous Galerkin method for the two dimensional shallow water equations on unstructured curvilinear meshes with discontinuous bathymetry. *Journal of Computational Physics*, (submitted), 2015.

[40] Thomas A. Zang. On the rotation and skew-symmetric forms for incompressible flow simulations. *Applied Numerical Mathematics*, 7:27 – 40, 1991.