







Generalized upwind summation-by-parts operators and their application to nodal discontinuous Galerkin methods

Jan Glaubitz^{a,b,,*}, Hendrik Ranocha^{c,}, Andrew R. Winters^{b,},
Michael Schlottke-Lakemper^{d,e,}, Philipp Öffner^{f,}, Gregor Gassner^{g,}

^a Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, USA

^b Department of Mathematics, Linköping University, Sweden

^c Institute of Mathematics, Johannes Gutenberg University Mainz, Germany

^d High-Performance Scientific Computing, University of Augsburg, Germany

^e High-Performance Computing Center Stuttgart, Germany

^f Institute of Mathematics, TU Clausthal, Germany

^g Department of Mathematics and Computer Science & Center for Data and Simulation Science, University of Cologne, Germany

ARTICLE INFO

Dataset link: <https://github.com/trixi-framework/paper-2024-generalized-upwind-sbp>

MSC:

65M12

65M60

65M70

65D25

Keywords:

Upwind summation-by-parts operators

Conservation laws

Flux vector splittings

Nodal discontinuous Galerkin methods

ABSTRACT

High-order numerical methods for conservation laws are highly sought after due to their potential efficiency. However, it is challenging to ensure their robustness, particularly for under-resolved flows. Baseline high-order methods often incorporate stabilization techniques that must be applied judiciously—sufficient to ensure simulation stability but restrained enough to prevent excessive dissipation and loss of resolution. Recent studies have demonstrated that combining upwind summation-by-parts (USBP) operators with flux vector splitting can increase the robustness of finite difference (FD) schemes without introducing excessive artificial dissipation. This work investigates whether the same approach can be applied to nodal discontinuous Galerkin (DG) methods. To this end, we demonstrate the existence of USBP operators on arbitrary grid points and provide a straightforward procedure for their construction. Our discussion encompasses a broad class of USBP operators, not limited to equidistant grid points, and enables the development of novel USBP operators on Legendre–Gauss–Lobatto (LGL) points that are well-suited for nodal DG methods. We then examine the robustness properties of the resulting DG-USBP methods for challenging examples of the compressible Euler equations, such as the Kelvin–Helmholtz instability. Similar to high-order FD-USBP schemes, we find that combining flux vector splitting techniques with DG-USBP operators does not lead to excessive artificial dissipation. Furthermore, we find that combining lower-order DG-USBP operators on three LGL points with flux vector splitting indeed increases the robustness of nodal DG methods. However, we also observe that higher-order USBP operators offer less improvement in robustness for DG methods compared to FD schemes. We provide evidence that this can be attributed to USBP methods adding dissipation only to unresolved modes, as FD schemes typically have more unresolved modes than nodal DG methods.

* Corresponding author at: Department of Mathematics, Linköping University, Sweden.

E-mail addresses: jan.glaubitz@liu.se (J. Glaubitz), hendrik.ranocha@uni-mainz.de (H. Ranocha), andrew.ross.winters@liu.se (A.R. Winters), michael.schlottke-lakemper@uni-a.de (M. Schlottke-Lakemper), mail@philippoeffner.de (P. Öffner), ggassner@uni-koeln.de (G. Gassner).

<https://doi.org/10.1016/j.jcp.2025.113841>

Received 14 June 2024; Received in revised form 7 February 2025; Accepted 10 February 2025

1. Introduction

The simulation of many phenomena in science and engineering relies on efficient and robust methods for numerically solving hyperbolic conservation laws. Entropy-based methods have become increasingly favored for developing robust high-order methods applicable to various applications. Initially introduced for second-order methods in the seminal work [102,103], these schemes have since evolved significantly. Expanding on these foundational studies, [52,27] introduced high-order extensions tailored to periodic domains. Further advancements were made in [26], where the authors developed the *flux differencing* approach to entropy stability. This approach, based on SBP operators and two-point numerical fluxes, naturally extends to high-order operators based on complex geometries.

SBP operators are instrumental in ensuring a discrete analog of integration by parts, thereby facilitating the extension of properties of numerical fluxes to high-order methods. SBP operators were initially developed in the 1970s, specifically for FD methods [50,51,93]. More contemporary overviews and references are provided in [101,20,12]. While the corpus of literature on SBP-based methods is too extensive to enumerate fully, notable implementations span a range of schemes, including spectral element [8], DG [28,30,11], continuous Galerkin [1,2], finite volume (FV) [68,69], flux reconstruction (FR) [85,72,15,14,65], and radial basis function (RBF) [35] methods.

Several studies have established that flux differencing schemes, particularly DG spectral element methods (DGSEMs), are effective for under-resolved flows, such as the Taylor–Green vortex, when constructed to ensure entropy stability and positivity preservation. These properties are critical, especially in real applications involving strong shocks and high Reynolds numbers. See [30,96,46,61,60,91,75,57,71,110,66] and references therein. Furthermore, see [3,18,88,89] for additional applications of the DGSEM approach to multicomponent and multiphase flows. Entropy-stable high-order methods, like those studied in [61,75,110,66], possessed enhanced robustness for under-resolved flows without any built-in positivity preservation considerations. However, works such as [29,83] have demonstrated linear stability issues for high-order entropy-stable methods. Despite their non-linear stability, these schemes exhibit shortcomings in seemingly straightforward test cases, such as Euler equations with constant velocity and pressure—reducing them to a simple linear advection for the density. In contrast, central schemes without entropy properties perform well in this case but are prone to failure in more challenging simulations of under-resolved flows. Some resulting challenges in numerical simulations can be mitigated using shock-capturing or invariant domain-preserving techniques. For example, issues such as non-negativity and spurious entropy growth can be effectively addressed by implementing positivity-preserving methods [112,78] and artificial viscosity approaches [77,84], respectively. However, it is crucial to apply these stabilization techniques judiciously—sufficient to ensure simulation stability but restrained enough to prevent excessive dissipation and loss of resolution.

In this work, we are concerned with developing robust and efficient baseline schemes by employing USBP operators within a nodal DG framework—without focusing on positivity preservation, however. Introduced in [62] for FD methods, USBP operators emerged as a particular class of dual-pair SBP operators [23]. Henceforth, we refer to these operators as *FD-USBP operators*. These FD-USBP operators have been extended to staggered grids in [63] and applied in the context of the shallow water equations and residual-based artificial viscosity methods in [58,98], respectively. Furthermore, for instance, [74] demonstrated that DG discretizations of linear advection-diffusion equations can be analyzed using the framework of global USBP operators. For nonlinear conservation laws, USBP operators are combined with flux vector splitting techniques [104, Chapter 8]. Flux vector splitting techniques were extensively developed and applied throughout the previous century [97,105,7,41,56,16] yet eventually fell out of favor. This shift occurred as these methods were superseded by alternative approaches, primarily due to the considerable numerical dissipation they introduced [106]. Yet, previous studies [87,24] have demonstrated that combining high-order USBP operators with flux vector splittings can increase the robustness of FD schemes for under-resolved simulations without introducing excessive artificial dissipation.

Our contribution

In this work, we investigate whether combining USBP operators with flux vector splittings can be applied to nodal DG methods with similar success. To this end, we first demonstrate the existence of USBP operators on arbitrary grid points and provide a straightforward procedure for their construction. While previous works have developed diagonal-norm FD-USBP operators, our discussion encompasses a broader class of USBP operators. Importantly, the proposed construction procedure applies to arbitrary grid points, not limited to equidistant spacing. This generalization enables the development of novel USBP operators on LGL points that are well-suited for nodal DG methods. In contrast to, for instance, [74], we not only use a global upwind formulation of DG methods but also develop local USBP operators for nodal DG-type methods. These operators achieve a given order of accuracy with fewer degrees of freedom (nodes) than existing FD-USBP operators [62,63]. We then examine the robustness properties of the resulting DG-USBP methods for challenging examples of the compressible Euler equations, such as Kelvin-Helmholtz instabilities. Similar to high-order FD-USBP schemes, we find that combining flux vector splitting techniques with DG-USBP operators does not lead to excessive artificial dissipation. Furthermore, we demonstrate local linear/energy stability for the DG-USBP semi-discretizations in a setting where existing high-order entropy-stable flux differencing DGSEM produce unstable semi-discretizations. At the same time, we observe that USBP operators offer less improvement in robustness for high-order DG methods than for FD schemes for multi-dimensional systems. Specifically, we note in our numerical simulations that the increase in robustness becomes less significant as more LGL points are used per element and coordinate direction. We suspect that this reduced improvement can be attributed to USBP methods adding dissipation only to unresolved modes, and FD schemes typically have a larger number of unresolved modes than nodal DG methods. Still, DG-USBP methods can serve as a viable alternative approach, as it may be more straightforward to apply in complex problems where a robust flux splitting, such as Lax-Friedrichs, is readily available while conducting an entropy analysis proves too cumbersome.

Outline

In Section 2, we present all necessary background on SBP and USBP operators. Section 3 is devoted to delineating the conditions for the existence of USBP operators. Section 4 outlines a systematic approach for constructing USBP operators. In Section 5, we use the previously developed DG-USBP operators to formulate DG-USBP baseline schemes. Section 6 is reserved for various numerical experiments. Finally, Section 7 offers concluding remarks and explores potential avenues for future research.

2. Preliminaries on SBP and USBP operators

We start by revisiting SBP and USBP operators. For simplicity, we focus on one-dimensional operators on a compact interval.

2.1. Notation

Consider $\Omega = [x_{\min}, x_{\max}]$ as the computational spatial domain for solving a given time-dependent partial differential equation (PDE). Our focus is on SBP operators, which apply to a generic element $\Omega_{\text{ref}} = [x_L, x_R]$. In pseudo-spectral or global FD schemes, $\Omega_{\text{ref}} = \Omega$. However, in DG and multi-block FD methods, Ω is divided into multiple smaller non-overlapping elements Ω_j which collectively cover Ω (i.e., $\bigcup_j \Omega_j = \Omega$). In this case, any Ω_j can serve as Ω_{ref} . Let $\mathbf{x} = [x_1, \dots, x_N]^T$ represent a vector of grid points on the reference element Ω_{ref} , where $x_L \leq x_1 < \dots < x_N \leq x_R$. For a continuously differentiable function f on Ω_{ref} , we denote

$$\mathbf{f} = [f(x_1), \dots, f(x_N)]^T, \quad \mathbf{f}' = [f'(x_1), \dots, f'(x_N)]^T, \quad (2.1)$$

as the nodal values and first derivative of f at the grid points \mathbf{x} . Additionally, \mathcal{P}_d denotes the space of polynomials of degree up to d .

2.2. SBP operators

We are now positioned to define SBP operators. See the reviews [101,20,12] and references therein for more details.

Definition 2.1 (SBP operators). The operator $D = P^{-1}(Q + B/2)$ approximating ∂_x is a *degree d SBP operator* on $\Omega_{\text{ref}} = [x_L, x_R]$ if

- (i) $D\mathbf{f} = \mathbf{f}'$ for all $f \in \mathcal{P}_d$;
- (ii) P is a symmetric and positive definite (SPD) matrix;
- (iii) The boundary matrix B satisfies $\mathbf{f}^T B \mathbf{g} = f g \Big|_{x_L}^{x_R}$ for all $f, g \in \mathcal{P}_d$;
- (iv) $Q + Q^T = 0$ (Q is anti-symmetric).

Relation (i) ensures that D accurately approximates the continuous derivative operator ∂_x by requiring D to be exact for polynomials up to degree d . Condition (ii) guarantees that P induces a proper discrete inner product and norm, which are given by $\langle \mathbf{f}, \mathbf{g} \rangle = \mathbf{f}^T P \mathbf{g}$ and $\|\mathbf{f}\|_P^2 = \mathbf{f}^T P \mathbf{f}$, respectively. Relation (iv) encodes the SBP property, which allows us to mimic integration-by-parts (IBP) on a discrete level. Recall that IBP reads

$$\int_{x_L}^{x_R} f(\partial_x g) dx + \int_{x_L}^{x_R} (\partial_x f) g dx = f g \Big|_{x_L}^{x_R}. \quad (2.2)$$

The discrete version of (2.2), which follows from (iv) in Definition 2.1, is

$$\mathbf{f}^T P D \mathbf{g} + \mathbf{f}^T D^T P^T \mathbf{g} = \mathbf{f}^T B \mathbf{g}. \quad (2.3)$$

Finally, relation (iii) in Definition 2.1 ensures that the right-hand side of (2.3) accurately approximates the one of (2.2) by requiring the boundary operator B to be exact for polynomials up to degree d as well.

2.3. USBP operators

Next, we introduce USBP operators, which originate from the FD setting [62] and are a subclass of dual-pair SBP operators [23].

Definition 2.2 (USBP operators). The operators $D_+ = P^{-1}(Q_+ + B/2)$ and $D_- = P^{-1}(Q_- + B/2)$ approximating ∂_x are *degree d USBP operators* on $\Omega_{\text{ref}} = [x_L, x_R]$ if

- (i) $D_+ \mathbf{f} = D_- \mathbf{f} = \mathbf{f}'$ for all $f \in \mathcal{P}_d$;
- (ii) The norm matrix P is an SPD matrix;
- (iii) The boundary matrix B satisfies $\mathbf{f}^T B \mathbf{g} = f g \Big|_{x_L}^{x_R}$ for all $f, g \in \mathcal{P}_\tau$ with $\tau \geq d$;
- (iv) $Q_+ + Q_- = 0$;

(v) $Q_+ + Q_+^T = S$, where the *dissipation matrix* S is symmetric and negative semi-definite.

The motivation for conditions (i), (ii), and (iii) in Definition 2.2 is the same as in Definition 2.1 of SBP operators. However, the discrete version of IBP (2.2) that follows from (iv) in Definition 2.2 is

$$\mathbf{f}^T P D_+ \mathbf{g} + \mathbf{f}^T D_-^T P^T \mathbf{g} = \mathbf{f}^T B \mathbf{g}. \quad (2.4)$$

At the same time, (v) introduces accurate artificial dissipation into the USBP discretization. This can be seen as follows (also see [62,55,53]): Using (iv) and (v), we observe that

$$D_+ = D_- + P^{-1} S, \quad (2.5)$$

and it follows that $D_+ \mathbf{f} = D_- \mathbf{f}$ if and only if $S \mathbf{f} = \mathbf{0}$. This means that S adds artificial dissipation that acts only on unresolved modes, $\mathbf{f} \in \mathbb{R}^N$ for which $f \notin \mathcal{P}_d$. The artificial dissipation included in USBP operators only acting on unresolved modes distinguishes them from combining SBP operators with traditional artificial dissipation terms based on second (or higher-order even) derivatives [64,84]. Lemma 2.3 connects SBP and USBP operators.

Lemma 2.3 (USBP operators induce SBP operators). *If $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ are degree d USBP operators, then*

$$D := (D_+ + D_-)/2 = P^{-1}(Q + B/2) \quad \text{with} \quad Q = (Q_+ + Q_-)/2 \quad (2.6)$$

is a degree d SBP operator. Moreover, $D_+ - D_- = P^{-1} S$.

Proof. Lemma 2.3 was demonstrated in [62] and we thus omit its proof. \square

We will need Lemma 2.3 to prove Theorem 3.1 below, which characterizes the existence of USBP operators.

3. Characterizing the existence of USBP operators

We now characterize the existence of USBP operators as in Definition 2.2 for arbitrary grid points. We subsequently leverage this analysis in Section 4 to construct novel USBP operators.

Theorem 3.1. *Let $\mathbf{x} = [x_1, \dots, x_N]^T$ be a grid on $\Omega_{\text{ref}} = [x_L, x_R]$ and let S be a symmetric and negative semi-definite dissipation matrix.*

- (a) *If there exists a degree d SBP operator $D = P^{-1}(Q + B/2)$ and S satisfies $S \mathbf{f} = \mathbf{0}$ for all $f \in \mathcal{P}_d$, then $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ with $Q_{\pm} = Q \pm S/2$ are degree d USBP operators with $Q_+ + Q_+^T = S$.*
- (b) *If there exist degree d USBP operators $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ with $Q_+ + Q_+^T = S$, then $D = (D_+ + D_-)/2$ is a degree d SBP operator and S satisfies $S \mathbf{f} = \mathbf{0}$ for all $f \in \mathcal{P}_d$.*

In a nutshell, Theorem 3.1 tells us that there exist degree d USBP operators D_{\pm} on a given grid if and only if there exists a degree d SBP operator D on the same grid and a dissipation matrix S with $S \mathbf{f} = \mathbf{0}$ for all $f \in \mathcal{P}_d$. The proof of Theorem 3.1 is provided in Appendix A.

While the dissipation matrix S should not add dissipation to resolved modes, i.e., $S \mathbf{f} = \mathbf{0}$ for all $f \in \mathcal{P}_d$, it should do so for *unresolved* modes. The latter means that

$$\mathbf{f}^T S \mathbf{f} < 0 \quad \forall f \notin \mathcal{P}_d. \quad (3.1)$$

We follow up with Lemma 3.2, which shows that a desired dissipation matrix S with $S \mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ and $\mathbf{f}^T S \mathbf{f} < 0$ otherwise always exists.

Lemma 3.2. *Let $d \in \mathbb{N}$, \mathbf{x} be a grid with $N \geq d + 1$ distinct points. There always exists a symmetric and negative semi-definite matrix S with $S \mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ and $\mathbf{f}^T S \mathbf{f} < 0$ otherwise. Moreover, $S = 0$ if $N = d + 1$ and $S \neq 0$ if $N > d + 1$.*

Proof. If $N = d + 1$, then any vector $\mathbf{f} \in \mathbb{R}^N$ corresponds to the nodal values of a function $f \in \mathcal{P}_d$ on the grid \mathbf{x} . There are no unresolved modes in this case, and $S = 0$ is the desired matrix.

Now assume that $N > d + 1$. Since the matrix S is real and symmetric, it can be decomposed as

$$S = V \Lambda V^{-1}, \quad (3.2)$$

where $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ contains the eigenvalues of S and the columns of $V = [\mathbf{v}_1, \dots, \mathbf{v}_N]$ are linearly independent eigenvectors of S ; see [39, Theorem 8.1.1]. Let $\{f_k\}_{k=1}^{d+1}$ be a basis of \mathcal{P}_d , e.g., monomials or Legendre polynomials up to degree d . Then, $S \mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ is equivalent to $S \mathbf{f}_k = \mathbf{0}$ for $k = 1, \dots, d + 1$. At the same time, the latter condition means that $\mathbf{f}_1, \dots, \mathbf{f}_{d+1}$ are eigenvectors

of S with the corresponding eigenvalue being zero. For simplicity, we position these as the first $d + 1$ eigenvalues and eigenvectors in (3.2), i.e.,

$$\lambda_k = 0, \quad \mathbf{v}_k = \mathbf{f}_k, \quad k = 1, \dots, d + 1. \quad (3.3)$$

Then, by construction, the matrix S in (3.2) is symmetric and satisfies $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$. Furthermore, since $N > d + 1$, we still have $N - d - 1 > 0$ free eigenvalues that we can use to ensure that $\mathbf{f}^T S\mathbf{f} < 0$ if $f \notin \mathcal{P}_d$. To this end, the remaining eigenvalues must be negative, i.e., $\lambda_{d+2}, \dots, \lambda_N < 0$. Finally, we can set $\mathbf{v}_{d+2}, \dots, \mathbf{v}_N$ as arbitrary vectors that are linearly independent among themselves and to the first $d + 1$ eigenvectors. Then, the resulting matrix S in (3.2) is symmetric and negative semi-definite while satisfying $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$. It remains to show that $\mathbf{f}^T S\mathbf{f} < 0$ if $f \notin \mathcal{P}_d$. To this end, suppose that $f \notin \mathcal{P}_d$. In this case, the corresponding vector of nodal values \mathbf{f} can be expressed as $\mathbf{f} = \gamma_1 \mathbf{v}_1 + \dots + \gamma_N \mathbf{v}_N$ with $\gamma_k \neq 0$ for at least one $k > d + 1$. (At least one of the last $N - d - 1$ eigenvectors contributes to representing \mathbf{f} .) Then, we see that

$$\mathbf{f}^T S\mathbf{f} = \underbrace{\sum_{k=1}^{d+1} \lambda_k \gamma_k^2 \mathbf{v}_k^T \mathbf{v}_k}_{=0} + \underbrace{\sum_{k=d+2}^N \lambda_k \gamma_k^2 \mathbf{v}_k^T \mathbf{v}_k}_{<0} < 0, \quad (3.4)$$

which completes the proof. \square

Many modern collocation-type DG methods are based on interpolatory ($N = d + 1$) SBP operators. See [28,30,46,60,12,75,110] and references therein. However, Lemma 3.2 indicates that there *cannot* be interpolatory USBP operators with a non-trivial dissipation matrix S . Conversely, Lemma 3.2 also suggests that if $N > d + 1$ (even with just an oversampling of one point), the construction of such USBP operators becomes viable.

4. Constructing USBP operators

We now provide a simple procedure to construct degree d USBP operators $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$. While previous works, such as [62], have developed diagonal-norm FD-USBP operators, our method encompasses a broader class of USBP operators. Importantly, it applies to arbitrary grid points, not limited to equidistant spacing. This allows us to derive novel USBP operators on LGL points suited for nodal DG methods.

4.1. Proposed procedure

The idea behind our construction procedure is to start from a degree d SBP operator $D = P^{-1}(Q + B/2)$ and a dissipation matrix S with $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ and $\mathbf{f}^T S\mathbf{f} < 0$ otherwise. Then part (a) of Theorem 3.1 implies that $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ with $Q_{\pm} = Q \pm S/2$ are degree d USBP operators. This observation motivates the following simple construction procedure:

- (P1) Construct a degree d SBP operator $D = P^{-1}(Q + B/2)$ on the grid $\mathbf{x} = [x_1, \dots, x_N]^T$ with $N > d + 1$.
- (P2) Find a dissipation matrix S with $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ and $\mathbf{f}^T S\mathbf{f} < 0$ otherwise.
- (P3) Choose $Q_{\pm} = Q \pm S/2$ and get degree d USBP operators $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ with $Q_+ + Q_-^T = S$ on the grid \mathbf{x} .

In (P3), equivalently, we get degree d USBP operators as $D_{\pm} = D \pm P^{-1}S$. Note that we choose $N > d + 1$ in (P2) because Lemma 3.2 states that we can find a dissipation matrix $S \neq 0$ if and only if $N > d + 1$.

The first step in our construction procedure, (P1), is to find a degree d SBP operator on a grid $\mathbf{x} = [x_1, \dots, x_N]^T$ with $N > d + 1$. Several works have discussed the construction of SBP operators. See [44,54,34,36] or the reviews [101,20] and references therein. Therefore, we omit such a discussion and focus instead on the construction of S in step (P2), which is addressed in Subsection 4.2, along with a robust implementation strategy in Subsection 4.3. Furthermore, we exemplify the above procedure in Subsection 4.4.

4.2. Constructing a dissipation matrix

The second step of our construction procedure, (P2), is to find a dissipation matrix S with $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ and $\mathbf{f}^T S\mathbf{f} < 0$ otherwise. To this end, we can follow the proof of Lemma 3.2. Let us briefly recall the most important steps and comment on their implementation for completeness. Since S is real and symmetric, we can decompose it as

$$S = V\Lambda V^{-1}, \quad (4.1)$$

where $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ contains the eigenvalues of S and the columns of $V = [\mathbf{v}_1, \dots, \mathbf{v}_N]$ are linearly independent eigenvectors of S . (See [39, Theorem 8.1.1].) Let $\{f_k\}_{k=1}^{d+1}$ be a basis of \mathcal{P}_d . We can ensure $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_d$ (the dissipation matrix does not act on resolved modes) by using the nodal value vectors of the basis functions as the first $d + 1$ eigenvectors with the corresponding eigenvalues being zero, i.e.,

$$\mathbf{v}_k = \mathbf{f}_k, \quad \lambda_k = 0, \quad k = 1, \dots, d + 1. \quad (4.2)$$

These are linearly independent since \mathcal{P}_d is a Chebyshev system. Now suppose that $N > d + 1$. Then, we still have $N - d - 1 > 0$ free eigenvalues and eigenvectors. These have to be chosen such that (i) the eigenvectors $\mathbf{v}_{d+2}, \dots, \mathbf{v}_N$ are linearly independent among themselves and to the first $d + 1$ eigenvectors and (ii) one has $\mathbf{f}^T S \mathbf{f} < 0$ if $f \notin \mathcal{P}_d$. We can ensure these properties by choosing the eigenvalues $\lambda_{d+2}, \dots, \lambda_N$ as arbitrary *negative* numbers, e.g., $\lambda_{d+2}, \dots, \lambda_N = -1$, and $\mathbf{v}_{d+2}, \dots, \mathbf{v}_N$ as the corresponding eigenvectors satisfying $S \mathbf{v}_j = \lambda_j \mathbf{v}_j$ for $j = d + 2, \dots, N$. Then, S is a desired dissipation matrix, and we can formally find it by computing the matrix product on the right-hand side of (4.1). That said, computing the matrix product in (4.1) is generally not recommended since it involves matrix inversion. We discuss an approach to bypass this problem in Subsection 4.3.

4.3. Robust implementation using discrete orthogonal polynomials

We present a simple and robust method for computing the dissipation matrix S in (4.1) using discrete orthogonal polynomials (DOPs), see [31]. Suppose we are given a grid $\mathbf{x} = [x_1, \dots, x_N]^T$. A basis $\{f_k\}_{k=1}^N$ of \mathcal{P}_N consists of DOPs if the f_k 's are orthogonal with respect to the discrete inner product $[f, g]_{\mathbf{x}} = \sum_{n=1}^N f(x_n)g(x_n)$, i.e.,

$$\sum_{n=1}^N f_j(x_n)f_k(x_n) = \delta_{j,k}, \quad j, k = 1, \dots, N, \tag{4.3}$$

where $\|f_k\|_{\mathbf{x}}^2 = [f_k, f_k]_{\mathbf{x}}$ is the induced norm. Note that (4.3) is equivalent to $\langle \mathbf{f}_j, \mathbf{f}_k \rangle = \delta_{j,k}$, where $\langle \mathbf{f}, \mathbf{g} \rangle = \mathbf{f}^T \mathbf{g}$ and $\|\mathbf{f}\|^2 = \mathbf{f}^T \mathbf{f}$ are the usual Euclidean inner product and norm for $\mathbf{f}, \mathbf{g} \in \mathbb{R}^N$, respectively. Hence, if we choose $\mathbf{v}_k = \mathbf{f}_k$ for the eigenvectors in (4.1), where $\{f_k\}_{k=1}^N$ is a DOP basis of \mathcal{P}_N , then $V = [\mathbf{v}_1, \dots, \mathbf{v}_N]$ is orthogonal ($V^{-1} = V^T$) and (4.1) becomes

$$S = V \Lambda V^T. \tag{4.4}$$

The advantage of (4.4) over (4.1) is that we no longer need to invert V . To summarize the above discussion, we can efficiently and robustly construct a dissipation matrix S as follows:

- (S1) Find a DOP basis $\{f_k\}_{k=1}^N$ of \mathcal{P}_N with respect to the grid $\mathbf{x} = [x_1, \dots, x_N]^T$.
- (S2) Choose the columns of V as the vectors of nodal values of the DOP basis, i.e., $\mathbf{v}_k = \mathbf{f}_k$ for $k = 1, \dots, N$.
- (S3) Set the first $d + 1$ diagonal entries of Λ in (4.4) to zero, i.e., $\lambda_k = 0$ for $k = 1, \dots, d + 1$.¹
- (S4) Choose the remaining eigenvalues $\lambda_{d+2}, \dots, \lambda_N < 0$ and compute S using (4.4).

A few remarks are in order.

Remark 4.1. Since the dissipation matrix S is symmetric, the eigenvectors corresponding to *different* eigenvalues are automatically orthogonal. However, in our case, we encounter multiple linear independent eigenvectors associated with the same eigenvalue (for instance, $\lambda = 0$). Consequently, these are not guaranteed to be orthogonal; therefore, we rely on the additional orthogonalization step described above.

Remark 4.2 (Finding DOP bases). The existing literature offers explicit formulas of DOPs for different point distributions \mathbf{x} , including Chebyshev–Gauss–Lobatto [25] and equidistant points [109]. Suppose an explicit formula is not readily available. In that case, we can start from a basis of Legendre polynomials, orthogonal with respect to the continuous L^2 -inner product, and transform them into a DOP basis using the modified Gram–Schmidt process. See [31,37] and references therein.

4.4. An example: USBP operators on LGL points

We exemplify the construction of USBP operators for a degree one ($d = 1$) diagonal-norm USBP operator for three ($N = 3$) LGL points on $\Omega_{\text{ref}} = [-1, 1]$. This case allows us to represent exactly all involved matrices below. Additional diagonal-norm USBP operators on up to six LGL nodes are provided in Appendix B. Moreover, Appendix C exemplifies the construction of USBP operators for Gauss–Legendre nodes (that do not include the boundary points) and a dense-norm USBP operator on equidistant points.

Let us first address (P1): To compute a degree one diagonal-norm SBP operator on three LGL points, we have two options: The first option is to determine a diagonal-norm SBP operator with exactly degree one, i.e., the operator is exact for linear functions but not necessarily for quadratic ones. We can find such an operator following the procedure described in [44,54,34]. Note that this operator is not unique but comes with one degree of freedom that can be used to optimize the SBP operator based on different criteria, such as the truncation error and spectral properties [99,101,20]. The second option is to determine a degree two diagonal-norm SBP operator. This operator is unique and readily available by evaluating the derivatives of the corresponding Lagrange basis functions at the LGL grid points [28,8,9,11,15,65]. We follow the second option since it is computationally more convenient. The associated grid points, quadrature weights (diagonal entries of P), and SBP operator on $\Omega_{\text{ref}} = [-1, 1]$ are

¹ We assume that the first $d + 1$ DOPs form a basis of \mathcal{P}_d . This ensures that the dissipation operator S does not act on the resolved modes corresponding to polynomials of degree up to d .

$$\mathbf{x} = \begin{bmatrix} -1 \\ 0 \\ 1 \end{bmatrix}, \quad \mathbf{p} = \frac{1}{3} \begin{bmatrix} 1 \\ 4 \\ 1 \end{bmatrix}, \quad D = \frac{1}{2} \begin{bmatrix} -3 & 4 & -1 \\ -1 & 0 & 1 \\ 1 & -4 & 3 \end{bmatrix}. \quad (4.5)$$

The boundary matrix is $B = \text{diag}(-1, 0, 1)$ since the boundary points are included in the grid.

We next address (P2), finding a dissipation matrix S with $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_1$ and $\mathbf{f}^T S \mathbf{f} < 0$ otherwise. To this end, we follow Subsection 4.3 and robustly construct S using a DOP basis on the three LGL grid points. We get these by starting from the first three Legendre polynomials, spanning \mathcal{P}_3 , and then applying the Gram–Schmidt procedure. The Vandermonde matrix of this DOP basis is

$$V = \frac{1}{\sqrt{6}} \begin{bmatrix} \sqrt{2} & -\sqrt{3} & 1 \\ \sqrt{2} & 0 & -2 \\ \sqrt{2} & \sqrt{3} & 1 \end{bmatrix}. \quad (4.6)$$

Consequently, we get the dissipation matrix as $S = V\Lambda V^T$ with $\Lambda = \text{diag}(\lambda_1, \lambda_2, \lambda_3)$. To ensure that the desired USBP operator has degree one ($d = 1$), we choose the first two eigenvalues as zero, i.e., $\lambda_1 = \lambda_2 = 0$. Furthermore, we choose $\lambda_3 = -1$, which introduces artificial dissipation to the highest, unresolved, mode.² The resulting dissipation matrix is

$$S = \frac{1}{6} \begin{bmatrix} -1 & 2 & -1 \\ 2 & -4 & 2 \\ -1 & 2 & -1 \end{bmatrix}. \quad (4.7)$$

Finally, (P3), we get degree one diagonal-norm USBP operators as $D_{\pm} = D \pm P^{-1}S/2$, yielding

$$D_+ = \frac{1}{8} \begin{bmatrix} -14 & 20 & -6 \\ -3 & -2 & 5 \\ 2 & -12 & 10 \end{bmatrix}, \quad D_- = \frac{1}{8} \begin{bmatrix} -10 & 12 & -2 \\ -5 & 2 & 3 \\ 6 & -20 & 14 \end{bmatrix}. \quad (4.8)$$

The norm and boundary matrix of the degree one USBP operators are the same as those of the degree two SBP operator.

5. Application of USBP operators to nodal DG methods

We are now positioned to combine the above-developed DG-USBP operators with flux vector splitting approaches to construct new DG-USBP baseline schemes for nonlinear conservation laws.

5.1. The flux vector splitting approach

We briefly review the classical flux vector splitting approach. See [97,105,7,41,56,16] as well as [104, Chapter 8] and references therein for more details. Consider the generic one-dimensional hyperbolic conservation law

$$\partial_t u + \partial_x f(u) = 0, \quad x \in \Omega = (x_{\min}, x_{\max}), \quad (5.1)$$

with conserved variable $u = u(t, x)$ and flux $f = f(u)$. Henceforth, we assume that (5.1) is equipped with appropriate initial and boundary conditions. In flux vector splitting approaches, the flux f is split into two parts,

$$f(u) = f_-(u) + f_+(u), \quad (5.2)$$

where the eigenvalues $\lambda_i^{\pm}(u)$ of the Jacobian $\nabla_u f_{\pm}(u)$ satisfy $\lambda_i^-(u) \leq 0$ and $\lambda_i^+(u) \geq 0$ for all i , respectively. Applying (5.2) to (5.1), we get

$$\partial_t u + \partial_x f_-(u) + \partial_x f_+(u) = 0, \quad x \in \Omega = [x_{\min}, x_{\max}]. \quad (5.3)$$

We can now semi-discretize (5.3) using ‘global’ USBP operators D_{\pm} on $\Omega = [x_{\min}, x_{\max}]$, resulting in

$$\frac{d}{dt} \mathbf{u} + D_+ \mathbf{f}_- + D_- \mathbf{f}_+ = 0, \quad (5.4)$$

where \mathbf{u} and \mathbf{f}_{\pm} are the nodal values of the numerical solution and the two flux components at the global grid points. Note that we have ignored boundary conditions for the moment.

Remark 5.1. The mismatch between the indices \pm of the upwind operators and the fluxes is intentional. This choice is made to maintain backward compatibility, preserving the established conventions of both flux vector splitting methods and upwind SBP operators for historical consistency.

² Note that $\lambda_3 = -1$ is an arbitrary choice. Any negative value yields an admissible dissipation matrix. We investigate the influence of the choice of λ_3 in more detail in Section 6.

5.2. Application to nodal collocated DG methods

In DG-type methods, the computational domain $\Omega = [x_{\min}, x_{\max}]$ is partitioned into J disjoint elements Ω_j with $\bigcup_{j=1}^J \Omega_j = \Omega$. Each element Ω_j is mapped diffeomorphically onto a reference element, say $\Omega_{\text{ref}} = [-1, 1]$, in which all computations are carried out. This has the advantage of only computing USBP operators D_{\pm} once for the reference element rather than for each element separately. Applying this DG-type approach to (5.3), we get the DG-USBP semi-discretization

$$\frac{d}{dt} \mathbf{u}^j + D_+ \mathbf{f}_-^j + D_- \mathbf{f}_+^j = \mathbf{0}, \quad j = 1, \dots, J, \quad (5.5)$$

where \mathbf{u}^j and \mathbf{f}_{\pm}^j are the nodal values of the numerical solution and the two flux vector splitting components on the j th element. We allow for the numerical solution to be discontinuous at element interfaces. Hence, at each element interface, we have two values. We distinguish between values from inside Ω_j , e.g., u_L^j and u_R^j , and values from the neighboring elements, Ω_{j-1} and Ω_{j+1} , e.g., u_R^{j-1} and u_L^{j+1} . Thereby, the subindices ‘‘L’’ and ‘‘R’’ denote that the value of the numerical solution at the left and right element boundary is taken, respectively. From conservation and stability (upwinding) considerations—and to allow for information to be shared between neighboring elements—we introduce a weak coupling between elements in (5.5) using simultaneous approximation terms (SATs), resulting in the semi-discretization

$$\frac{d}{dt} \mathbf{u}^j + D_+ \mathbf{f}_-^j + D_- \mathbf{f}_+^j = \text{SAT}^j, \quad j = 1, \dots, J. \quad (5.6)$$

Please see [101,20,8,21,12,60] and references therein for details on SATs. Here, the SAT in the j -th element is

$$\text{SAT}^j = -P^{-1} \mathbf{e}_R \left[f^{\text{num}}(u_R^j, u_L^{j+1}) - f_R^j \right] + P^{-1} \mathbf{e}_L \left[f^{\text{num}}(u_R^{j-1}, u_L^j) - f_L^j \right], \quad (5.7)$$

where $u_{L/R}^j = \mathbf{e}_{L/R}^T \mathbf{u}^j$ and $f_{L/R}^j = \mathbf{e}_{L/R}^T \mathbf{f}^j$ abbreviate the values of the numerical solution and the corresponding values of the flux at the left/right boundary of the element Ω_j . For simplicity, we suppose that the grid includes the boundary points, i.e., $x_1 = -1$ and $x_N = 1$ in the reference element. Otherwise, one has to incorporate certain projection operators, which introduce additional technical details that we omit here. If the boundary points are included, $\mathbf{e}_L = [1, 0, \dots, 0]^T$ and $\mathbf{e}_R = [0, \dots, 0, 1]^T$, $u_{L/R}^j = u_{1/N}^j$, and $f_{L/R}^j = f_{1/N}^j$. Furthermore, $f^{\text{num}}(u_R^j, u_L^{j+1})$ and $f^{\text{num}}(u_R^{j-1}, u_L^j)$ are two-point numerical flux functions at the right and left boundary of the element Ω_j , respectively. There are many existing numerical fluxes that we can use. Alternatively, as described in [87], we can use the flux vector splitting to design numerical fluxes. In this approach, we use the same splitting for f and f^{num} , yielding

$$\begin{aligned} f^{\text{num}}(u_R^j, u_L^{j+1}) - f_R^j &= \left[f_+(u_R^j) + f_-(u_L^{j+1}) \right] - \left[f_+(u_R^j) + f_-(u_R^j) \right] = f_-(u_L^{j+1}) - f_-(u_R^j), \\ f^{\text{num}}(u_R^{j-1}, u_L^j) - f_L^j &= \left[f_+(u_R^{j-1}) + f_-(u_L^j) \right] - \left[f_+(u_L^j) + f_-(u_L^j) \right] = f_+(u_R^{j-1}) - f_+(u_L^j). \end{aligned} \quad (5.8)$$

Substituting (5.8) into the USBP-SAT semi-discretization (5.6) results in

$$\frac{d}{dt} \mathbf{u}^j + D_+ \mathbf{f}_-^j + D_- \mathbf{f}_+^j = -P^{-1} \mathbf{e}_R \left[f_-(u_L^{j+1}) - f_-(u_R^j) \right] + P^{-1} \mathbf{e}_L \left[f_+(u_R^{j-1}) - f_+(u_L^j) \right] \quad (5.9)$$

for $j = 1, \dots, N$. In our implementation, we opted for the scheme given by (5.9). This choice has the advantage of preventing the need to resolve a Riemann problem in terms of numerical fluxes, which streamlines the implementation and improves efficiency. Finally, one evolves the numerical solution in time by solving the semi-discrete equation (5.9) with an appropriate ODE solver.

6. Numerical tests

We now examine the robustness properties of the DG-USBP method described in Section 5 based on the new DG-USBP operators we constructed in Section 4. Specifically, we focus on diagonal-norm USBP operators on LGL nodes, which are predominantly used in spectral element methods since they enable splitting techniques [67,30,73] and the extension to variable coefficients including curvilinear coordinates [10,100,70,76]. In many cases, we compare the DG-USBP method with an entropy-stable flux differencing DGSEM on the same LGL points. For brevity, we will henceforth refer to the entropy-stable flux-differencing DGSEM simply as ‘‘DGSEM.’’ The numerical experiments are conducted using the Julia programming language [6]. We employ Runge-Kutta methods from the OrdinaryDiffEq.jl package for time integration [79]. Spatial discretization is handled using the Trixi.jl framework [86] in combination with the SummationByPartsOperators.jl package [81]. Visualization of the results is achieved through Plots.jl [13] and ParaView [4]. All source code required to reproduce the numerical experiments is available online in the reproducibility repository [38].

6.1. Convergence for the linear advection equation

We initiate our study with a convergence analysis of the proposed USBP operators for the one-dimensional scalar linear advection equation

Table 1

Convergence results using DG-USBP discretizations of the linear advection equation with Lax–Friedrichs splitting and J elements. We use degree one (first row), two (second row), and three (third row) USBP operators on three, four, and five LGL nodes per element, respectively, with different values for the artificial dissipation parameters $\lambda_3, \lambda_4,$ and λ_5 .

$\lambda_3 = -10^{-3}$			$\lambda_3 = -10^{-2}$			$\lambda_3 = -10^{-1}$			$\lambda_3 = -1$		
J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC
2	7.41e-01		2	7.38e-01		2	7.47e-01		2	8.14e-01	
4	1.22e-01	2.61	4	1.21e-01	2.61	4	2.09e-01	1.83	4	3.52e-01	1.21
8	1.06e-02	3.53	8	1.05e-02	3.52	8	3.37e-02	2.63	8	6.61e-02	2.41
16	9.75e-04	3.44	16	9.74e-04	3.43	16	5.32e-03	2.66	16	1.08e-02	2.61
32	1.07e-04	3.18	32	1.08e-04	3.17	32	1.02e-03	2.38	32	2.09e-03	2.37
64	1.29e-05	3.06	64	1.34e-05	3.01	64	2.31e-04	2.14	64	4.75e-04	2.14
128	1.60e-06	3.01	128	1.86e-06	2.85	128	5.61e-05	2.04	128	1.15e-04	2.04
$\lambda_4 = -10^{-3}$			$\lambda_4 = -10^{-2}$			$\lambda_4 = -10^{-1}$			$\lambda_4 = -1$		
J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC
2	1.06e-01		2	1.06e-01		2	1.02e-01		2	1.02e-01	
4	4.66e-03	4.51	4	4.64e-03	4.51	4	4.63e-03	4.46	4	4.63e-03	4.46
8	2.73e-04	4.09	8	2.72e-04	4.09	8	2.92e-04	3.99	8	2.92e-04	3.99
16	1.70e-05	4.00	16	1.70e-05	4.00	16	2.30e-05	3.67	16	2.30e-05	3.67
32	1.07e-06	4.00	32	1.08e-06	3.98	32	2.27e-06	3.34	32	2.27e-06	3.34
64	6.66e-08	4.00	64	7.16e-08	3.92	64	2.62e-07	3.12	64	2.62e-07	3.12
128	4.18e-09	4.00	128	5.34e-09	3.74	128	3.21e-08	3.03	128	3.21e-08	3.03
$\lambda_5 = -10^{-3}$			$\lambda_5 = -10^{-2}$			$\lambda_5 = -10^{-1}$			$\lambda_5 = -1$		
J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC
2	6.01e-03		2	6.11e-03		2	7.17e-03		2	1.62e-02	
4	3.04e-04	4.31	4	3.03e-04	4.34	4	3.05e-04	4.56	4	6.42e-04	4.66
8	9.75e-06	4.96	8	9.72e-06	4.96	8	1.10e-05	4.79	8	3.89e-05	4.05
16	3.07e-07	4.99	16	3.08e-07	4.98	16	4.71e-07	4.54	16	2.42e-06	4.01
32	9.62e-09	5.00	32	9.88e-09	4.96	32	2.48e-08	4.25	32	1.51e-07	4.00
64	3.01e-10	5.00	64	3.36e-10	4.88	64	1.47e-09	4.08	64	9.43e-09	4.00
128	9.47e-12	4.99	128	1.34e-11	4.65	128	9.05e-11	4.02	128	5.89e-10	4.00

$$\begin{aligned} \partial_t u(t, x) + \partial_x u(t, x) &= 0, & t \in (0, 5), \quad x \in (-1, 1), \\ u(0, x) &= \sin(\pi x), & x \in [-1, 1], \end{aligned} \tag{6.1}$$

equipped with periodic boundary conditions. The analysis utilizes the classical Lax–Friedrichs flux vector splitting with $\lambda_{\max} = 1$, i.e., $f^-(u) = 0$ and $f^+(u) = u$. For time integration, we employ the nine-stage, fourth-order accurate Runge–Kutta (RK) method of [82] with error-based step size control and a tolerance of 10^{-12} to integrate the semi-discretizations in time. We measure the discrete L^2 error using the quadrature rule associated with the norm matrix P .

Table 1 presents the findings of our convergence study, detailing the discrete L^2 error using the quadrature rule associated with the norm matrix P and the resulting experimental orders of convergence (EOC) for degree one ($d = 1$), degree two ($d = 2$), and three ($d = 3$) USBP operators on three ($N = 3$), four ($N = 4$), and five ($N = 5$) LGL nodes, respectively. These USBP operators are characterized by a non-trivial dissipation matrix $S = V\Lambda V^T$, where V denotes an orthogonal Vandermonde matrix, and $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ encapsulates the dissipation parameters, which regulate the dissipation added to unresolved modes. In the instance of the degree one ($d = 1$) USBP operator on three ($N = 3$) LGL nodes, the parameters λ_1 and λ_2 are set to zero, ensuring that the operator is exact for polynomials up to degree one. Conversely, $\lambda_3 \leq 0$ introduces artificial dissipation to unresolved modes. The impact of different λ_3 values on the accuracy and convergence behavior of the USBP scheme is delineated in the top row of Table 1. Setting λ_3 to zero recovers the central degree three SBP operator on three LGL points, yielding fourth-order convergence. A gradual decrease in λ_3 results in a reduction of the EOC from three to two. Moreover, as we will show, lower values of λ_3 diminish the scheme’s accuracy but enhance its robustness. Regarding the degree two ($d = 2$) USBP operator on four ($N = 4$) LGL nodes, the parameters λ_1 through λ_3 are zeroed, rendering the operator exact for polynomials up to degree two. This time, the parameter $\lambda_4 \leq 0$ adds artificial dissipation to unresolved modes. The second row of Table 1 examines the effects of varying λ_4 on the USBP scheme’s accuracy and convergence. Decreasing λ_4 reduces the EOC from four to three due to adding artificial dissipation to the scheme. For the degree three ($d = 3$) USBP operator on five ($N = 5$) LGL nodes, the parameters λ_1 through λ_4 are zeroed, rendering the operator exact for polynomials up to degree three. The third row of Table 1 reports the effects of varying λ_5 on the USBP scheme’s accuracy and convergence. Decreasing λ_5 reduces the EOC from five to four.

Discussion of the results

The above findings indicate that combining flux vector splitting techniques with DG-USBP operators does not lead to excessive artificial dissipation, similar to high-order FD-USBP schemes.

Table 2

Convergence results using the DGSEM and DG-USBP discretizations of the Euler equations for different numbers of elements J . For the DGSEM, we use polynomial degrees two (first row), three (second row), and four (third row) with the HLL surface flux, and the volume fluxes of Shima et al. and Ranocha. For the DG-USBP method, we use degree one (first row), two (second row), and three (third row) USBP operators with the van Leer–Hänel splitting and different values for the artificial dissipation parameters λ_3 , λ_4 , and λ_5 . For both methods, we use three (first row), four (second row), and five (third row) LGL nodes per element, respectively.

DGSEM, Ranocha flux			DGSEM, Shima flux			$\lambda_3 = -10^{-3}$			$\lambda_3 = -1$		
J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC
2	2.58e-02		2	2.51e-02		2	3.74e-02		2	3.38e-02	
4	4.00e-03	2.69	4	4.06e-03	2.63	4	5.89e-03	2.66	4	9.10e-03	1.89
8	3.75e-04	3.42	8	3.98e-04	3.35	8	6.38e-04	3.21	8	1.64e-03	2.48
16	6.05e-05	2.63	16	5.42e-05	2.88	16	6.90e-05	3.21	16	3.31e-04	2.31
32	8.62e-06	2.81	32	7.13e-06	2.93	32	8.24e-06	3.07	32	7.68e-05	2.11
64	1.14e-06	2.92	64	8.47e-07	3.07	64	1.07e-06	2.95	64	1.88e-05	2.03
128	1.43e-07	3.00	128	1.03e-07	3.04	128	1.51e-07	2.82	128	4.68e-06	2.01
DGSEM, Ranocha flux			DGSEM, Shima flux			$\lambda_4 = -10^{-3}$			$\lambda_4 = -1$		
J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC
2	3.71e-03		2	3.83e-03		2	7.13e-03		2	4.88e-03	
4	2.06e-04	4.17	4	2.47e-04	3.95	4	3.53e-04	4.33	4	5.94e-04	3.04
8	2.79e-05	2.89	8	1.98e-05	3.64	8	3.33e-05	3.41	8	6.81e-05	3.12
16	3.32e-06	3.07	16	1.75e-06	3.50	16	4.49e-06	2.89	16	8.75e-06	2.96
32	2.67e-07	3.64	32	8.62e-08	4.34	32	2.12e-07	4.41	32	1.17e-06	2.91
64	1.36e-08	4.29	64	4.56e-09	4.24	64	9.04e-09	4.55	64	1.48e-07	2.98
128	7.51e-10	4.18	128	2.75e-10	4.05	128	5.17e-10	4.13	128	1.85e-08	3.00
DGSEM, Ranocha flux			DGSEM, Shima flux			$\lambda_5 = -10^{-3}$			$\lambda_5 = -1$		
J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC	J	L^2 error	EOC
2	3.32e-04		2	6.26e-04		2	7.79e-04		2	1.21e-03	
4	1.87e-05	4.15	4	1.80e-05	5.12	4	2.98e-05	4.71	4	6.48e-05	4.22
8	1.73e-06	3.43	8	7.43e-07	4.60	8	1.26e-06	4.57	8	4.81e-06	3.75
16	1.46e-07	3.57	16	3.61e-08	4.36	16	5.31e-08	4.56	16	4.15e-07	3.53
32	6.16e-09	4.57	32	8.69e-10	5.38	32	9.82e-10	5.76	32	1.70e-08	4.61
64	1.25e-10	5.62	64	2.12e-11	5.36	64	3.34e-11	4.88	64	9.50e-10	4.16
128	3.51e-12	5.16	128	9.15e-13	4.53	128	1.41e-12	4.57	128	5.73e-11	4.05

6.2. Convergence for the compressible Euler equations

We continue our study with a convergence analysis for the one-dimensional compressible Euler equations

$$\partial_t \begin{pmatrix} \rho \\ \rho v \\ \rho e \end{pmatrix} + \partial_x \begin{pmatrix} \rho v \\ \rho v^2 + p \\ (\rho e + p)v \end{pmatrix} = 0 \tag{6.2}$$

of an ideal gas with density ρ , velocity v , total energy density ρe , and pressure $p = (\gamma - 1)(\rho e - \rho v^2/2)$, where the ratio of specific heat is chosen as $\gamma = 1.4$. We add a source term to manufacture the solution

$$\rho(t, x) = h(t, x), \quad v(t, x) = 1, \quad \rho e(t, x) = h(t, x)^2, \tag{6.3}$$

with $h(t, x) = 2 + 0.1 \sin(\pi(x - t))$ for $t \in [0, 2]$ and $x \in [0, 2]$. The time-integration is done as in Subsection 6.1, here with a tolerance of 10^{-13} .

Table 2 summarizes the results of our convergence study for the van Leer–Hänel flux vector splitting [105,41,56]. We obtained similar results for the Steger–Warming flux vector splitting [97], which are not reported here but can be reproduced using the openly available code repository [38]. We use degree one (first row), two (second row), and three (third row) USBP operators on three, four, and five LGL nodes per element, respectively, with different values for the artificial dissipation parameters λ_3 , λ_4 , and λ_5 . Similar to the convergence test for the linear advection equation in Subsection 6.1, we observe that the EOC reduces by one as the respective dissipation parameter λ_N is gradually decreased from -10^{-3} to -1 for each operator. Furthermore, we compare the performance of the DG-USBP method with a DGSEM employing the HLL surface flux [42] in combination with the volume fluxes of Shima et al. [94] and Ranocha [80]. To allow for a fair comparison, we use polynomial degrees two (first row), three (second row), and four (third row) on the same three (first row), four (second row), and five (third row) LGL nodes per element as the DG-USBP method. That is, we use the same set of points for both the DGSEM and the DG-USBP schemes.

Discussion of the results

Based on the above results, we observe that the accuracies of the DGSEM and DG-USBP approaches are comparable, particularly for small values of the parameter λ , although the DGSEM consistently yields more accurate results. We suspect that the reduction in

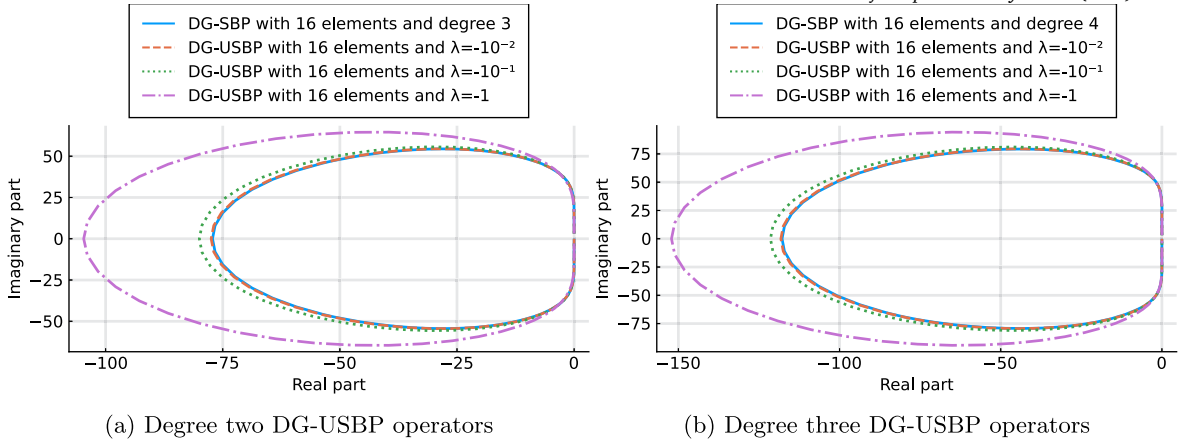


Fig. 1. Spectra of the DG-USBP and the DGSEM semi-discretizations with $J = 16$ elements. We use degree two ($d = 2$, Fig. 1a) and three ($d = 3$, Fig. 1b) DG-USBP operators on four ($N = 4$) and five ($N = 5$) LGL nodes, respectively, with different parameters λ_4 and λ_5 . For the DGSEM method, we use traditional degree three and four DG-SBP operators on the same LGL nodes as the respective DG-USBP operators.

accuracy for the DG-USBP method is due to the additional artificial dissipation introduced. Therefore, while combining flux vector splitting techniques with DG-USBP operators does not necessarily lead to excessive artificial dissipation, the accuracy of the DG-USBP method remains lower than that of the DGSEM considered here.

6.3. Spectral analysis

Next, we compare the spectra of the DG-USBP to the DGSEM and FD-USBP semi-discretization for the linear advection equation (6.1) with periodic boundary conditions. Fig. 1 illustrates the spectra of the DG-USBP semi-discretization for degree two ($d = 2$) and three ($d = 3$) DG-USBP operators on four ($N = 4$) and five ($N = 5$) LGL nodes, respectively, with $J = 16$ elements and different parameters λ_4 and λ_5 . For λ_4 and λ_5 close to zero, the DG-USBP spectra are comparable to the DGSEM method (labeled “DG-SBP”) on the same LGL points. At the same time, for smaller λ_4 and λ_5 , the spectra suggest that the DG-USBP method is stiffer than the DGSEM method.

Discussion of the results

We find that the DG-USBP method is stiffer than the DGSEM method on the same LGL points in all cases. This could be explained by the DG-USBP method having more artificial dissipation than the DGSEM method. A drawback of increasing stiffness when using DG-USBP operators is that explicit time-stepping may require smaller time steps, which can impact the method’s overall performance.

6.4. Local linear/energy stability

We now demonstrate the local linear/energy stability properties of the proposed DG-USBP operators for the inviscid Burgers’ equation

$$\partial_t u(t, x) + \partial_x [u(t, x)^2/2] = 0, \quad x \in (-1, 1), \quad (6.4)$$

with periodic boundary conditions. To this end, we linearize (6.4) locally around a baseflow $\tilde{u}(x)$, so that $u(t, x) = \tilde{u}(x) + v(t, x)$, which yields

$$\partial_t v(t, x) + \partial_x [\tilde{u}(x)v(t, x)] = 0, \quad x \in (-1, 1). \quad (6.5)$$

Observe that (6.5) is a linear advection equation with a spatially varying coefficient $\tilde{u}(x)$ that is solved for the perturbation component $v(t, x)$. Notably, when the baseflow $\tilde{u}(x)$ is positive across $[-1, 1]$, the operator $v \mapsto \partial_x [\tilde{u}v]$ becomes skew-symmetric for a weighted L^2 inner product. As a result, this operator exhibits a purely imaginary spectrum [59,29]. A semi-discretization should replicate this behavior, ensuring a linearization around a positive baseflow yields eigenvalues with non-positive real parts. However, it was observed in [29] that existing entropy-stable high-order flux differencing DGSEM semi-discretizations can produce a linearized discrete operator that has eigenvalues with a significant positive real part. Now consider the full-upwind DG-USBP semi-discretization,

$$\frac{d}{dt} \mathbf{u}^j + D_- \mathbf{f}^j = \text{SAT}^j, \quad j = 1, \dots, J, \quad (6.6)$$

where J is the number of elements and $\mathbf{f}^j = \frac{1}{2} \mathbf{u}^j \odot \mathbf{u}^j$ with \odot denoting the usual element-wise Hadamard product. Our numerical investigations demonstrate that the full-upwind DG-USBP semi-discretization, as per (6.6), results in a linearized discrete operator

Table 3

The maximum real parts of the spectrum of the DG-USBP method for the Burgers equation using a full-upwind semi-discretization. We use degree one ($d = 1$), two ($d = 2$), and three ($d = 3$) USBP operators on three ($N = 3$), four ($N = 4$), and five ($N = 5$) LGL nodes per element, respectively, varying numbers of elements J , and different values for the artificial dissipation parameter λ_3, λ_4 , and λ_5 .

Degree one operator				Degree two operator				Degree three operator			
$J \setminus \lambda_3$	0	-10^{-2}	-1	$J \setminus \lambda_4$	0	-10^{-2}	-1	$J \setminus \lambda_5$	0	-10^{-2}	-1
2	-1.5e-18	1.2e-16	9.9e-17	2	6.7e-16	4.9e-17	-3.3e-16	2	5.8e-16	-8.4e-16	7.8e-17
4	-2.7e-16	-6.2e-16	2.9e-17	4	-4.9e-16	6.0e-16	5.5e-17	4	1.5e-17	-5.3e-16	4.7e-16
8	4.3e-16	-4.0e-16	6.5e-16	8	-5.8e-16	2.2e-15	-4.1e-17	8	1.0e-15	3.6e-16	-1.8e-16
16	-9.9e-16	2.3e-16	-3.6e-15	16	3.3e-17	-1.1e-15	-3.5e-16	16	-7.0e-16	-7.4e-15	-1.7e-15
32	7.1e-18	8.8e-15	1.1e-14	32	1.0e-14	-4.3e-14	3.9e-15	32	2.0e-15	-3.5e-16	-3.2e-15

whose eigenvalues have non-positive real parts, by the analysis of [87]. To stress the method, we consider a completely under-resolved case by computing the Jacobian at a random non-negative state using automatic differentiation via ForwardDiff.jl [90]. Table 3 reports the maximum real parts of the spectrum of the DG-USBP method for the linearized Burgers equation (6.5). This is done using degree one ($d = 1$), two ($d = 2$), and three ($d = 3$) DG-USBP operators on three ($N = 3$), four ($N = 4$), and five ($N = 5$) LGL nodes per element, respectively, varying the number of elements J , and different values for the artificial dissipation parameters λ_3, λ_4 , and λ_5 . In all cases, the maximum real part of the eigenvalue is around the machine precision level, aligning with our desired outcome.

Discussion fo the results

We find that DG-USBP methods yield local linear/energy stability. This finding contrasts the behavior observed for existing entropy-stable high-order flux differencing DGSEM semi-discretizations in the same setting. Parallel findings were obtained in [87] for traditional FD-USBP operators, complemented by a theoretical analysis.

6.5. Curvilinear meshes and free-stream preservation

One particular strength of DG-type methods is their geometric flexibility and ability to create high-order approximations on curvilinear, unstructured meshes. This geometric flexibility is retained with the DG-USBP operators developed in the present work, which we subsequently demonstrate in a two-dimensional setting.

The curvilinear DG-USBP semi-discretization

We start by describing conservation laws in curvilinear coordinates and their DG-USBP semi-discretization. See Kopriva [47] for a complete discussion on curvilinear coordinates and transformations. Recall that a conservation law in two dimensions is of the form

$$\partial_t u + \partial_x f_1(u) + \partial_y f_2(u) = 0, \quad t \in (0, T), (x, y) \in \Omega \subset \mathbb{R}^2, \quad (6.7)$$

with fluxes f_1, f_2 , and conserved variable $u = u(t, x, y)$. Suppose the domain Ω has been subdivided into J non-overlapping quadrilateral elements, $\Omega_j, j = 1, \dots, J$. For ease of notation, we subsequently consider the conservation law (6.7) on an individual element Ω_j and suppress the index j . The element Ω_j in the physical coordinates (x, y) is transformed into a reference element $\Omega_{\text{ref}} = [-1, 1]^2$ in the computational coordinates (ξ, η) via the coordinate transformation

$$x = X(\xi, \eta), \quad y = Y(\xi, \eta). \quad (6.8)$$

Under this transformation, the conservation law in the physical coordinates (6.7) becomes a conservation law in the reference coordinates:

$$J \partial_t u + \partial_\xi \tilde{f}_1(u) + \partial_\eta \tilde{f}_2(u) = 0, \quad t \in (0, T), (\xi, \eta) \in \Omega_{\text{ref}}, \quad (6.9)$$

with $u = u(t, \xi, \eta)$. The contravariant fluxes are $\tilde{f}_1 = Y_\eta f_1 - X_\eta f_2$ and $\tilde{f}_2 = -Y_\xi f_1 + X_\xi f_2$, while the Jacobian is $J = X_\xi Y_\eta - X_\eta Y_\xi$. The partial derivatives X_ξ, X_η, Y_ξ , and Y_η , are called the *metric terms*. The metric terms in DG approximations are typically created from a transfinite interpolation with linear blending [47,40]. The curvilinear DG-USBP semi-discretization of (6.9) has the form

$$J \partial_t \mathbf{u} + D_- \tilde{\mathbf{f}}_1^+ + D_+ \tilde{\mathbf{f}}_1^- + \tilde{\mathbf{f}}_2^+ D_-^T + \tilde{\mathbf{f}}_2^- D_+^T = \widetilde{\text{SAT}}^j. \quad (6.10)$$

Note that multiplication from the left with D_\pm approximates the derivative in the ξ -direction and multiplication from the right with D_\pm^T approximates the derivative in the η -direction. Furthermore, we use a generic statement of the SAT in the normal direction on an interface in element j

$$\widetilde{\text{SAT}}^j = -P^{-1} \mathbf{e}_R \left(f^{\text{num}}(\mathbf{u}_R^j, \mathbf{u}_L^{j+1}; \hat{\mathbf{n}}) - \tilde{\mathbf{f}}_R^j \right) + P^{-1} \mathbf{e}_L \left(f^{\text{num}}(\mathbf{u}_R^{j-1}, \mathbf{u}_L^j; \hat{\mathbf{n}}) - \tilde{\mathbf{f}}_L^j \right), \quad (6.11)$$

where $\hat{\mathbf{n}}$ is the normal direction on a particular element. The tilde notation indicates that the physical fluxes are computed in the contravariant direction.

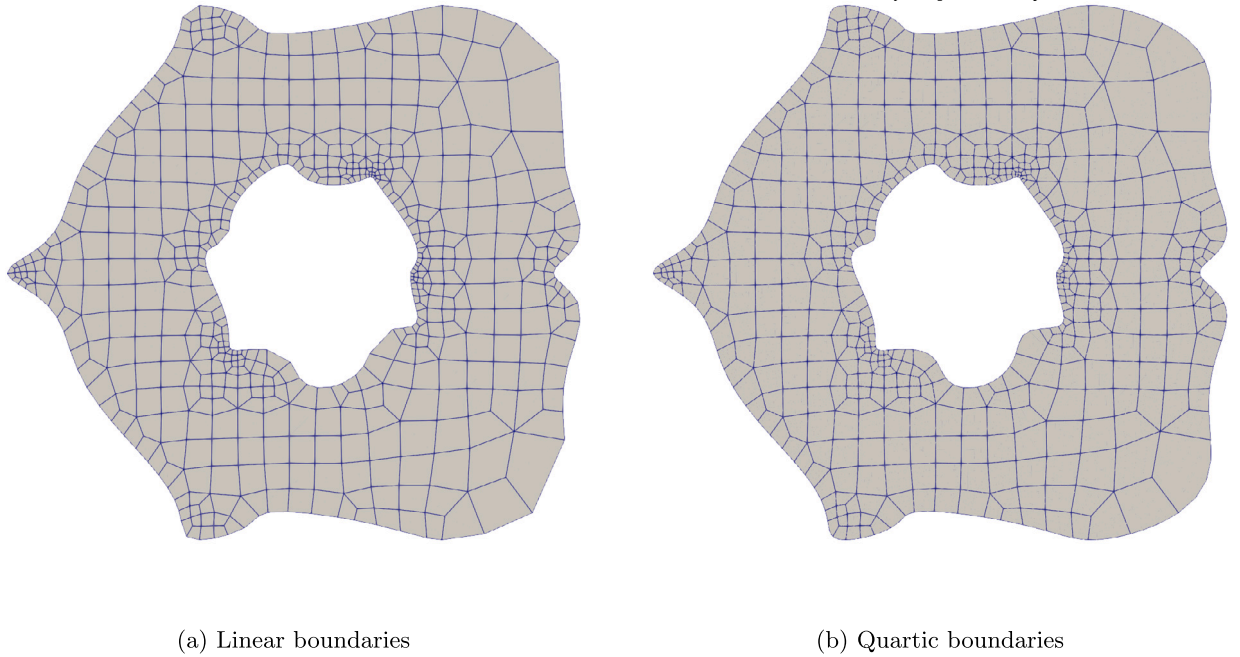


Fig. 2. Meshes of 550 non-overlapping quadrilaterals with linear and quartic boundary polynomial order for the FSP test. Observe—see the lower right corner—that the boundary is approximated more accurately for the higher (quartic) boundary polynomial.

Free-stream preservation

Notably, the metric terms satisfy the two metric identities

$$\partial_{\xi} Y_{\eta} - \partial_{\eta} Y_{\xi} = 0, \quad \partial_{\xi} X_{\eta} - \partial_{\eta} X_{\xi} = 0, \tag{6.12}$$

which are essential to ensure *free-stream preservation (FSP)*; see [108,107,47]. That is, given a constant flux in space, its divergence vanishes, and the constant solution of (6.7) does not change in time. As demonstrated in [87], the ability of existing FD-USBP methods to remain FSP on curvilinear meshes is delicate. There is a subtle relationship between the boundary closure accuracy of an FD-SBP operator, the dependency of flux vector splitting on the metric terms, and the polynomial degree of curved boundaries. The boundary closure of an FD-SBP operator relates to the highest polynomial degree the operator can differentiate exactly. For instance, a 4-2 FD-SBP operator is fourth-order accurate in the interior with second-order boundary closures and can differentiate up to quadratic polynomials exactly. In the following discussion, we use the term boundary closure in the DG context to refer to the highest polynomial degree a discrete derivative operator can differentiate exactly.

The curvilinear flux vector splittings \mathbf{f}_1^{\pm} and \mathbf{f}_2^{\pm} in (6.10) implicitly depend on the metric terms [5,87]. The polynomial dependency of the curvilinear flux vector splitting on the metric terms must “agree” with the boundary order closure of a USBP operator. Precisely, suppose the polynomial dependency of a mapped flux vector splitting on the metric terms has order m . In that case, the USBP operator must have a boundary closure that can differentiate polynomials up to degree mN_{geo} exactly, where N_{geo} is the polynomial order of the curved boundaries. For example, the Lax–Friedrichs splitting is linear in the metric terms while the van Leer–Hänel splitting is quadratic in the metric terms. This means that $m = 1$ for the Lax–Friedrichs splitting and the curvilinear approximation remains FSP regardless of the boundary closure order. In contrast, the van Leer–Hänel splitting has $m = 2$ (quadratic dependency on the metric terms), which restricts the possible values of N_{geo} combined with the boundary closure of the SBP operator.

Remark 6.1. To create DG-USBP operators for polynomials of degree d , we sacrifice one degree of accuracy on the boundary closure to construct the internal dissipation matrix S , i.e., we use $N = d + 2$ LGL points. A “classic” DG-SBP operator with $N = 4$ LGL points has a boundary closure of the order $d = 3$ and can differentiate up to cubic polynomials exactly. In contrast, a DG-USBP operator for $N = 4$ LGL points will be endowed with a boundary closure of degree $d = 2$ and can only differentiate up to quadratic polynomials exactly.

Computational results

We now numerically investigate how the type of flux vector splitting in curvilinear coordinates combined with the DG-USBP operator influences the FSP property. To this end, we consider an unstructured curvilinear mesh with varying N_{geo} values and present the errors for an FSP problem for different flux vector splittings and operator orders. Specifically, we consider a domain with curvilinear outer and inner boundaries of different geometric polynomial degrees, generated in Julia with HOHQMesh.jl [48]. The

Table 4

FSP error for the Lax–Friedrichs splitting. The splitting depends linearly on the metric terms and satisfies the FSP property for all approximation and mesh orders.

interior order	2	3	4	5	6	7	8	9
bi-linear mesh	2.92×10^{-14}	9.96×10^{-15}	1.06×10^{-14}	2.08×10^{-14}	1.83×10^{-14}	3.76×10^{-14}	4.12×10^{-14}	5.96×10^{-14}
quadratic mesh	1.19×10^{-14}	1.12×10^{-14}	5.53×10^{-14}	1.93×10^{-14}	1.81×10^{-14}	4.12×10^{-14}	3.89×10^{-14}	2.84×10^{-14}
cubic mesh	5.52×10^{-14}	1.45×10^{-14}	7.49×10^{-15}	1.92×10^{-14}	2.05×10^{-14}	6.67×10^{-14}	4.56×10^{-14}	5.67×10^{-14}
quartic mesh	2.29×10^{-15}	8.95×10^{-15}	6.03×10^{-15}	2.00×10^{-14}	2.16×10^{-14}	5.83×10^{-14}	2.37×10^{-14}	4.81×10^{-14}

Table 5

FSP error for the van Leer–Hänel splitting. The splitting depends quadratically on the metric terms and satisfies the FSP property if the boundary closure accuracy of the DG-USBP operator is not smaller than $2N_{\text{geo}}$.

interior order	2	3	4	5	6	7	8	9
bi-linear mesh	1.21×10^{-5}	4.44×10^{-15}	6.49×10^{-15}	1.20×10^{-14}	1.40×10^{-14}	3.85×10^{-14}	5.31×10^{-13}	9.56×10^{-13}
quadratic mesh	1.38×10^{-5}	1.43×10^{-6}	1.81×10^{-7}	1.11×10^{-14}	1.51×10^{-14}	1.15×10^{-14}	2.22×10^{-13}	7.56×10^{-13}
cubic mesh	1.37×10^{-5}	1.56×10^{-6}	2.28×10^{-7}	2.40×10^{-8}	2.42×10^{-9}	3.09×10^{-14}	3.75×10^{-13}	9.19×10^{-13}
quartic mesh	1.38×10^{-5}	1.57×10^{-6}	3.93×10^{-7}	4.21×10^{-8}	1.05×10^{-8}	9.93×10^{-10}	2.21×10^{-10}	9.87×10^{-13}

domain is illustrated in Fig. 2 for linear and quartic boundaries. The curvilinear domain is a heavily warped annulus-type domain with the outer and inner curved boundaries given by

$$\begin{aligned} \text{outer} &= \begin{cases} x(t) = 4 \cos(2\pi t) - 0.6 \cos^3(8\pi t) \\ y(t) = 4 \sin(2\pi t) - 0.5 \sin(10\pi t) \end{cases} \\ \text{inner} &= \begin{cases} x(t) = 1.5 \cos(2\pi t) - 0.2 \cos^2(8\pi t) \\ y(t) = 1.5 \sin(2\pi t) - 0.2 \sin(11\pi t) \end{cases} \end{aligned} \tag{6.13}$$

with the parameter $t \in [0, 1]$. The domain is divided into 550 non-overlapping quadrilateral elements. The domain boundaries are highly nonlinear, and we compute results on four versions of the approximate mesh boundaries: bi-linear boundary elements ($N_{\text{geo}} = 1$), quadratic boundary elements ($N_{\text{geo}} = 2$), cubic boundary elements ($N_{\text{geo}} = 3$), and quartic boundary elements ($N_{\text{geo}} = 4$). Fig. 2 shows the bilinear and quartic boundary meshes to compare the resolution of the curved boundary elements as N_{geo} increases. Furthermore, we consider the two-dimensional compressible Euler equations on the curvilinear domain illustrated in Fig. 2 with a constant solution state given in the conservative variables as

$$u_\infty = \begin{pmatrix} \rho_\infty \\ (\rho v_1)_\infty \\ (\rho v_2)_\infty \\ (\rho e)_\infty \end{pmatrix} = \begin{pmatrix} 1.0 \\ 0.1 \\ -0.2 \\ 10.0 \end{pmatrix}, \tag{6.14}$$

where the compressible Euler fluxes are all constant, and their divergence vanishes. Importantly, this is not necessarily the case for the numerical solution. To demonstrate this, we examine the FSP property for the Lax–Friedrichs and van Leer–Hänel splittings for different values of N_{geo} and varying orders of the proposed DG-USBP operators. See Tables 4 and 5, respectively. In all cases, the final time of all the FSP tests is 10 and we impose Dirichlet boundary conditions via the background solution state (6.14).

Discussion of the results

In accordance with [87], the Lax–Friedrichs splitting, depending linearly on the metric terms, satisfies the FSP property for all considered orders. In contrast, the van Leer–Hänel splitting, depending quadratically on the metric terms, only satisfies the FSP property if the boundary closure of the DG-USBP operator can exactly differentiate polynomials up to order $mN_{\text{geo}} = 2N_{\text{geo}}$. Finally, our results in Table 5 imply that more complex flux vector splittings require higher boundary closure accuracy to ensure the boundary is sufficiently accurate.

6.6. Isentropic vortex

Next, we investigate the long-term stability of the DG-USBP method. To this end, we follow [96] and consider the classical isentropic vortex test case of [95] for the two-dimensional Euler equations with initial conditions

$$\begin{aligned} T &= T_0 - \frac{(\gamma - 1)\epsilon^2}{8\gamma\pi^2} \exp(1 - r^2), \\ \rho &= \rho_0 (T/T_0)^{1/(\gamma-1)}, \\ v &= v_0 + \frac{\epsilon}{2\pi} \exp((1 - r^2)/2) (-x_2, x_1)^T. \end{aligned} \tag{6.15}$$

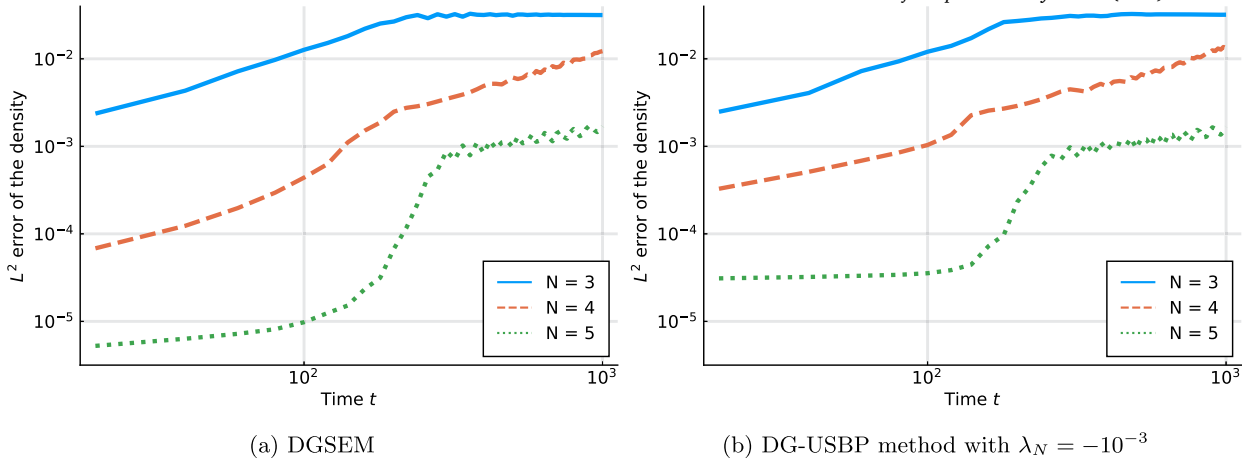


Fig. 3. Density error for the isentropic vortex test case of the DG-USBP and DGSEM method for $J = 256$ elements. We use degree one ($d = 1$, blue dashed line), two ($d = 2$, orange dotted line), and three ($d = 3$, green dash-dotted line) tensor-product DG-USBP operators on $N = 3, 4, 5$ LGL nodes per coordinate direction and element. For the DGSEM method, we use traditional degree two, three, and four DG-SBP operators on the same LGL nodes as the DG-USBP operators.

Here, $\epsilon = 10$ is the vortex strength, r is the distance from the origin, $T = p/\rho$ the temperature, $\rho_0 = 1$ the background density, $v_0 = (1, 1)^T$ the background velocity, $p_0 = 10$ the background pressure, $\gamma = 1.4$, and $T_0 = p_0/\rho_0$ the background temperature. The domain $[-5, 5]^2$ is equipped with periodic boundary conditions. We use the same time integration method and approach to compute the discrete L^2 error of the density as in Subsection 6.2.

Fig. 3 illustrates the discrete L^2 -error of the density for long-time simulations for the DG-USBP method with Steger–Warming splitting and the DGSEM using the volume flux of Ranocha. We use three ($N = 3$, blue straight line), four ($N = 4$, orange dashed line), and five ($N = 5$, green dotted line) LGL nodes per coordinate direction and element. As before, the DG-USBP and DGSEM methods use the same points. Fig. 3 report these results for $J = 256$ elements and $\lambda_N = -10^{-3}$. The remaining parameters are zero to preserve exactness for polynomials up to degree $d = N - 2$. We observe that both the DG-USBP and DGSEM methods remain stable and can run the simulations successfully for a long time. Notably, the two methods perform roughly the same for three LGL nodes. For four and five LGL nodes, however, we observe the DGSEM method to initially produce smaller density errors than the DG-USBP method. In all cases, the two methods perform roughly the same towards the end of the simulation, indicating a comparable long-time behavior.

Discussion of the results

The above results indicate that USBP operators offer less improvement in robustness for DG methods compared to FD schemes.

6.7. Kelvin–Helmholtz instability

The above results for the isentropic vortex test (see Subsection 6.6) indicated that USBP operators offer less improvement in robustness for DG methods compared to FD schemes. To further investigate this hypothesis, we consider the Kelvin–Helmholtz instability setup for the two-dimensional compressible Euler equations of an ideal fluid as in [92]. The corresponding initial condition is

$$\rho = \frac{1}{2} + \frac{3}{4}B(x, y), \quad p = 1, \quad v_1 = \frac{1}{2}(B(x, y) - 1), \quad v_2 = \frac{1}{10} \sin(2\pi x), \quad (6.16)$$

where $B(x, y) = \tanh(15y + 7.5) - \tanh(15y - 7.5)$ is a smoothed approximation to a discontinuous step function. The domain is $[-1, 1]^2$ with time interval $[0, 15]$. We perform time integration of the semi-discretizations using the third-order, four-stage SSP method detailed in [49]. This is coupled with the embedded method proposed in [17] and an error-based step size controller as in [82]. The adaptive time step controller tolerances are set to 10^{-6} . Here, we compare the DG-USBP method outlined in this paper with the DGSEM, which employs various volume fluxes and a local Lax–Friedrichs (Rusanov) surface flux.

Table 6 presents the final simulation times for various numerical experiments of the Kelvin–Helmholtz instability. Final times below 15 signify that the simulation terminated prematurely. For the tensor-product DG-USBP and DGSEM methods, we use the same three ($N = 3$), four ($N = 4$), and five ($N = 5$) LGL nodes per coordinate direction and element. To preserve exactness for polynomials up to degree $d = N - 2$, we have chosen the other dissipation parameters as zero, i.e., $\lambda_1 = \dots = \lambda_{N-1} = 0$, and test different parameters λ_N . A notable observation is that reducing the parameters λ_N in the DG-USBP method often leads to an extension in the final time of the corresponding simulations. This trend suggests an enhancement in the method’s robustness, although the difference decreases with an increased number of elements. Specifically, for three LGL points ($N = 3$) and sufficiently small λ_3 , we observe the DG-USBP method to finish the simulation in all cases, while the DGSEM crashed for most of them. At the same time, we observe that the DG-USBP method offers less improvement in robustness as the number of LGL points (N) increases. For instance, the DGSEM with

Table 6

Final times for the numerical simulations of the Kelvin–Helmholtz instability with J elements. Final times less than 15 indicate that the simulation crashed. We use the same three ($N = 3$), four ($N = 4$), and five ($N = 5$) LGL nodes per coordinate direction and element for the tensor-product DG-USBP and DGSEM method. We compare different parameters λ_N , splittings, and numerical fluxes. The highest final times for each combination of J and N are highlighted in bold.

$N = 3$					$N = 4$					$N = 5$				
J	16	64	256	1024	J	16	64	256	1024	J	16	64	256	1024
DG-USBP, van Leer–Hänel splitting					DG-USBP, van Leer–Hänel splitting					DG-USBP, van Leer–Hänel splitting				
λ_3	final times				λ_4	final times				λ_5	final times			
-10^{-3}	15.00	4.93	1.89	2.00	-10^{-3}	4.57	10.73	1.60	1.85	-10^{-3}	1.38	2.91	1.48	3.62
-10^{-2}	15.00	15.00	1.84	4.41	-10^{-2}	4.58	10.55	1.61	1.80	-10^{-2}	1.39	2.92	1.51	3.62
-10^{-1}	15.00	15.00	15.00	15.00	-10^{-1}	15.00	15.00	4.82	3.48	-10^{-1}	1.52	6.23	3.69	3.61
DG-USBP, Steger–Warming splitting					DG-USBP, Steger–Warming splitting					DG-USBP, Steger–Warming splitting				
λ_3	final times				λ_4	final times				λ_5	final times			
-10^{-3}	15.00	4.85	1.96	1.85	-10^{-3}	4.64	1.68	1.61	1.73	-10^{-3}	1.27	2.59	1.51	3.62
-10^{-2}	15.00	15.00	1.88	4.30	-10^{-2}	4.62	1.78	1.62	1.81	-10^{-2}	1.29	2.88	1.52	3.64
-10^{-1}	15.00	15.00	15.00	15.00	-10^{-1}	15.00	1.91	4.78	3.49	-10^{-1}	1.47	5.65	3.67	3.60
DGSEM					DGSEM					DGSEM				
flux	final times				flux	final times				flux	final times			
Shima	15.00	2.92	3.25	3.03	Shima	2.73	1.38	2.82	2.88	Shima	1.81	3.05	3.29	3.36
Ranocha	15.00	4.68	4.81	4.12	Ranocha	4.46	1.53	3.77	3.66	Ranocha	2.47	4.04	4.44	4.27

the Ranocha flux yields the longest run times in almost all cases for $N = 5$. Once more, this highlights that USBP operators offer less improvement in robustness for high-order DG methods than for FD schemes.

Discussion of the results

We observe in Table 6 that combining lower-order DG-USBP with flux vector splittings can increase the robustness of nodal DG methods. Specifically, for three LGL points ($N = 3$) and sufficiently small λ_3 , we observe the DG-USBP method to finish the simulation in all cases, while the DGSEM crashed for most of them. At the same time, we observe that the DG-USBP method offers less improvement in robustness as the number of LGL points (N) increases. For instance, the DGSEM with the Ranocha flux yields the longest run times in almost all cases for $N = 5$. Once more, this highlights that USBP operators offer less improvement in robustness for high-order DG methods than for FD schemes. This reduced improvement could be attributed to USBP methods adding dissipation only to unresolved modes. After all, FD schemes typically have more unresolved modes than nodal DG methods.

Density profiles at crash time

Fig. 4 compares the density profiles from simulations of the Kelvin–Helmholtz instability captured at the crash time. These simulations were conducted on $J = 1024$ elements using the DG-USBP method and the DGSEM. For both the DG-USBP method and DGSEM, we used four ($N = 4$) LGL nodes per coordinate direction and element. That is, all simulations were performed with the same number of DOFs, ensuring a consistent comparison base. Furthermore, we used $\lambda_4 = -10^{-1}$ for the DG-USBP method and the volume fluxes of Shima et al. and Ranocha for the DGSEM. In Fig. 4, the white spots indicate locations where non-positive values are observed for pressure (in the case of DG-USBP) or density (for DGSEM). Notably, these problematic nodes consistently occur at the interfaces between elements. Interestingly, the same was reported for non-periodic FD-USBP operators in [87], suggesting a general trend. We also observed this phenomenon for other simulations with different numbers of elements and nodes per element.

6.8. *The inviscid Taylor–Green vortex*

Recall that previous works [62,63,58,98,87,24] have demonstrated that combining high-order USBP operators with flux vector splittings can enhance the robustness of FD schemes without introducing excessive artificial dissipation. In this work, we have investigated whether combining USBP operators with flux vector splittings can be applied to nodal DG methods with similar success. However, the numerical tests presented above demonstrated both advantages and disadvantages for DG-USBP schemes compared to existing high-order DGSEM methods. On the one hand, we observed that DG-USBP methods do not necessarily introduce excessive artificial dissipation (Subsections 6.1 and 6.2), yield local linear/energy stability in a setting where existing high-order entropy-stable flux differencing DGSEM produce unstable semi-discretizations (Subsection 6.4), and extend to curvilinear while ensuring FSP (Subsection 6.5). On the other hand, we found that DG-USBP schemes are often less accurate than existing high-order DGSEM (Subsections 6.2 and 6.6), generally yield stiffer semi-discretizations (Subsection 6.3), and do not always offer improvements in robustness (Subsection 6.7)—which contrasts with observations for FD schemes. Specifically, in simulating the Kelvin–Helmholtz instability in Subsection 6.7, we observed that USBP operators improve robustness for lower-order operators with three LGL points. Still, this improvement vanishes when higher-order operators with four and five LGL points are used. Recall that USBP operators add artificial dissipation only to the unresolved modes. We thus suspect that the reduced improvement can be explained by the ratio $1/(N - 1)$

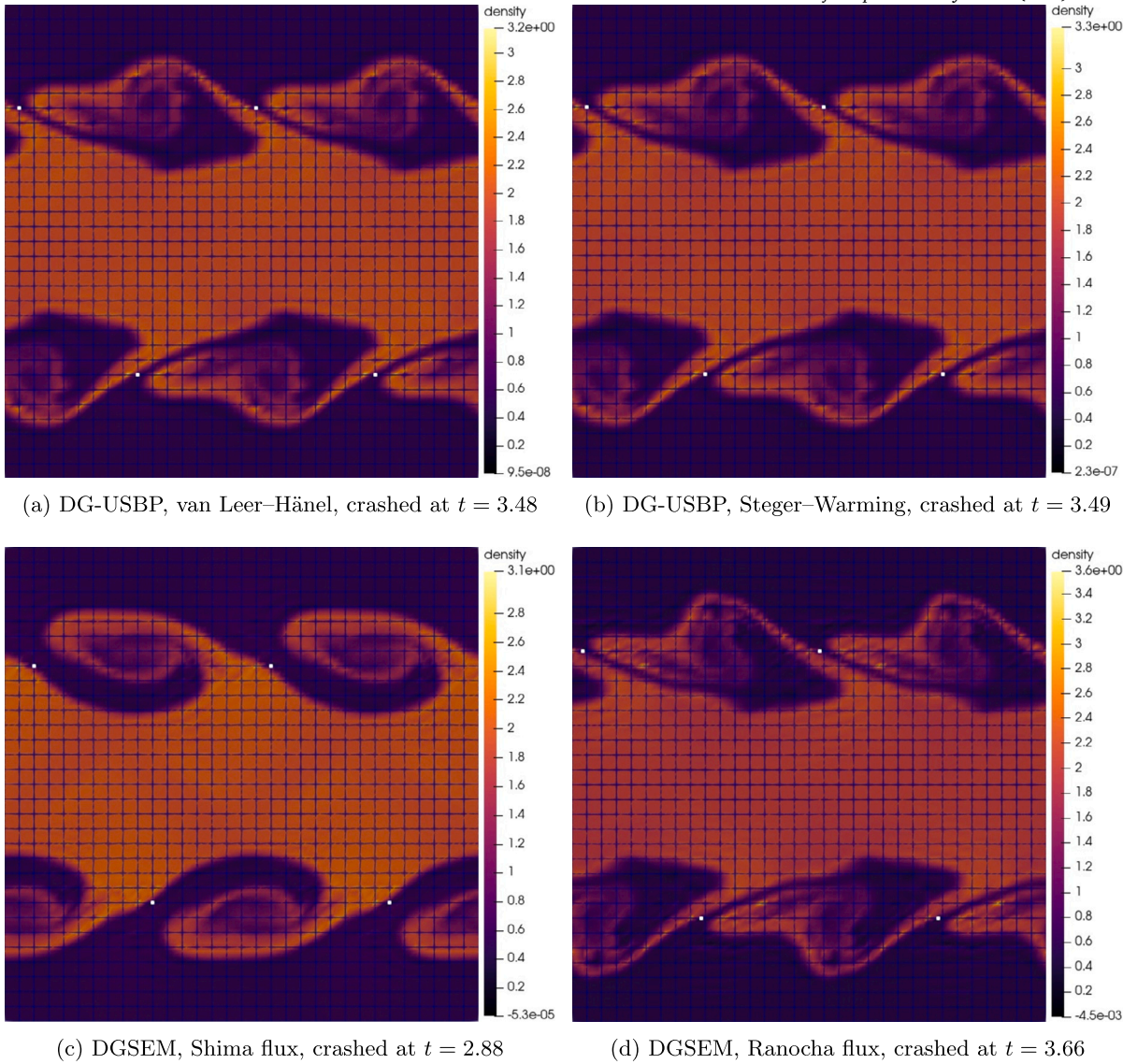


Fig. 4. Density profile for the DG-USBP method and DGSEM on $J = 1024$ elements when the Kelvin-Helmholtz instability simulations crashed. Both methods use six LGL nodes per coordinate direction and element. Hence, all simulations use the same number of DOFs. Furthermore, the DG-USBP methods use $\lambda_5 = -10^{-1}$. The white spots mark points where the pressure (DG-USBP) or the density (DGSEM) is non-positive.

between the number of unresolved modes,³ which is 1, and the number of resolved modes, which is $N - 1$, decreasing as more LGL nodes are used.

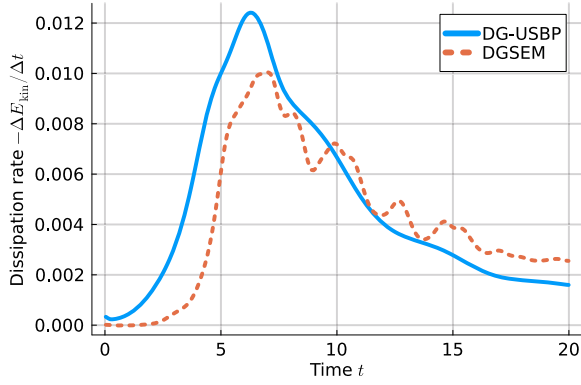
To shed additional light on these observations, we investigate the kinetic energy dissipation properties of the DG-USBP method, following the approach of [30]. To this end, we consider the inviscid Taylor-Green vortex test case for the three-dimensional compressible Euler equations of an ideal gas. The corresponding initial condition is

$$\rho = 1, \quad p = \frac{\rho}{\text{Ma}^2 \gamma} + \rho \frac{\cos(2x_1) \cos(2x_3) + 2 \cos(2x_2) + 2 \cos(2x_1) + \cos(2x_2) \cos(2x_3)}{16}, \quad (6.17)$$

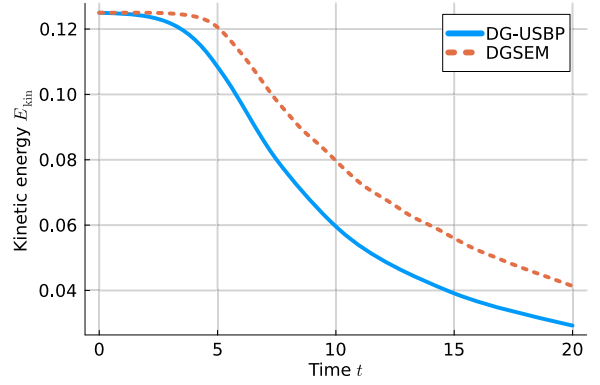
$$v_1 = \sin(x_1) \cos(x_2) \cos(x_3), \quad v_2 = -\cos(x_1) \sin(x_2) \cos(x_3), \quad v_3 = 0,$$

with Mach number $\text{Ma} = 0.1$. The domain is $[-\pi, \pi]^3$ with periodic boundary conditions and the time interval is $[0, 20]$. Time integration is performed as in Subsection 6.7.

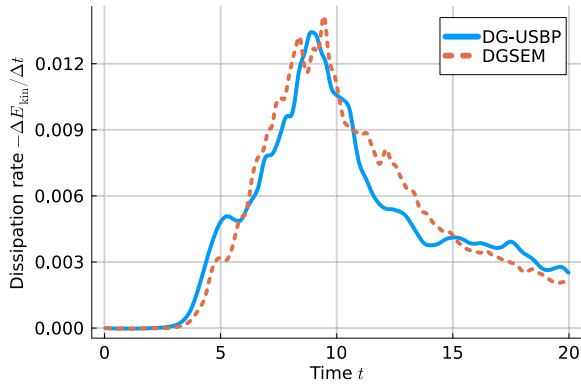
³ The DG-USBP operators considered here add artificial dissipation only to the highest mode.



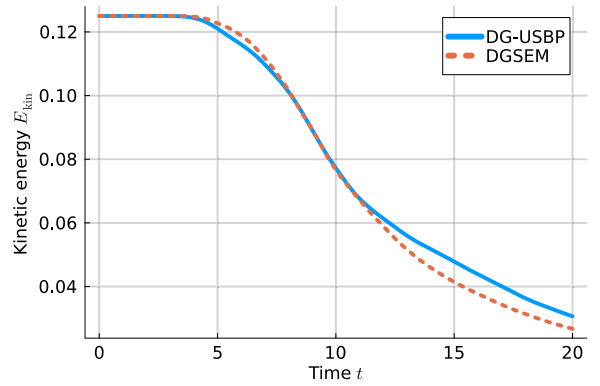
(a) Dissipation rate for $N = 3$



(b) Kinetic energy for $N = 3$



(c) Dissipation rate for $N = 4$



(d) Kinetic energy for $N = 4$

Fig. 5. Discrete kinetic energy and dissipation rate for the inviscid Taylor–Green vortex. We compare the results of a DG-USBP method with the DGSEM. Both methods use $J = 4096$ elements with three ($N = 3$, top row) and four ($N = 4$, bottom row) LGL points per element and coordinate direction. The DGSEM uses the entropy-conservative volume flux of Ranocha and the local Lax-Friedrichs (Rusanov) surface flux. The DG-USBP method uses the Steger–Warming splitting and $\lambda_N = -10^{-1}$.

Fig. 5 depicts the discrete kinetic energies and their dissipation rates of a DG-USBP method with a DGSEM. Both methods use $J = 4096$ elements with three ($N = 3$, Figs. 5a and 5b) and four ($N = 4$, Figs. 5c and 5d) LGL points per element and coordinate direction. The DGSEM uses the entropy-conservative volume flux of Ranocha and the local Lax-Friedrichs (Rusanov) surface flux. The DG-USBP method uses the Steger–Warming splitting and $\lambda_N = -10^{-1}$. The discrete version of the total kinetic energy is computed as

$$E_{\text{kin}}(t) = \int \frac{1}{2} \rho(t, x) v(t, x)^2 dx \tag{6.18}$$

using the quadrature rule associated with the SBP norm matrix P at every tenth accepted time step. Central finite differences are then employed to calculate the discrete kinetic energy dissipation rate, $-\Delta E_{\text{kin}}/\Delta t$, approximating $-dE_{\text{kin}}/dt$. Notably, in the case of three LGL points ($N = 3$, Figs. 5a and 5b), we observe the DG-USBP method often has a higher dissipation rate than the DGSEM and consequently to have smaller kinetic energies. This implies that the DG-USBP method introduces more artificial dissipation, which also explains the improved robustness we have observed in Subsection 6.7 when using three LGL points. At the same time, for four LGL points ($N = 4$, Figs. 5c and 5d), we find the kinetic energy of the DG-USBP method to be larger than that of the DGSEM towards the end of the simulation. This implies that now the DG-USBP method introduces less artificial dissipation, which explains the reduced robustness we have observed in Subsection 6.7 when using four LGL points.

Discussion of the results

The above results indicate that the DG-USBP method introduces more artificial dissipation than DGSEM for three LGL points and less for four LGL points. This also explains the varying improvements in robustness we have observed in Subsection 6.7. Moreover, the above findings provide additional evidence that combining USBP operators with flux vector splittings generally improves the robustness of nodal DG methods less than high-order FD schemes. The above results indicate that this reduced improvement in robustness can be explained by the decreasing ratio $1/(N - 1)$ between the number of unresolved modes, which is 1, and the number of resolved modes, which is $N - 1$, decreasing as more LGL nodes are used.

7. Concluding remarks

Previous studies, such as [87,24], have demonstrated that combining high-order USBP operators with flux vector splittings can increase the robustness of FD schemes for under-resolved simulations without introducing excessive artificial dissipation. Here, we investigated whether combining USBP operators with flux vector splittings can be applied to nodal DG methods with similar success. We began by demonstrating the existence of USBP operators on arbitrary grid points and providing a straightforward procedure for their construction. While previous works focused on diagonal-norm FD-USBP operators, our discussion encompassed a broader class of USBP operators. Importantly, the proposed construction procedure is applied to arbitrary grid points and is not limited to equidistant spacing. This generalization enabled us to develop novel USBP operators on LGL points, which are well-suited for nodal DG methods. We then examined the robustness properties of the resulting DG-USBP methods for challenging examples of the compressible Euler equations, such as Kelvin-Helmholtz instabilities. Similar to high-order FD-USBP schemes, we found that combining flux vector splitting techniques with DG-USBP operators does not lead to excessive artificial dissipation. Furthermore, we demonstrated local linear/energy stability for the DG-USBP semi-discretizations in a setting where existing high-order entropy-stable flux differencing DGSEM produce unstable semi-discretizations. However, we also observed that the improvement in robustness offered by USBP operators is less significant for high-order DG methods compared to FD schemes. Specifically, the increase in robustness diminished as more LGL points were used per element and coordinate direction. We suspected this reduced improvement could be attributed to USBP methods adding dissipation only to unresolved modes. After all, FD schemes typically have more unresolved modes than nodal DG methods. Still, the DG-USBP approach presents an alternative to entropy stable flux differencing DGSEM. It is potentially more convenient for tackling complex problems where a reliable flux splitting method, like the Lax-Friedrichs one, is readily available while conducting an entropy analysis becomes particularly challenging.

While the current iteration of DG-USBP methods has limitations, there is room for improvement. Future endeavors could explore the potential of optimally selecting these operators' dissipation parameters in a manner similar to [22,111,45]. In this study, we focused on establishing the general theory, providing a construction methodology, and demonstrating the practical application of USBP operators in DG methods rather than fine-tuning the operators. Additionally, although not explicitly demonstrated here, the established general theory and construction framework for USBP operators extend to multi-dimensional SBP operators [68,43] and FSBP operators for general function spaces [34,32,33]. These avenues will form the core of future research efforts, as said methods typically use oversampling and, therefore, have more unresolved modes that can be dissipated using USBP operators. Thus, we expect USBP operators may offer greater improvements in such cases, similar to what was observed for FD schemes in [87,24].

CRedit authorship contribution statement

Jan Glaubitz: Conceptualization, Software, Writing – original draft, Writing – review & editing. **Hendrik Ranocha:** Writing – review & editing, Software, Conceptualization. **Andrew R. Winters:** Writing – review & editing, Writing – original draft, Software, Conceptualization. **Michael Schlottke-Lakemper:** Writing – review & editing, Writing – original draft, Software, Conceptualization. **Philipp Öffner:** Writing – review & editing, Conceptualization. **Gregor Gassner:** Writing – review & editing, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

JG was supported by the US DOD (ONR MURI) grant #N00014-20-1-2595. HR was supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation, project numbers 513301895 and 528753982 as well as within the DFG priority program SPP 2410 with project number 526031774) and the Daimler und Benz Stiftung (Daimler and Benz foundation, project number 32-10/22). ARW was funded through Vetenskapsrådet, Sweden grant agreement 2020-03642 VR. MSL received funding through the DFG research unit FOR 5409 “Structure-Preserving Numerical Methods for Bulk- and Interface Coupling of Heterogeneous Models (SNUBIC)” (project number 463312734), as well as through a DFG individual grant (project number 528753982). PÓ was supported by the DFG within the priority research program SPP 2410, project OE 661/5-1 (525866748) and under the personal grant 520756621 (OE 661/4-1). GG acknowledges funding through the Klaus-Tschira Stiftung via the project “HiFiLab” and received funding through the DFG research unit FOR 5409 “SNUBIC” and through the BMBF funded project “ADAPTEX”. Some of the computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research Council through grant agreement no. 2022-06725.

Appendix A. Proof of Theorem 3.1

Henceforth, let $\mathbf{x} = [x_1, \dots, x_N]^T$ be a grid on the computational domain $[x_L, x_R]$ and \mathcal{P}_d be the space of polynomials up to degree d with $d + 1 \leq N$. We first demonstrate part (b) of Theorem 3.1.

Lemma A.1 (Part (b) of Theorem 3.1). *Let $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ be degree d USBP operators with $Q_+ + Q_+^T = S$. Then, $D = (D_+ + D_-)/2$ is a degree d SBP operator and S satisfies $S\mathbf{f} = \mathbf{0}$ for all $f \in \mathcal{P}_d$.*

Proof. The first assertion, $D = (D_+ + D_-)/2$ being a degree d SBP operator, follows immediately from Lemma 2.3. The second assertion, S satisfying $Sf = \mathbf{0}$ for all $f \in \mathcal{P}_d$, has implicitly already been noted in [62,55,53]. For completeness, we briefly revisit the argument. The relation (2.5) between D_- and D_+ implies

$$D_+f - D_-f = P^{-1}Sf. \tag{A.1}$$

Furthermore, for $f \in \mathcal{P}_d$, (i) in Definition 2.2 yields $D_+f - D_-f = \mathbf{0}$. In this case, (A.1) becomes $P^{-1}Sf = \mathbf{0}$, which shows the assertion. \square

It remains to prove statement (a) of Theorem 3.1.

Lemma A.2 (Part (a) of Theorem 3.1). Let $D = P^{-1}(Q + B/2)$ be a degree d USBP operator and let S be a symmetric and negative semi-definite matrix with $Sf = \mathbf{0}$ for all $f \in \mathcal{P}_d$. Then, $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ with $Q_{\pm} = Q \pm S/2$ are degree d USBP operators with $Q_+ + Q_+^T = S$.

Proof. We must show that $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ satisfy (i), (iv), and (v) in Definition 2.2. We first demonstrate (i) in Definition 2.2. Let $f \in \mathcal{P}_d$. Since $D_{\pm} = P^{-1}(Q_{\pm} + B/2)$ and $Sf = \mathbf{0}$, we have

$$\begin{aligned} D_+f &= Df + P^{-1}Sf/2 = f', \\ D_-f &= Df - P^{-1}Sf/2 = f', \end{aligned} \tag{A.2}$$

which shows that the exactness conditions (i) in Definition 2.2 are satisfied. Next, (iv) in Definition 2.2 results from

$$Q_+ + Q_-^T = (Q + S/2) + (Q - S/2)^T = 0, \tag{A.3}$$

where the last equation follows from S being symmetric and Q being anti-symmetric (see condition (iv) in Definition 2.1). Finally, condition (v) in Definition 2.2 holds since

$$Q_+ + Q_+^T = (Q + S/2) + (Q + S/2)^T = S, \tag{A.4}$$

where the last equation follows from S being symmetric and Q being anti-symmetric. This completes the proof. \square

Theorem 3.1 follows from combining Lemmas A.1 and A.2.

Appendix B. Diagonal-norm USBP operators on LGL nodes

Below, the DG-USBP operators on four ($N = 4$), five ($N = 5$), and six ($N = 6$) LGL nodes on $\Omega_{\text{ref}} = [-1, 1]$ are presented. The operators are defined by the LGL nodes $\mathbf{x} = [x_1, \dots, x_N]^T$, the weights $\mathbf{p} = [p_1, \dots, p_N]^T$ associated with the diagonal norm matrix $P = \text{diag}(\mathbf{p})$, the central degree $N - 1$ SBP operator D , and the symmetric negative semi-definite dissipation matrix S . Recall that we get the USBP operators as $D_{\pm} = D \pm P^{-1}S/2$. Furthermore, the dissipation matrix S can be decomposed as $S = V\Lambda V^T$ with $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_N)$ and V being the Vandermonde matrix of a basis of DOPs up to degree $N - 1$.

B.1. DG-USBP operators on four LGL nodes

The four ($N = 4$) LGL nodes and associate weights on $\Omega_{\text{ref}} = [-1, 1]$ are $x_1 = -1$, $x_2 = -1/\sqrt{5}$, $x_3 = -x_2$, $x_4 = -x_1$ and $p_1 = 1/6$, $p_2 = 5/6$, $p_3 = p_2$, $p_4 = p_1$. The central SBP operator $D = [D_{n,m}]_{n,m=1}^N$ has the entries

$$\begin{aligned} D_{1,1} &= -3.0000000000000000, & D_{1,2} &= 4.045084971874736368, & D_{1,3} &= -1.545084971874737478, \\ D_{1,4} &= 0.5000000000000000, & D_{2,1} &= -0.809016994374947229, & D_{2,2} &= -0.00000000000000333, \\ D_{2,3} &= 1.118033988749894903, & D_{2,4} &= -0.309016994374947451, & D_{3,1} &= 0.309016994374947451, \\ D_{3,2} &= -1.118033988749894903, & D_{3,3} &= 0.00000000000000222, & D_{3,4} &= 0.809016994374947340, \\ D_{4,1} &= -0.5000000000000000, & D_{4,2} &= 1.545084971874737256, & D_{4,3} &= -4.045084971874737256, \\ D_{4,4} &= 3.0000000000000000. \end{aligned}$$

The Vandermonde matrix $V = [V_{n,m}]_{n,m=1}^N$ of a basis of DOPs up to degree $N - 1$ is given by

$$\begin{aligned} V_{1,1} &= 0.5000000000000000, & V_{1,2} &= -0.645497224367902800, & V_{1,3} &= 0.49999999999999889, \\ V_{1,4} &= -0.288675134594812921, & V_{2,1} &= 0.5000000000000000, & V_{2,2} &= -0.288675134594812866, \\ V_{2,3} &= -0.5000000000000000, & V_{2,4} &= 0.645497224367902800, & V_{3,1} &= 0.5000000000000000, \end{aligned}$$

$$\begin{aligned} V_{3,2} &= 0.288675134594812866, & V_{3,3} &= -0.500000000000000000, & V_{3,4} &= -0.645497224367902800, \\ V_{4,1} &= 0.500000000000000000, & V_{4,2} &= 0.645497224367902800, & V_{4,3} &= 0.499999999999999889, \\ V_{4,4} &= 0.288675134594812921. \end{aligned}$$

Finally, once $\lambda_1, \dots, \lambda_N$ have been fixed, we get the dissipation matrix as $S = V \Lambda V^T$ and the DG-USBP operators as $D_{\pm} = D \pm P^{-1} S / 2$.

B.2. DG-USBP operators on five LGL nodes

The five ($N = 5$) LGL nodes and associate weights on $\Omega_{\text{ref}} = [-1, 1]$ are $x_1 = -1, x_2 = -\sqrt{3/7}, x_3 = 0, x_4 = -x_2, x_5 = -x_1$ and $p_1 = 1/10, p_2 = 49/90, p_3 = 32/45, p_4 = p_2, p_5 = p_1$. The central SBP operator $D = [D_{n,m}]_{n,m=1}^N$ has the entries

$$\begin{aligned} D_{1,1} &= -5.000000000000000000, & D_{1,2} &= 6.756502488724240862, & D_{1,3} &= -2.6666666666666667407, \\ D_{1,4} &= 1.410164177942426988, & D_{1,5} &= -0.500000000000000000, & D_{2,1} &= -1.240990253030982648, \\ D_{2,2} &= -0.000000000000000444, & D_{2,3} &= 1.745743121887939342, & D_{2,4} &= -0.763762615825973379, \\ D_{2,5} &= 0.259009746969017129, & D_{3,1} &= 0.374999999999999889, & D_{3,2} &= -1.336584577695453246, \\ D_{3,3} &= 0.000000000000000000, & D_{3,4} &= 1.336584577695453024, & D_{3,5} &= -0.37499999999999833, \\ D_{4,1} &= -0.259009746969017129, & D_{4,2} &= 0.763762615825973268, & D_{4,3} &= -1.745743121887939342, \\ D_{4,4} &= 0.000000000000000444, & D_{4,5} &= 1.240990253030982648, & D_{5,1} &= 0.500000000000000111, \\ D_{5,2} &= -1.410164177942426766, & D_{5,3} &= 2.6666666666666667407, & D_{5,4} &= -6.756502488724240862, \\ D_{5,5} &= 5.000000000000000000, \end{aligned}$$

The Vandermonde matrix $V = [V_{n,m}]_{n,m=1}^N$ of a basis of DOPs up to degree $N - 1$ is given by

$$\begin{aligned} V_{1,1} &= 0.447213595499957928, & V_{1,2} &= -0.591607978309961591, & V_{1,3} &= 0.500000000000000000, \\ V_{1,4} &= -0.387298334620741758, & V_{1,5} &= 0.223606797749978908, & V_{2,1} &= 0.447213595499957928, \\ V_{2,2} &= -0.387298334620741702, & V_{2,3} &= -0.166666666666666685, & V_{2,4} &= 0.591607978309961591, \\ V_{2,5} &= -0.521749194749950962, & V_{3,1} &= 0.447213595499957928, & V_{3,2} &= 0.000000000000000000, \\ V_{3,3} &= -0.666666666666666741, & V_{3,4} &= 0.000000000000000000, & V_{3,5} &= 0.596284793999943941, \\ V_{4,1} &= 0.447213595499957928, & V_{4,2} &= 0.387298334620741702, & V_{4,3} &= -0.166666666666666685, \\ V_{4,4} &= -0.591607978309961591, & V_{4,5} &= -0.521749194749950962, & V_{5,1} &= 0.447213595499957928, \\ V_{5,2} &= 0.591607978309961591, & V_{5,3} &= 0.500000000000000000, & V_{5,4} &= 0.387298334620741758, \\ V_{5,5} &= 0.223606797749978908. \end{aligned}$$

Finally, once $\lambda_1, \dots, \lambda_N$ have been fixed, we get the dissipation matrix as $S = V \Lambda V^T$ and the DG-USBP operators as $D_{\pm} = D \pm P^{-1} S / 2$.

B.3. DG-USBP operators on six LGL nodes

The six ($N = 6$) LGL nodes and associate weights on $\Omega_{\text{ref}} = [-1, 1]$ are $x_1 = -1, x_2 = -\sqrt{(7 + 2\sqrt{7})/21}, x_3 = -\sqrt{(7 - 2\sqrt{7})/21}, x_4 = -x_3, x_5 = -x_2, x_6 = -x_1$ and $p_1 = 1/15, p_2 = (14 - \sqrt{7})/30, p_3 = (14 + \sqrt{7})/30, p_4 = p_3, p_5 = p_2, p_6 = p_1$. The central SBP operator $D = [D_{n,m}]_{n,m=1}^N$ has the entries

$$\begin{aligned} D_{1,1} &= -7.500000000000000000, & D_{1,2} &= 10.141415936319667424, & D_{1,3} &= -4.036187270305348740, \\ D_{1,4} &= 2.244684648176165975, & D_{1,5} &= -1.349913314190487768, & D_{1,6} &= 0.500000000000000111, \\ D_{2,1} &= -1.786364948339095315, & D_{2,2} &= 0.000000000000001110, & D_{2,3} &= 2.523426777429455203, \\ D_{2,4} &= -1.152828158535929015, & D_{2,5} &= 0.653547507429800167, & D_{2,6} &= -0.237781177984231401, \\ D_{3,1} &= 0.484951047853569239, & D_{3,2} &= -1.721256952830232834, & D_{3,3} &= 0.000000000000000222, \\ D_{3,4} &= 1.752961966367866165, & D_{3,5} &= -0.786356672223240794, & D_{3,6} &= 0.269700610832039001, \\ D_{4,1} &= -0.269700610832039001, & D_{4,2} &= 0.786356672223240682, & D_{4,3} &= -1.752961966367865942, \\ D_{4,4} &= 0.000000000000000222, & D_{4,5} &= 1.721256952830233278, & D_{4,6} &= -0.484951047853569350, \end{aligned}$$

$$\begin{aligned}
 D_{5,1} &= 0.237781177984231346, & D_{5,2} &= -0.653547507429800167, & D_{5,3} &= 1.152828158535929237, \\
 D_{5,4} &= -2.523426777429455203, & D_{5,5} &= -0.000000000000000888, & D_{5,6} &= 1.786364948339095537, \\
 D_{6,1} &= -0.500000000000000000, & D_{6,2} &= 1.349913314190487768, & D_{6,3} &= -2.244684648176165531, \\
 D_{6,4} &= 4.036187270305348740, & D_{6,5} &= -10.141415936319669200, & D_{6,6} &= 7.500000000000000888.
 \end{aligned}$$

The Vandermonde matrix $V = [V_{n,m}]_{n,m=1}^N$ of a basis of DOPs up to degree $N - 1$ is given by

$$\begin{aligned}
 V_{1,1} &= 0.408248290463863073, & V_{1,2} &= -0.547722557505166074, & V_{1,3} &= 0.483045891539647831, \\
 V_{1,4} &= -0.408248290463863017, & V_{1,5} &= 0.316227766016837830, & V_{1,6} &= -0.182574185835055497, \\
 V_{2,1} &= 0.408248290463863073, & V_{2,2} &= -0.419038058655589740, & V_{2,3} &= 0.032338332982759031, \\
 V_{2,4} &= 0.367654222400928044, & V_{2,5} &= -0.576443896275456780, & V_{2,6} &= 0.435014342463467985, \\
 V_{3,1} &= 0.408248290463863073, & V_{3,2} &= -0.156227735687855862, & V_{3,3} &= -0.515384224522407175, \\
 V_{3,4} &= 0.445155822251155409, & V_{3,5} &= 0.260216130258618783, & V_{3,6} &= -0.526715472069829382, \\
 V_{4,1} &= 0.408248290463863073, & V_{4,2} &= 0.156227735687855862, & V_{4,3} &= -0.515384224522407175, \\
 V_{4,4} &= -0.445155822251155409, & V_{4,5} &= 0.260216130258618783, & V_{4,6} &= 0.526715472069829382, \\
 V_{5,1} &= 0.408248290463863073, & V_{5,2} &= 0.419038058655589740, & V_{5,3} &= 0.032338332982759031, \\
 V_{5,4} &= -0.367654222400928044, & V_{5,5} &= -0.576443896275456780, & V_{5,6} &= -0.435014342463467985, \\
 V_{6,1} &= 0.408248290463863073, & V_{6,2} &= 0.547722557505166074, & V_{6,3} &= 0.483045891539647831, \\
 V_{6,4} &= 0.408248290463863017, & V_{6,5} &= 0.316227766016837830, & V_{6,6} &= 0.182574185835055497.
 \end{aligned}$$

Finally, once $\lambda_1, \dots, \lambda_N$ have been fixed, we get the dissipation matrix as $S = V\Lambda V^T$ and the DG-USBP operators as $D_{\pm} = D_{\pm} P^{-1} S / 2$.

Appendix C. More examples of USBP operators

We exemplify the construction of USBP operators on the reference element $\Omega_{\text{ref}} = [-1, 1]$. Henceforth, we round all reported numbers to the second decimal place for ease of presentation.

C.1. USBP operators on Gauss–Legendre points

We demonstrate the construction of a degree two ($d = 2$) diagonal-norm USBP operator on four ($N = 4$) Gauss–Legendre points in $\Omega_{\text{ref}} = [-1, 1]$. Notably, the Gauss–Legendre points do not include the boundary points.

Let us first address (P1). We start from a degree three diagonal-norm SBP operator, obtained by evaluating the derivatives of the corresponding Lagrange basis functions at the Gauss–Legendre grid points. The associated points and quadrature weights (diagonal entries of P) are $\mathbf{x} \approx \frac{1}{5}[-43, -17, 17, 43]$ and $\mathbf{p} \approx \frac{1}{20}[7, 13, 13, 7]$, respectively. Moreover, the SBP and boundary operator are

$$D \approx \frac{1}{100} \begin{bmatrix} 333 & 486 & -211 & 58 \\ -76 & -38 & 147 & -33 \\ 33 & -147 & 38 & 76 \\ -58 & 211 & -486 & 333 \end{bmatrix}, \quad B \approx \frac{1}{100} \begin{bmatrix} -153 & 81 & -40 & 11 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -11 & 40 & -81 & 153 \end{bmatrix}. \tag{C.1}$$

Note that the first and last row of B corresponds to the function values of the Lagrange basis functions at the left and right boundary point, $x_L = -1$ and $x_R = 1$, respectively.

We next address (P2), finding a dissipation matrix S with $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_2$ and $\mathbf{f}^T S\mathbf{f} < 0$ otherwise. We follow Subsection 4.3 and robustly construct S using a DOP basis on the four Gauss–Legendre grid points. The resulting Vandermonde matrix of this DOP basis is

$$V \approx \begin{bmatrix} 1/2 & -33/50 & 1/2 & -13/50 \\ 1/2 & -13/50 & -1/2 & 33/50 \\ 1/2 & 13/50 & -1/2 & -33/50 \\ 1/2 & 33/50 & 1/2 & 13/50 \end{bmatrix}. \tag{C.2}$$

Consequently, we get the dissipation matrix as $S = V\Lambda V^T$ with $\Lambda = \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$. We choose the first three eigenvalues as zero to ensure that the USBP operator has degree two. Furthermore, we choose $\lambda_4 = -1$, which introduces artificial dissipation to the highest, unresolved, mode.⁴ The resulting dissipation matrix is

⁴ Note that $\lambda_4 = -1$ is an arbitrary choice. Any negative value yields an admissible dissipation matrix.

$$S \approx \frac{1}{100} \begin{bmatrix} -7 & 17 & -17 & 7 \\ 17 & -43 & 43 & -17 \\ -17 & 43 & -43 & 17 \\ 7 & -17 & 17 & -7 \end{bmatrix}. \quad (\text{C.3})$$

Finally, we get degree two diagonal-norm USBP operators on $[-1, 1]$ as $D_{\pm} = D \pm P^{-1}S/2$, which yields

$$D_+ \approx \frac{1}{100} \begin{bmatrix} -343 & 511 & -235 & 68 \\ -63 & -72 & 180 & -46 \\ 20 & -114 & 5 & 89 \\ -48 & 186 & -461 & 81 \end{bmatrix}, \quad (\text{C.4})$$

$$D_- \approx \frac{1}{100} \begin{bmatrix} -324 & 461 & -186 & 48 \\ -89 & -5 & 114 & -20 \\ 46 & -180 & 72 & 63 \\ -68 & 235 & -511 & 343 \end{bmatrix}.$$

The norm and boundary matrix of the degree two USBP operators are the same as those of the degree three SBP operator.

C.2. Dense-norm USBP operators on equidistant points

We exemplify our construction procedure for USBP operators using a dense-norm (the norm matrix P is non-diagonal) SBP operator of degree three ($d = 3$) on four ($N = 4$) equidistant points on $[0, 1]$. The degree three dense-norm SBP operator we start from was derived in [19, Section 4.2] and has the following norm and derivative matrices:

$$P \approx \frac{1}{8} \begin{bmatrix} 2 & 1 & 0 & 0 \\ 1 & 10 & -2 & 0 \\ 0 & -2 & 10 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix}, \quad D \approx \frac{1}{6} \begin{bmatrix} -11 & 18 & -9 & 2 \\ -2 & -3 & 6 & -1 \\ 1 & -6 & 3 & 2 \\ -2 & 9 & -18 & 11 \end{bmatrix}. \quad (\text{C.5})$$

Let us address (P2), finding a dissipation matrix S with $S\mathbf{f} = \mathbf{0}$ if $f \in \mathcal{P}_2$ and $\mathbf{f}^T S \mathbf{f} < 0$ otherwise. Once more, we follow subsection 4.3 and robustly construct S using a DOP basis on the four equidistant grid points. The resulting Vandermonde matrix of this DOP basis is

$$V \approx \begin{bmatrix} 1/2 & -67/100 & 1/2 & -22/100 \\ 1/2 & -22/100 & -1/2 & 67/100 \\ 1/2 & 22/100 & -1/2 & -67/100 \\ 1/2 & 67/100 & 1/2 & 22/100 \end{bmatrix}. \quad (\text{C.6})$$

Consequently, we get the dissipation matrix as $S = V\Lambda V^T$ with $\Lambda = \text{diag}(\lambda_1, \lambda_2, \lambda_3, \lambda_4)$. We choose the first three eigenvalues as zero to get USBP operators with degree two ($d = 2$). Furthermore, we choose $\lambda_4 = -1$, which introduces artificial dissipation to the highest, unresolved mode. The resulting dissipation matrix is

$$S \approx \frac{1}{20} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -9 & 9 & -3 \\ -3 & 9 & -9 & 3 \\ 1 & -3 & 3 & -1 \end{bmatrix}. \quad (\text{C.7})$$

Finally, we get degree two dense-norm USBP operators on four equidistant points on $[0, 1]$ as $D_{\pm} = D \pm P^{-1}S/2$, which yields

$$D_+ \approx \frac{1}{100} \begin{bmatrix} -189 & 317 & -167 & 39 \\ -30 & -59 & 109 & -20 \\ 14 & -91 & 41 & 36 \\ -28 & 133 & -283 & 178 \end{bmatrix}, \quad (\text{C.8})$$

$$D_- \approx \frac{1}{100} \begin{bmatrix} -178 & 283 & -133 & 28 \\ -36 & -41 & 91 & -14 \\ 20 & -109 & 59 & 30 \\ -39 & 167 & -317 & 189 \end{bmatrix}.$$

The norm and boundary matrix of the degree two USBP operators are the same as those of the degree three SBP operator.

Data availability

The code we use is openly available at <https://github.com/trixi-framework/paper-2024-generalized-upwind-sbp>.

References

- [1] R. Abgrall, J. Nordström, P. Öffner, S. Tokareva, Analysis of the SBP-SAT stabilization for finite element methods part I: linear problems, *J. Sci. Comput.* 85 (2020) 1–29.
- [2] R. Abgrall, J. Nordström, P. Öffner, S. Tokareva, Analysis of the SBP-SAT stabilization for finite element methods part II: entropy stability, *Commun. Appl. Math. Comput.* 5 (2023) 573–595.
- [3] R. Abgrall, P. Rai, F. Renac, A discontinuous Galerkin spectral element method for a nonconservative compressible multicomponent flow model, *J. Comput. Phys.* 472 (2023) 31.
- [4] J. Ahrens, B. Geveci, C. Law, ParaView: an end-user tool for large-data visualization, in: *The Visualization Handbook*, Elsevier, 2005, pp. 717–731.
- [5] W.K. Anderson, J.L. Thomas, B. Van Leer, Comparison of finite volume flux vector splittings for the Euler equations, *AIAA J.* 24 (1986) 1453–1460.
- [6] J. Bezanson, A. Edelman, S. Karpinski, V.B. Shah, Julia: a fresh approach to numerical computing, *SIAM Rev.* 59 (2017) 65–98.
- [7] P. Buning, J. Steger, Solution of the two-dimensional Euler equations with generalized coordinate transformation using flux vector splitting, in: *3rd Joint Thermophysics, Fluids, Plasma and Heat Transfer Conference*, 1982, p. 971.
- [8] M.H. Carpenter, T.C. Fisher, E.J. Nielsen, S.H. Frankel, Entropy stable spectral collocation schemes for the Navier–Stokes equations: discontinuous interfaces, *SIAM J. Sci. Comput.* 36 (2014) B835–B867.
- [9] M.H. Carpenter, M. Parsani, T.C. Fisher, E.J. Nielsen, Entropy stable staggered grid spectral collocation for the Burgers’ and compressible Navier–Stokes equations, Technical Memorandum NASA/TM–2015–218990, NASA, NASA Langley Research Center, Hampton, Virginia 23681-2199, 2015.
- [10] J. Chan, D.C. Del Rey Fernández, M.H. Carpenter, Efficient entropy stable Gauss collocation methods, *SIAM J. Sci. Comput.* 41 (2019) A2938–A2966.
- [11] T. Chen, C.-W. Shu, Entropy stable high order discontinuous Galerkin methods with suitable quadrature rules for hyperbolic conservation laws, *J. Comput. Phys.* 345 (2017) 427–461.
- [12] T. Chen, C.-W. Shu, Review of entropy stable discontinuous Galerkin methods for systems of conservation laws on unstructured simplex meshes, *CSIAM Trans. Appl. Math.* 1 (2020) 1–52.
- [13] S. Christ, D. Schwabeneder, C. Rackauckas, M.K. Borregaard, T. Breloff, Plots.jl — a user extendable plotting API for the Julia programming language, *J. Open Res. Softw.* (2023).
- [14] A. Cicchino, D.C.D.R. Fernández, S. Nadarajah, J. Chan, M.H. Carpenter, Provably stable flux reconstruction high-order methods on curvilinear elements, *J. Comput. Phys.* 463 (2022) 111259.
- [15] A. Cicchino, S. Nadarajah, D.C.D.R. Fernández, Nonlinearly stable flux reconstruction high-order methods in split form, *J. Comput. Phys.* 458 (2022) 111094.
- [16] W. Coirier, B. Van Leer, Numerical flux formulas for the Euler and Navier–Stokes equations. II-progress in flux-vector splitting, in: *10th Computational Fluid Dynamics Conference*, 1991, p. 1566.
- [17] S. Conde, I. Fekete, J.N. Shadid, Embedded pairs for optimal explicit strong stability preserving Runge–Kutta methods, *J. Comput. Appl. Math.* 412 (2022) 114325.
- [18] F. Coquel, C. Marmignon, P. Rai, F. Renac, An entropy stable high-order discontinuous Galerkin spectral element method for the Baer–Nunziato two-phase flow model, *J. Comput. Phys.* 431 (2021) 29.
- [19] D.C. Del Rey Fernández, P.D. Boom, D.W. Zingg, A generalized framework for nodal first derivative summation-by-parts operators, *J. Comput. Phys.* 266 (2014) 214–239.
- [20] D.C. Del Rey Fernández, J.E. Hicken, D.W. Zingg, Review of summation-by-parts operators with simultaneous approximation terms for the numerical solution of partial differential equations, *Comput. Fluids* 95 (2014) 171–196.
- [21] D.C. Del Rey Fernández, J.E. Hicken, D.W. Zingg, Simultaneous approximation terms for multi-dimensional summation-by-parts operators, *J. Sci. Comput.* 75 (2018) 83–110.
- [22] N. Discacciati, J.S. Hesthaven, D. Ray, Controlling oscillations in high-order discontinuous Galerkin schemes using artificial viscosity tuned by neural networks, *J. Comput. Phys.* 409 (2020) 109304.
- [23] L. Dogvilovich, I. Sofronov, High-accuracy finite-difference schemes for solving elastodynamic problems in curvilinear coordinates within multiblock approach, *Appl. Numer. Math.* 93 (2015) 176–194.
- [24] K. Duru, D. Stewart, N. Lee, preprint, arXiv:2411.06629, 2024.
- [25] A. Eisinberg, G. Fedele, Discrete orthogonal polynomials on Gauss–Lobatto Chebyshev nodes, *J. Approx. Theory* 144 (2007) 238–246.
- [26] T.C. Fisher, M.H. Carpenter, High-order entropy stable finite difference schemes for nonlinear conservation laws: finite domains, *J. Comput. Phys.* 252 (2013) 518–557.
- [27] U.S. Fjordholm, S. Mishra, E. Tadmor, Arbitrarily high-order accurate entropy stable essentially nonoscillatory schemes for systems of conservation laws, *SIAM J. Numer. Anal.* 50 (2012) 544–573.
- [28] G.J. Gassner, A skew-symmetric discontinuous Galerkin spectral element discretization and its relation to SBP-SAT finite difference methods, *SIAM J. Sci. Comput.* 35 (2013) A1233–A1253.
- [29] G.J. Gassner, M. Svärd, F.J. Hindenlang, Stability issues of entropy-stable and/or split-form high-order schemes: analysis of linear stability, *J. Sci. Comput.* 90 (2022) 1–36.
- [30] G.J. Gassner, A.R. Winters, D.A. Kopriva, Split form nodal discontinuous Galerkin schemes with summation-by-parts property for the compressible Euler equations, *J. Comput. Phys.* 327 (2016) 39–66.
- [31] W. Gautschi, *Orthogonal Polynomials: Computation and Approximation*, OUP, Oxford, 2004.
- [32] J. Glaubitz, S.-C. Klein, J. Nordström, P. Öffner, Multi-dimensional summation-by-parts operators for general function spaces: theory and construction, *J. Comput. Phys.* 491 (2023) 112370.
- [33] J. Glaubitz, S.-C. Klein, J. Nordström, P. Öffner, Summation-by-parts operators for general function spaces: the second derivative, *J. Comput. Phys.* (2024) 112889.
- [34] J. Glaubitz, J. Nordström, P. Öffner, Summation-by-parts operators for general function spaces, *SIAM J. Numer. Anal.* 61 (2023) 733–754.
- [35] J. Glaubitz, J. Nordström, P. Öffner, Energy-stable global radial basis function methods on summation-by-parts form, *J. Sci. Comput.* 98 (2024).
- [36] J. Glaubitz, J. Nordström, P. Öffner, An optimization-based construction procedure for function space based summation-by-parts operators on arbitrary grids, preprint, arXiv:2405.08770, 2024.
- [37] J. Glaubitz, P. Öffner, Stable discretisations of high-order discontinuous Galerkin methods on equidistant and scattered points, *Appl. Numer. Math.* 151 (2020) 98–118.
- [38] J. Glaubitz, H. Ranocha, A.R. Winters, M. Schlottke-Lakemper, P. Öffner, G.J. Gassner, Reproducibility repository for “Generalized upwind summation-by-parts operators and their application to nodal discontinuous Galerkin methods”, <https://github.com/trixi-framework/paper-2024-generalized-upwind-sbp>, 2024, <https://doi.org/10.5281/zenodo.11661785>.
- [39] G.H. Golub, C.F. Van Loan, *Matrix Computations*, vol. 4, JHU Press, 2013.
- [40] W.J. Gordon, C.A. Hall, Construction of curvilinear coordinate systems and applications to mesh generation, *Int. J. Numer. Methods Eng.* 7 (1973) 461–477.
- [41] D. Hänel, R. Schwane, G. Seider, On the accuracy of upwind schemes for the solution of the Navier–Stokes equations, in: *8th Computational Fluid Dynamics Conference*, American Institute of Aeronautics and Astronautics, 1987, p. 1105.
- [42] A. Harten, P.D. Lax, B.v. Leer, On upstream differencing and Godunov-type schemes for hyperbolic conservation laws, *SIAM Rev.* 25 (1983) 35–61.

- [43] J.E. Hicken, D.C. Del Rey Fernández, D.W. Zingg, Multidimensional summation-by-parts operators: general theory and application to simplex elements, *SIAM J. Sci. Comput.* 38 (2016) A1935–A1958.
- [44] J.E. Hicken, D.W. Zingg, Summation-by-parts operators and high-order quadrature, *J. Comput. Appl. Math.* 237 (2013) 111–125.
- [45] D. Hillebrand, S.-C. Klein, P. Öffner, Applications of limiters, neural networks and polynomial annihilation in higher-order FD/FV schemes, *J. Sci. Comput.* 97 (2023) 13.
- [46] B.F. Klose, G.B. Jacobs, D.A. Kopriva, Assessing standard and kinetic energy conserving volume fluxes in discontinuous Galerkin formulations for marginally resolved Navier–Stokes flows, *Comput. Fluids* 205 (2020) 104557.
- [47] D.A. Kopriva, Metric identities and the discontinuous spectral element method on curvilinear meshes, *J. Sci. Comput.* 26 (2006) 301–327.
- [48] D.A. Kopriva, A.R. Winters, M. Schlottke-Lakemper, H. Ranocha, HOHQMesh.jl: a Julia frontend to the fortran-based hohqmesh mesh generator for high order elements, <https://github.com/trixi-framework/HOHQMesh.jl>, 2024, <https://doi.org/10.5281/zenodo.13959071>.
- [49] J.F.B.M. Kraaijevanger, Contractivity of Runge–Kutta methods, *BIT Numer. Math.* 31 (1991) 482–528.
- [50] H.-O. Kreiss, G. Scherer, Finite element and finite difference methods for hyperbolic partial differential equations, in: *Mathematical Aspects of Finite Elements in Partial Differential Equations*, Elsevier, 1974, pp. 195–212.
- [51] H.-O. Kreiss, G. Scherer, On the existence of energy estimates for difference approximations for hyperbolic systems, Technical report, 1977.
- [52] P.G. Lefloch, J.-M. Mercier, C. Rohde, Fully discrete, entropy conservative schemes of arbitrary order, *SIAM J. Numer. Anal.* 40 (2002) 1968–1992.
- [53] V. Linders, On an eigenvalue property of summation-by-parts operators, *J. Sci. Comput.* 93 (2022) 82.
- [54] V. Linders, T. Lundquist, J. Nordström, On the order of accuracy of finite difference operators on diagonal norm based summation-by-parts form, *SIAM J. Numer. Anal.* 56 (2018) 1048–1063.
- [55] V. Linders, J. Nordström, S.H. Frankel, Properties of Runge–Kutta-summation-by-parts methods, *J. Comput. Phys.* 419 (2020) 109684.
- [56] M.-S. Liou, C.J. Steffen Jr, High-order polynomial expansions (HOPE) for flux-vector splitting, Technical Memorandum NASA/TM-104452, NASA, NASA Lewis Research Center, Cleveland, OH, United States, 1991.
- [57] D. Lodaes, J. Manzanero, E. Ferrer, E. Valero, An entropy-stable discontinuous Galerkin approximation of the Spalart–Allmaras turbulence model for the compressible Reynolds averaged Navier–Stokes equations, *J. Comput. Phys.* 455 (2022) 110998.
- [58] L. Lundgren, K. Mattsson, An efficient finite difference method for the shallow water equations, *J. Comput. Phys.* 422 (2020) 109784.
- [59] J. Manzanero, G. Rubio, E. Ferrer, E. Valero, D.A. Kopriva, Insights on aliasing driven instabilities for advection equations with application to Gauss–Lobatto discontinuous Galerkin methods, *J. Sci. Comput.* 75 (2018) 1262–1281.
- [60] J. Manzanero, G. Rubio, D.A. Kopriva, E. Ferrer, E. Valero, An entropy-stable discontinuous Galerkin approximation for the incompressible Navier–Stokes equations with variable density and artificial compressibility, *J. Comput. Phys.* 408 (2020) 109241.
- [61] J. Manzanero, G. Rubio, D.A. Kopriva, E. Ferrer, E. Valero, Entropy-stable discontinuous Galerkin approximation with summation-by-parts property for the incompressible Navier–Stokes/Cahn–Hilliard system, *J. Comput. Phys.* 408 (2020) 109363.
- [62] K. Mattsson, Diagonal-norm upwind SBP operators, *J. Comput. Phys.* 335 (2017) 283–310.
- [63] K. Mattsson, O. O’Reilly, Compatible diagonal-norm staggered and upwind SBP operators, *J. Comput. Phys.* 352 (2018) 52–75.
- [64] K. Mattsson, M. Svärd, J. Nordström, Stable and accurate artificial dissipation, *J. Sci. Comput.* 21 (2004) 57–79.
- [65] T. Montoya, D.W. Zingg, A unifying algebraic framework for discontinuous Galerkin and flux reconstruction methods based on the summation-by-parts property, *J. Sci. Comput.* 92 (2022) 87.
- [66] T. Montoya, D.W. Zingg, Efficient entropy-stable discontinuous spectral-element methods using tensor-product summation-by-parts operators on triangles and tetrahedra, *J. Comput. Phys.* 516 (2024) 113360.
- [67] J. Nordström, Conservative finite difference formulations, variable coefficients, energy estimates and artificial dissipation, *J. Sci. Comput.* 29 (2006) 375–404.
- [68] J. Nordström, M. Björck, Finite volume approximations and strict stability for hyperbolic problems, *Appl. Numer. Math.* 38 (2001) 237–255.
- [69] J. Nordström, K. Forsberg, C. Adamsson, P. Eliasson, Finite volume methods, unstructured meshes and strict stability for hyperbolic problems, *Appl. Numer. Math.* 45 (2003) 453–473.
- [70] J. Nordström, A.A. Ruggitu, On conservation and stability properties for summation-by-parts schemes, *J. Comput. Phys.* 344 (2017) 451–464.
- [71] G. Ntoukas, J. Manzanero, G. Rubio, E. Valero, E. Ferrer, An entropy-stable p-adaptive nodal discontinuous Galerkin for the coupled Navier–Stokes/Cahn–Hilliard system, *J. Comput. Phys.* 458 (2022) 111093.
- [72] P. Öffner, J. Glaubitz, H. Ranocha, Stability of correction procedure via reconstruction with summation-by-parts operators for Burgers’ equation using a polynomial chaos approach, *ESAIM: Math. Model. Numer. Anal.* 52 (2018) 2215–2245.
- [73] P. Öffner, H. Ranocha, Error boundedness of discontinuous Galerkin methods with variable coefficients, *J. Sci. Comput.* 79 (2019) 1572–1607.
- [74] S. Ortleb, On the stability of IMEX upwind gSBP schemes for 1D linear advection-diffusion equations, *Commun. Appl. Math. Comput.* (2023).
- [75] M. Parsani, R. Boukharfane, I.R. Nolasco, D.C.D.R. Fernández, S. Zampini, B. Hadri, L. Dalcin, High-order accurate entropy-stable discontinuous collocated Galerkin methods with the summation-by-parts property for compressible CFD frameworks: scalable SSDC algorithms and flow solver, *J. Comput. Phys.* 424 (2021) 109844.
- [76] M. Parsani, M.H. Carpenter, T.C. Fisher, E.J. Nielsen, Entropy stable staggered grid discontinuous spectral collocation methods of any order for the compressible Navier–Stokes equations, *SIAM J. Sci. Comput.* 38 (2016) a3129–a3162.
- [77] P.-O. Persson, J. Peraire, Sub-cell shock capturing for discontinuous Galerkin methods, in: *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006, p. 112.
- [78] T. Qin, C.-W. Shu, Implicit positivity-preserving high-order discontinuous Galerkin methods for conservation laws, *SIAM J. Sci. Comput.* 40 (2018) A81–A107.
- [79] C. Rackauckas, Q. Nie, *DifferentialEquations.jl* – a performant and feature-rich ecosystem for solving differential equations in Julia, *J. Open Res. Softw.* 5 (2017) 15.
- [80] H. Ranocha, Comparison of some entropy conservative numerical fluxes for the Euler equations, *J. Sci. Comput.* 76 (2018) 216–242.
- [81] H. Ranocha, *SummationByPartsOperators.jl*: a Julia library of provably stable semidiscretization techniques with mimetic properties, *J. Open Sour. Softw.* 6 (2021) 3454.
- [82] H. Ranocha, L. Dalcin, M. Parsani, D.I. Ketcheson, Optimized Runge–Kutta methods with automatic step size control for compressible computational fluid dynamics, *Commun. Appl. Math. Comput.* 4 (2022) 1191–1228.
- [83] H. Ranocha, G.J. Gassner, Preventing pressure oscillations does not fix local linear stability issues of entropy-based split-form high-order schemes, *Commun. Appl. Math. Comput.* (2021) 1–24.
- [84] H. Ranocha, J. Glaubitz, P. Öffner, T. Sonar, Stability of artificial dissipation and modal filtering for flux reconstruction schemes using summation-by-parts operators, *Appl. Numer. Math.* 128 (2018) 1–23.
- [85] H. Ranocha, P. Öffner, T. Sonar, Summation-by-parts operators for correction procedure via reconstruction, *J. Comput. Phys.* 311 (2016) 299–328.
- [86] H. Ranocha, M. Schlottke-Lakemper, A.R. Winters, E. Faulhaber, J. Chan, G.J. Gassner, Adaptive numerical simulations with Trixi.jl: a case study of Julia for scientific computing, in: *Proceedings of the JuliaCon Conferences*, vol. 1, 2022, p. 77.
- [87] H. Ranocha, A.R. Winters, M. Schlottke-Lakemper, P. Öffner, J. Glaubitz, G.J. Gassner, On the robustness of high-order upwind summation-by-parts methods for nonlinear conservation laws, *J. Comput. Phys.* 520 (2025) 113471.
- [88] F. Renac, Entropy stable DGSEM for nonlinear hyperbolic systems in nonconservative form with application to two-phase flows, *J. Comput. Phys.* 382 (2019) 1–26.
- [89] F. Renac, Entropy stable, robust and high-order DGSEM for the compressible multicomponent Euler equations, *J. Comput. Phys.* 445 (2021) 28.

- [90] J. Revels, M. Lubin, T. Papamarkou, Forward-mode automatic differentiation in Julia, preprint, arXiv:1607.07892, 2016.
- [91] D. Rojas, R. Boukharfane, L. Dalcin, D.C.D.R. Fernández, H. Ranocha, D.E. Keyes, M. Parsani, On the robustness and performance of entropy stable discontinuous collocation methods, *J. Comput. Phys.* 426 (2021) 109891.
- [92] A. Rueda-Ramírez, G. Gassner, A subcell finite volume positivity-preserving limiter for DGSEM discretizations of the Euler equations, in: WCCM-ECCOMAS2020, 2021.
- [93] G. Scherer, On energy estimates for difference approximations to hyperbolic partial differential equations, PhD thesis, Uppsala University, 1977.
- [94] N. Shima, Y. Kuya, Y. Tamaki, S. Kawai, Preventing spurious pressure oscillations in split convective form discretization for compressible flows, *J. Comput. Phys.* 427 (2021) 110060.
- [95] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, Final Report NASA/CR-97-206253, vol. 11, NASA, Institute for Computer Applications in Science and Engineering, NASA Langley Research Center, Hampton VA United States, 1997.
- [96] B. Sjögren, H. Yee, High order entropy conservative central schemes for wide ranges of compressible gas dynamics and MHD flows, *J. Comput. Phys.* 364 (2018) 153–185.
- [97] J.L. Steger, R. Warming, Flux vector splitting of the inviscid gasdynamic equations with application to finite difference methods, Technical Memorandum NASA/TM-78605, NASA, NASA Ames Research Center; Moffett Field, CA, United States, 1979.
- [98] V. Stiernström, L. Lundgren, M. Nazarov, K. Mattsson, A residual-based artificial viscosity finite difference method for scalar conservation laws, *J. Comput. Phys.* 430 (2021) 110100.
- [99] B. Strand, Summation by parts for finite difference approximations for d/dx , *J. Comput. Phys.* 110 (1994) 47–67.
- [100] M. Svård, On coordinate transformations for summation-by-parts operators, *J. Sci. Comput.* 20 (2004) 29–42.
- [101] M. Svård, J. Nordström, Review of summation-by-parts schemes for initial-boundary-value problems, *J. Comput. Phys.* 268 (2014) 17–38.
- [102] E. Tadmor, The numerical viscosity of entropy stable schemes for systems of conservation laws. I, *Math. Comput.* 49 (1987) 91–103.
- [103] E. Tadmor, Entropy stability theory for difference approximations of nonlinear conservation laws and related time-dependent problems, *Acta Numer.* 12 (2003) 451–512.
- [104] E.F. Toro, *Riemann Solvers and Numerical Methods for Fluid Dynamics: A Practical Introduction*, Springer Science & Business Media, 2013.
- [105] B. van Leer, Flux-vector splitting for the Euler equations, in: E. Krause (Ed.), *Eighth International Conference on Numerical Methods in Fluid Dynamics*, in: *Lecture Notes in Physics*, vol. 170, Springer, 1982, pp. 507–512.
- [106] B. van Leer, Flux-vector splitting for the 1990s, in: NASA, Lewis Research Center, *Computational Fluid Dynamics Symposium on Aeropropulsion*, 1991.
- [107] M. Vinokur, H. Yee, Extension of efficient low dissipation high order schemes for 3-d curvilinear moving grids, in: *Frontiers of Computational Fluid Dynamics 2002*, World Scientific, 2002, pp. 129–164.
- [108] M.R. Visbal, D.V. Gaitonde, High-order-accurate methods for complex unsteady subsonic flows, *AIAA J.* 37 (1999) 1231–1239.
- [109] M.W. Wilson, On the Hahn polynomials, *SIAM J. Math. Anal.* 1 (1970) 131–139.
- [110] G. Yan, S. Kaur, J.W. Banks, J.E. Hicken, Entropy-stable discontinuous Galerkin difference methods for hyperbolic conservation laws, *J. Comput. Appl. Math.* 422 (2023) 114885.
- [111] J. Zeifang, A. Beck, A data-driven high order sub-cell artificial viscosity for the discontinuous Galerkin spectral element method, *J. Comput. Phys.* 441 (2021) 110475.
- [112] X. Zhang, C.-W. Shu, Maximum-principle-satisfying and positivity-preserving high-order schemes for conservation laws: survey and new developments, *Proc. R. Soc. A, Math. Phys. Eng. Sci.* 467 (2011) 2752–2776.